



15th International Conference on System Theory, Control and Computing

(Joint Conference SINTES 15, SACCS 11, SIMSIS 15)

**October 14 – 16, 2011
Sinaia, ROMANIA**

Editor: Mihail Voicu

Organizers:

- "Gheorghe Asachi" Technical University of Iași, Faculty of Automatic Control and Computer Engineering
- University of Craiova, Faculty of Automation, Computers and Electronics, Automatic Control Research Centre
- "Dunărea de Jos" University of Galați, Faculty of Automatic Control, Computers, Electrical and Electronics Engineering



**Technically co-sponsored by
IEEE Control System Society**

COMMITTEES

INTERNATIONAL PROGRAM COMMITTEE

Mihail Voicu

Department of Automatic Control and Applied Informatics
“Gheorghe Asachi” Technical University of Iași, Romania

General Chair**Rogelio Lozano**

Laboratoire HEUDIASYC, UMR-CNRS 6599
Université de Technologie de Compiègne, France

Program Chair**Silviu-Ioan Niculescu**

Laboratoire des signaux et systems (L2S), Supelec, France

CSS Representative**Vladimir Răsvan (RO)**

Department of Automatic Control, University of Craiova, Romania

Vice Chair**Vasile Manta (RO)**

Department of Computer Engineering
“Gheorghe Asachi” Technical University of Iași, Romania

Vice Chair**Adrian Filipescu (RO)**

Department of Automation and Industrial Informatics
“Dunărea de Jos” University of Galați, Romania

Vice Chair

MEMBERS

Mihail Abrudean (RO)

Costin Badica (RO)

Steve Banks (UK)

Viorel Barbu (RO)

Andrzej Bartoszewicz (PL)

Ioana Banicescu (USA)

Sergiu Caraman (RO)

Petru Cașcaval (RO)

Emil Ceangă (RO)

Ali Charara (FR)

Vincent Charvillat (FR)

Dorian Cojocaru (RO)

Sorin Cotofana (NL)

Mitică Craus (RO)

Vladimir Crețu (RO)

Valentin Cristea (RO)

Radu Dobrescu (RO)

Luc Dugard (FR)

Ioan Dumitrache (RO)

Florin Filip (RO)

Pasi Franti (FI)

Gerhard Freiling (DE)

Marian Gheorghe (UK)

Gheorghe Grigoras (RO)

Eduard Groeller (AT)

Remus Hanea (NL)

Mircea Ivănescu (RO)

Magnus Jonsson (SE)

Robin De Keyser (BE)

Vladimir L. Kharitonov (RU)

Gheorghe Lazea (RO)

Corneliu Lazăr (RO)

Frank L. Lewis (USA)

Viorel Mînză (RO)

Sergiu Nedevschi (RO)

Sorin Olaru (FR)

Nejat Olğac (USA)

Octavian Păstrăvanu (RO)

Emil Petre (RO)

Dan Popescu (RO)

Dumitru Popescu (RO)

Rodica Potolea (RO)

Alexander Poznyak (MX)

Radu-Emil Precup (RO)

Stefan Preitl (RO)

Werner Purgathofer (AT)

Sorel Reisman (USA)

Bogdan Sapinski (PL)

Dan Selișteanu (RO)

Manuel Silva (ES)

Vasile Sima (RO)

Horia Teodorescu (RO)

Marek Tudruj (PL)

Dan Tușiș (RO)

Nicolae Țepuș (RO)

Erik Verriest (USA)

Lucian Vintan (RO)

Draguna Vrabie (USA)

NATIONAL ORGANIZING COMMITTEE

Doru Pănescu (RO) - Chairman

Alexandru Archip (RO)

Eugen Bobașu (RO)

Adrian Burlacu (RO)

Daniela Cernega (RO)

Simona Caraiman (RO)

Lavinia Ferariu (RO)

Florin Leon (RO)

Letiția Mirea (RO)

Viorel Nicolau (RO)

Elvira Popescu (RO)

Elena Serban (RO)

Florina Ungureanu (RO)

WELCOME MESSAGE

It is already an established fact that the System Theory, Control and Computing have and will have an increasing impact on our development. And this fact includes not only the technical environment, but also the biological, ecological, economical, social, political, societal spheres, and, as a natural consequence, the global human society and the future of *homo sapiens sapiens*. Accordingly, the purpose of the 2011 15th International Conference on System Theory, Control and Computing (ICSTCC) is to bring specialists together and to facilitate the ideas exchange in these areas. We are confident that the activities during this meeting will give many good stimuli to all the participants to discuss recent developments, to provide information on problems that have to be solved, and to create and develop new scientific bridges and human links for future co-operations.

Our conference is for the second time managed in the actual format, joining three previous distinct scientific meetings. These have been organized in the following forms:

- ❖ the International Symposium on Automatic Control and Computer Science (SACCS) by the Faculty of Automatic Control and Computer Engineering of the “Gheorghe Asachi” Technical University of Iași

- ❖ the International Conference on System Theory and Control (SINTES) by the Faculty of Automation, Computers and Electronics of the University of Craiova

- ❖ the International Symposium on Modelling, Simulation and Identification Systems (SIMSIS) by the Faculty of Computer Science of the "Dunărea de Jos" University of Galați.

As these events had covered the most important and related fields from Automatic Control and Computers, by their gathering it was expected that the quality and interest can be improved. This fact has been already supported by the first edition of the joint Conference development in October 2010, which was technically co-sponsored by IEEE Control Systems Society (www.ieeecss.org/conferences/concluded) and it also is by the development of this second edition. Now, besides the technical co-sponsorship of IEEE Control Systems Society (www.ieeecss.org/conferences), we have obtained the permission to include, after the end of the conference, all the presented papers in the IEEE Xplore data base.

The final programme of this Conference comprises 4 plenary lectures and 112 contributed papers organized in 19 parallel sessions, among which 3 invited sessions are included. As one can see from the programme, you have an excellent job by attending the plenary lectures, by selecting the sessions and the papers of interest, by presenting your papers and by participating to discussions. Apart from this, our staffs are entirely dedicated to make your stay during the conference as pleasant as possible.

I would like to thank all the contributors and participants, the scientific and financial sponsors, who generously supported the preparation of the conference, to the International Program Committee and especially to the reviewers for the very dedicated work and, last but not least, to the Organizing Committee for the very meticulous and applied work during the last 10 months.

On behalf of the International Program Committee and of the Organizing Committee, I would like to take this opportunity to welcome all attendees and participants to the 2011 15th International Conference on System Theory, Control and Computing (ICSTCC). We are very pleased and honored to host our conference in these beautiful surroundings and we wish you a fruitful and pleasant stay in Sinaia.

Mihail Voicu

General Chair of ICSTCC 2011

LIST OF AUTHORS AND PAPERS

Hesham Abdel-Ghaffar, Yasser Shokry, Ahmed Hassan, Sherif Hammad and Hazem Abbas <i>Neural Generalized Predictive Controller Stability Analysis.....</i>	9
Ioan Agavriloaei, Adrian Alexandrescu and Mitica Craus <i>Improving Web Clustering through a New Modeling for Web Documents.....</i>	15
Adrian Alexandrescu, Ioan Agavriloaei and Mitica Craus <i>A Genetic Algorithm for Mapping Tasks in Heterogeneous Computing Systems.....</i>	21
Andrei Amariutei and Simona Caraiman <i>Parallel Quantum Computer Simulation on the GPU.....</i>	27
Bogdan Apostol and Vasile Manta <i>Linear diffusion for generating plant distribution in natural environment simulation.....</i>	33
Alexandru Archip and Mitica Craus <i>The Fast Itemset Miner: A Detailed Analysis of the Candidate Generation Stages.....</i>	40
Ovidiu Aritoni and Viorel Negru <i>Multi-Threading System for Human Behavior Simulation based on Constraints Propagation ..</i>	45
Timotei Asaftei and Mihaela-Hanako Matcovschi <i>On the Boundedness of Timed Continuous Petri Nets.....</i>	51
Maen Atli, Zied Achour, Alexandre Sava and Kondo H. Adjallah <i>Control law for Timed Marked Graph constrained by Marking Exclusion Constraint Or/And ...</i>	57
Cezar Babici and Alexandru Onea <i>A Rule-Based Energy Management Strategy for Parallel Hybrid Vehicles with Supercapacitors</i>	63
Costin Badica and Mihnea Scafes <i>Experimental Evaluation of Service Negotiation Protocols in Agent Systems.....</i>	69
Vadim Baitoiu, Alexandru Barleanu and Andrei Stan <i>Floating-point to fixed-point code conversion with variable trade-off between computational complexity and accuracy loss.....</i>	75
Naim Bajcinca and Y. Kouhi <i>Decentralized set-valued state estimation and prediction for hybrid systems: A symbolic approach.....</i>	81
Andreea E. Balau and Corneliu Lazăr <i>Predictive control of an electro-hydraulic actuated wet clutch.....</i>	87
Marian Barbu, Viorel Minzu, Daniela Carp and Emil Ceanga <i>Identification and Sensitivity Analysis of a Trickling Biofilter Viewed as a Distributed Parameters System.....</i>	93
Tudor Barbu <i>Automatic Unsupervised Shape Recognition Technique using Moment Invariants.....</i>	99
Viorel Barbu <i>Stabilization of Navier–Stokes equations by oblique boundary feedback controllers (extended abstract)</i>	103
Lucian-Florentin Barbulescu <i>An algorithm designed to determine the optimum number of communication channels in a modular simulator.....</i>	105
Corneliu Botan and Florin Ostafi <i>Current Controlled Optimal Electrical Drive System.....</i>	110
Alexandru Ionut Botas and Razvan Solea <i>Design and Control of a Biped Robot.....</i>	115
Nicolae Botezatu, Vasile Manta and George Vieriu <i>A Clustering Algorithm for Negotiating Security in Wireless Mesh Networks.....</i>	120
Florin Catalin Braescu, Lavinia Ferariu and Andrei Franciuc <i>Monitoring CAN Performances in Distributed Embedded Systems.....</i>	126

Cristina Budaciu and Corneliu Lazar <i>The influence of Finite Word Length Representation of the Delta Predictive Controller Design for Automotive Powertrain Systems.....</i>	132
Elena Bunciu, Emil Petre and Viorel Adrian Stanciu <i>Interval observer used in state estimation for a class of biodegradable polymers.....</i>	138
Adrian Burlacu, Mihai Obada and Vladimir Tiganas <i>An Image Based Visual Servoing Scheme for Wheeled Mobile Robots.....</i>	144
Constantin - Florin Caruntu, Doina Onu and Corneliu Lazar <i>Real-time Simulation of a Vehicle Drivetrain Controlled through CAN using a Robust MPC Strategy.....</i>	149
Gabriel Antonio Perez Castañeda, Nicolae Brinzei and Jean-Francois Aubry <i>Influence of supervisory control on the dependability of a hybrid system.....</i>	156
Elena Cimpoeșu, Bogdan Ciubotaru and Dumitru Popescu <i>Model Following with Output Feedback for Fault Tolerant Control.....</i>	162
Valeriu Codreanu, Lucian Petrica and Radu Hobincu <i>Increasing Vector Processor Pipeline Efficiency with a Thread-Interleaved Controller.....</i>	168
Mihaela Colhon <i>Parallel Treebank From Word-Aligned Bilingual Corpus. Language Engineering for Phrasal Alignments.....</i>	172
Salvatore Cuomo, Pasquale De Michele and Marta Chinnici <i>Parallel Tools and Techniques for Biological Cells Modelling.....</i>	178
Gilney Damm and Van Cuong Nguyen <i>MIMO Conditional Integrator Control for a Class of Nonlinear Systems.....</i>	184
Constantin-Catalin Dosoftei and Lucian Mastacan <i>Optimal Positioning System with Fuzzy Logic Controller.....</i>	190
Claudia-Adina Dragos, Stefan Preitl, Radu-Emil Precup, Emil M. Petriu, Mircea-Bogdan Radac and Alexandra-Iulia Stinean <i>Alternative Control Solutions for Vehicles with Continuously Variable Transmission. A Case Study.....</i>	194
Radu Duma, Petru Dobra, Mirela Dobra and Ioan Valentin Sita <i>Low Cost Embedded Solution for BLDC Motor Control.....</i>	200
Bogdan Dumitrascu, Adrian Filipescu, Viorel Minzu and A. Filipescu Jr. <i>Backstepping Control of Wheeled Mobile Robots.....</i>	206
Sorin Dumitru, Dorian Cojocaru and Florin Manta <i>Experiments in Determining the Displacements and Angular Amplitudes of a Hyper-Redundant Robot.....</i>	212
Abhishek Dutta, Robin De Keyser, Yu Zhong, Bart Wyns, Gregory Pinte and Julian Stoev <i>Robust predictive control of a wet-clutch using evolutionary algorithm optimized engagement profile.....</i>	218
Marius Enachescu, George Voicu and Sorin Cotofana <i>Leakage-enhanced 3D-Stacked NEMFET-based Power Management Architecture for Autonomous Sensors Systems.....</i>	224
Mona Faraji-Niri, Mohammad Reza Jahed-Motlagh and Houman Sadjadian <i>Relay Feedback Identification of SOPDT Transfer Functions with NMP Zero.....</i>	230
Lavinia Ferariu and Bogdan Burlacu <i>Multiobjective Graph Genetic Programming with Encapsulation Applied to Neural System Identification.....</i>	238
Dorian Florescu, Mihaela Matcovschi and Octavian Pastravanu <i>Improving Rohn Approximation for the Real Parts of Interval Matrix Eigenvalues.....</i>	244
Matteo Gagliolo, Kevin Van Vaerenbergh, Abdel Rodríguez, Ann Nowé, Stijn Goossens, Gregory Pinte and Wim Symens <i>Policy search reinforcement learning for automatic wet clutch engagement.....</i>	250
Marius Gavrilescu, Vasile Manta and Eduard Groeller <i>Gradient-based Classification and Representation of Features from Volume Data.....</i>	256

Morteza Ghateei, Bahram Karimi, Ghazanfar Shahgholian and Masoud Mosayebi <i>Singular perturbation approach to RBNN adaptive control of unknown flexible joint manipulators</i>	262
Narcis Ghita, Marius Kloetzer and Octavian Pastravanu <i>Probabilistic car-like robot path planning from temporal logic specifications.....</i>	267
Sorin Mihai Grigorescu, Gigel Macesanu, Tiberiu Teodor Cocias and Florin Moldoveanu <i>On the Real-time Modelling of a Robotic Scene Perception and Estimation System.....</i>	273
Paul Herghelegiu, Vasile Manta and Eduard Gröller <i>Needle Stability Maps for Brain Tumor Biopsies.....</i>	277
Mircea Hulea <i>The mathematical model of a biologically inspired electronic neuron for ease the design of spiking neural networks topology.....</i>	282
Cristina Huzum and Petru Cascaval <i>A Multibackground March Test For All Static Simple Neighborhood Pattern-Sensitive Faults in RAMs.....</i>	288
Przemyslaw Ignaciuk and Andrzej Bartoszewicz <i>Modeling and Control of Continuous-Review Perishable Inventory Systems with Multiple Supply Alternatives.....</i>	294
Lucian Mihai Itu, Constantin Suci, Florin Moldoveanu and Adrian Postelnicu <i>GPU optimized computation of the artificial compressibility method.....</i>	300
Mirjana Ivanovic, Dejan Mitrovic, Zoran Budimac and Milan Vidakovic <i>Metadata harvesting learning resources - an agent-oriented approach.....</i>	306
Petros Karamanakos, Georgios Papafotiou and Stefanos Manias <i>Model Predictive Control of the Interleaved DC-DC Boost Converter.....</i>	312
Patrick Kühtreiber, Martin Knecht and Christoph Traxler <i>BRDF approximation and estimation for Augmented Reality.....</i>	318
Florin Leon, Bogdan Aignatoaiei and Andrei Leca <i>A General Evolutionary Optimization Framework for Intelligent Agents.....</i>	324
Lucian Luncean <i>An overview on Interrelated Negotiations.....</i>	330
Robert Gabriel Lupu, Radu Bozomitu, Florina Ungureanu and Vlad Cehan <i>Eye Tracking Based Communication System for Patient with Major Neuro-locomotor Disabilities.....</i>	336
Gigel Macesanu, Sorin Mihai Grigorescu and Vasile Comnac <i>Time-delay Analysis of a Robotic Stereo Active Vision System.....</i>	341
Sorin Manoiu-Olaru, Mircea Nitulescu and Viorel Stoian <i>Hexapod Robot. Mathematical Support for Modeling and Control.....</i>	347
Sorin Manoiu-Olaru and Mircea Nitulescu <i>Hexapod Robot. Virtual Models for Preliminary Studies.....</i>	353
Florin Manta, Sorin Dumitru and Dorian Cojocaru <i>Closed loop control for a 3D tentacle robot using artificial vision system.....</i>	359
Dan Adrian Marior and Constantin Circiumaru <i>Management workflow for consultants in a company.....</i>	365
Mihaela-Hanako Matcovschi, Octavian Pastravanu and Mihail Voicu <i>On Some Properties of Diagonally Stable Polytopic Systems.....</i>	371
F. A. de Alencar Menezes and Giovanni Barroso <i>Fusion of Supervisors of Discrete Event Systems Using Coloured Petri Nets.....</i>	377
Jován O. Mérida and Luis T. Aguilar <i>Nonlinear H_∞ Control of Variable Speed Wind Turbines for Power Regulation and Load Reduction.....</i>	382
Letiția Mirea <i>Fault Diagnosis Using Hybrid Wavelet/ Elman Neural Networks.....</i>	388
Radu Miron, Tiberiu Letia and Mihai Hulea <i>Two Server Topologies for a Distributed Fingerprint-Based Recognition System.....</i>	394

Vlad Muresan, Mihail Abrudean and Tiberiu Colosi	
<i>Modeling and Simulation of the billet's heating process in a furnace with rotary hearth.....</i>	400
Pal-Stefan Murvay and Ioan Silea	
<i>Design and implementation of a reliable gas leak detection system for domestic environments.....</i>	406
Alex Muscar and Costin Bădică	
<i>Investigating F# as a development tool for distributed multi-agent systems.....</i>	412
Andreea Maria Neaca and Mitica Iustinian Neaca	
<i>Modeling and simulation of the heating elements into an electrical oven.....</i>	418
Van Cuong Nguyen and Gilney Damm	
<i>MIMO Conditional Integrator Control for Unmanned Airlaunch.....</i>	424
Eitaku Nobuyama, Takahiko Aoyagi and Yasushi Kami	
<i>Robust Control Synthesis of Bilinear Systems with Nonlinear Weights by Sum of Squares Optimization.....</i>	430
Virginia Ecaterina Oltean, Radu Dobrescu, Dan Popescu, Maximilian Nicolae and Andreea Udrea	
<i>Some Aspects Concerning the Oscillations Damping in a Class of Affine Systems – a Driveline Simulation Example.....</i>	436
Andreea - Raluca Olteanu, Ionel - Cristinel Simion and Nicu - George Bizdoaca	
<i>On Robotic Emotional Behavior.....</i>	444
Mihaela Oprea, Elia Dragomir and Madalina Carbureanu	
<i>On the Use of Collaborative Intelligence in an Agent-based Environmental Monitoring and Analysis System.....</i>	450
Carlos Pascal and Doru Panescu	
<i>On Resource Allocation in a Holonic Manufacturing Execution System.....</i>	456
Lucian Petrescu, Anca Morar, Florica Moldoveanu and Victor Asavei	
<i>Real Time Reconstruction of Volumes from Very Large Datasets Using CUDA.....</i>	462
Gregory Pinte, Julian Stoev, Wim Symens, Abhishek Dutta, Yu Zhong, Bart Wyls, Robin De Keyser, Bruno Depraetere, Jan Swevers, Matteo Gagliolo and Ann Nowé	
<i>Learning strategies for wet clutch control.....</i>	467
Dumitru Popescu, Madalina Mircioiu and Adrian Moise	
<i>Numerical Control for Petrochemical Installations.....</i>	475
Elvira Popescu and Laurentiu Manafu	
<i>Repurposing a Wiki for Collaborative Learning - Pedagogical and Technical View.....</i>	481
Dan Puiu and Florin Moldoveanu	
<i>Real-time Path Planner for Arm Robot Collision Avoidance.....</i>	487
Alexandru Ionut Pustianu, Adriana Serbencu and Daniela Cristina Cernega	
<i>Mobile Robot Control Using Face Recognition Algorithms.....</i>	493
Adrian Radaschin, Adrian Filipescu, Viorel Minzu, Eugenia Minca and Adriana Filipescu Jr.	
<i>Adaptive disassembly sequence control by using mobile robots and system information.....</i>	499
Lala H. Rajaoarisoa, Jean François Balmat and Nacer K. M'Sirdi	
<i>Identifiability and identification of a class of hybrid system.....</i>	505
George Constantin Rascanu and Razvan Solea	
<i>Electric Wheelchair Control for People with Locomotor Disabilities Using Eye Movements.....</i>	511
Vladimir Rasvan	
<i>Forced periodic oscillations and almost linear behavior.....</i>	516
Cristina Resceanu	
<i>Control algorithms for multi-legged robots in fault conditions using fuzzy logic.....</i>	522
Cristian Rotariu, Hariton Costin, Gladiola Andrusac, R. Ciobotariu and F. Adochiei	
<i>An Integrated System for Wireless Monitoring of Chronic Patients and Elderly People.....</i>	526
Liviu Sandu, Raluca Sboru, Sorin Ilie and Costin Badica	
<i>Scalable Distributed Agent-Based English Auction Server.....</i>	531
Enrique Santiago, Emmanuel Witrant, Marc Goniche and Frédéric Clairet	
<i>An Optimal Feedback Approach to Particle Source Identification in Tokamaks.....</i>	537

Miloš Savić, Mirjana Ivanović and Miloš Radovanović <i>Connectivity properties of the Apache Ant class collaboration network.....</i>	544
Dorin Sendrescu and Eugen Bobasu <i>Model Predictive Control of a Biotechnological Process.....</i>	550
Manuel de Silva and Cristian Mahulea <i>Fluidization and Fluid views of Discrete Event Systems.....</i>	554
Vasile Sima <i>Recent Developments in SLICOT Library and Related Tools.....</i>	564
Cristian Smochina, Vasile Manta and Walter Kropatsch <i>Sampling step importance in hierarchical semantic segmentation of microscopic images.....</i>	570
Razvan Solea and Daniela Cristina Cernega <i>Sliding-mode Control of Coordinated Nonholonomic Mobile Manipulators.....</i>	576
Andrei Stan, Alexandru Valachi and Alexandru Barleanu <i>The design of a run-time monitoring structure for a MPSoC.....</i>	582
Roxana Stanica and Emil Petre <i>Achieve Faster Spanning Tree Convergence.....</i>	586
Florin Stinga and Dan Popescu <i>Predictive Control of a bioprocess – a hybrid approach.....</i>	592
Viorel Stoian, Mircea Nitulescu and Sorin Manoiu-Olaru <i>Hybrid Control for a Hyper-Redundant Robot in a Climbing Task.....</i>	598
Zhen Sun and Zhenyu Yang <i>System Identification for Nonlinear FOPDT Model with Input-Dependent Dead-Time.....</i>	603
Ioan Susnea and Grigore Vasiliu <i>On Defining and Following Robot Paths in a RFID Augmented Environment.....</i>	609
Oleg Tarabuta <i>Use of “Petri Nets System” Concept in Modeling Dynamics with Increased Complexity.....</i>	614
Catalin-Stefan Teodorescu and Sorin Olaru <i>Issues in Modeling and Control of Electro-Pneumatic EGR Valve Actuator for Automotive Engineering.....</i>	620
Ionut Urziceanu, Fernando Herrero-Carrón, Juan González-Gómez, Mircea Nitulescu, Francisco de Borja Rodríguez and Pablo Varona <i>Central Pattern Generator Control of a Differential Wheeled Robot.....</i>	628
Alexandru Valachi, Bogdan-Ioan Aignatoaiei and Mihai Timis <i>The Comparative Study of Two Analytical Methods for Detection and Elimination of the Static Hazard in Combinational Logic Circuits.....</i>	634
Gabriela Varvara and Violeta Sandu <i>Implementation and Covering Analysis for The T-way Combinatorial Testing Strategy.....</i>	638
Boban Vesin, Mirjana Ivanović, Aleksandra Klačnja-Milićević and Zoran Budimac <i>Rule-based Reasoning for Altering Pattern Navigation in Programming Tutoring System</i>	644
Gabriel Vladuț, Monica Mateescu, Liana Simona Sbirna, Sebastian Sbirna and Anisoara Oubraham <i>Statistical analysis and modeling of the PM₁₀ concentration values in Dolj County (Romania) between 2008 and 2010.....</i>	650
Karol Waga, Andrei Tabarcea and Pasi Fränti <i>Context Aware Recommendation of Location-based Data.....</i>	658
Hamed Yahoui, Dorin Popescu, Jean-Marc Thiriet, Olivier Bonnaud, Tony Ward and Cristian Perra <i>ELLEIEC Project: Network of Networks in Electrical and Information Engineering - an European Area of Education for New Skills for New Jobs.....</i>	664
Zizilia Zamudio, Rogelio Lozano, Jorge Torres and Jose-Luis Rullan-Lara <i>Vision based stabilisation of a quadrotor using nested saturation control approach</i>	671
Index of authors.....	677
List of reviewers.....	680

Neural Generalized Predictive Controller Stability Analysis

Hesham Abdel-Ghaffar (hesham.fouad@invensys.com), Yasser Shoukry
(yasser_shoukry@eng.asu.edu.eg), Ahmed Hassan (ahyousef@ieee.org), Sherif Hammad
(sherif_hammad@mentor.com), Hazem Abbas (hazem_abbas@mentor.com)

Abstract— this paper emphasizes on the stability analysis of the Neural Generalized Predictive Controller (NGPC) algorithm using Lyapunov methods. NGPC is a hybrid combination between the well known GPC algorithm and a Feed Forward Multi Layer Perceptron (FF MLP) neural network model identifier. This combination leads to a better stability characteristics in the closed loop systems in the presence of high nonlinearities in the process. In this paper, we prove the stability characteristics of NGPC and then present simulation results showing the efficiency of using NGPC over ordinary GPC.

Index Terms—Neural Generalized Predictive Controller, Neural Identifier, Lyapunov Stability, and Newton-Raphson.

I. INTRODUCTION

THE generalized predictive control (GPC) belongs to a class of long range model-based predictive control (MPC). The control objective is the minimization of a quadratic criterion involving future inputs and outputs. The minimization is done in a receding horizon sense in which controller computes sequence of future control signals each sample; however the first optimized one is only applied. GPC requires a reasonably accurate system identification model that captures the essential nonlinearities of the process under control [1]. In many practical applications, restrict mathematical model based on physical principles is either unknown or too complicated to be used for control. In this case, nonlinear system identification is an inevitable step in a nonlinear GPC project. Although neural GPC (NGPC) may theoretically consume more processing time, efficient numerical methods have been implemented to overcome this issue [2].

The previous work of GPC was focusing on pure linear processes or approximated linear around set point for nonlinear processes [3]. This problem has been resolved using artificial neural network (ANN) in identifying complex nonlinear processes. Therefore for nonlinear plants, the ability of the GPC to make accurate predictions can be enhanced if a neural network (NN) is used to learn the

dynamics of the plant [4]-[6]. Another problem discovered from the fact that nonlinear predictor or Cost Function Minimization (CFM) algorithms used inside neural MPC are massive time consuming and may not be convenient for fast processes. The Newton-Raphson algorithm [7]-[9] resolved this computation problem using efficient quadratic minimizing algorithm converging better than others in less iterations times.

Many publications have been issued regarding theory of neural predictive controllers [6]-[12]. In some of them, a detail mathematical structure comparison between GPC and NGPC was presented [7], [11]. However, only few papers focused on nonlinear stability analysis of such controllers [13]-[15]. Among those studied stability, very few papers studied the Lyapunov stability analysis of GPC using recurrent neural network and Adaptive Learning Rate (ALR) algorithm [14]. To the best of our knowledge, there is lack of literature coverage for the Feed Forward Multi Layer Perceptron (FF MLP) neural GPC with Newton Raphson algorithm.

In this paper, new formula of Lyapunov stability analysis for NGPC using Newton-Raphson CFM algorithm is derived. Also simulation comparison between traditional GPC & NGPC regarding the capability of controlling severe nonlinear process is presented. This paper is organized as follow. Section II presents the structure of NGPC and neural network model building of nonlinear process. Section III focuses on the Lyapunov mathematical analysis and stability conditions. In section IV paper presents simulations showing the comparisons between traditional GPC using Recursive Least Square (RLS) identification methods and nonlinear predictor neural GPC. Section V, concludes the results and stability limits.

II. NEURAL GPC STRUCTURE

The NGPC analyzed is based on Feed Forward MLP neural network consists of two-layers: hidden and output layer. The hidden layer composed of number (L) neurons each has hyperbolic tangent transfer function; whereas the output layer has one neuron with linear transfer function. The neural network plant model uses previous inputs and previous plant outputs to predict future values of the plant output. The structure of the neural network model is given in Fig. 1, where $u(t)$ is the system input, $y_p(t)$ is the plant output, $y_m(t)$ is the neural network model plant output, the blocks labeled TDL are tapped delay lines that store previous values of the input signal, (IW_{ij}) is the weight

Manuscript received May 30, 2011.

(Neural Generalized Predictive Controller Stability Analysis).

H. Abdel-Ghaffar, corresponding author, is with Invensys Engineering Excellence Centers, Cairo, Egypt (phone: 20-101411316; fax: (+202) 25177053 ; e-mail: hesham.fouad@Invensys.com).

Y. Shoukry, and A. Hassan, are with Computers and Systems Engineering, Ain Shams University, Cairo, Egypt
(e-mail: yasser_shoukry@eng.asu.edu.eg ; ahyousef@ieee.org).

S. Hammad and H.Abbas, are with Mentor Graphics, Heliopolis, Cairo, Egypt (e-mail: {sherif_hammad, hazem_abbas}@mentor.com).

matrix from the input j to the layer i . ($LW_{i,j}$) is the weight matrix from the layer j to the layer i . This network can be trained off-line in batch mode, using data collected from the operation of the plant. In this work, offline training of neural network has been executed using one of famous training techniques called Levenberg-Marquardt (LM). The LM algorithm is one of most effective and fast convergence neural network training algorithms [6].

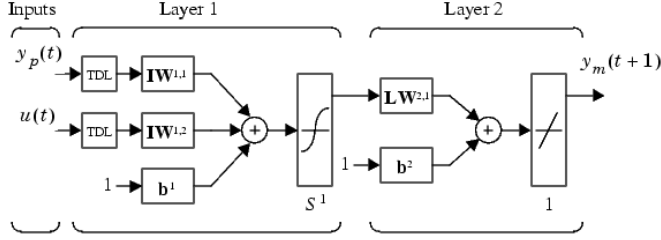


Fig. 1: NGPC feed forward MLP neural network.

The Neural Generalized Predictive Control (NGPC) structure can be seen in Fig. 2. It consists of four components, the plant to be controlled, a reference model that specifies the desired performance of the plant, a neural network that models the plant, and the Cost Function Minimization (CFM) algorithm that determines the input needed to produce the plant's desired performance. The NGPC controller algorithm, represented by dotted line below, consists of the CFM block and the neural network block. In this paper, the Newton-Raphson cost function minimization method is used. The main cost of the Newton-Raphson algorithm is in the calculation of the Hessian (inverse of second derivative of cost function). Although there is overhead time consumed in calculating the Hessian, the low iteration numbers of Newton-Raphson algorithm make it a faster algorithm for real-time control.

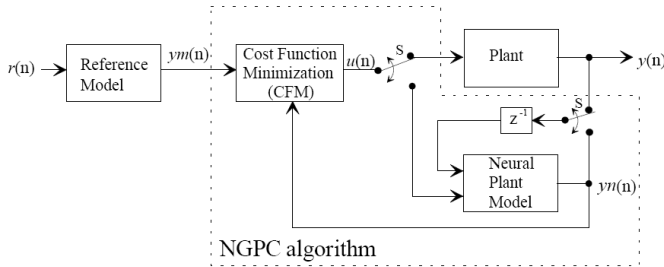


Fig. 2: Neural GPC structure block diagram.

The NGPC system starts with the input signal, $r(n)$, which is presented to the reference model. This model produces a tracking reference signal, $y_m(n)$, that is used as an input to the Cost Function Minimization (CFM) block. The CFM algorithm produces an output that is either used as an input to the plant or the plant's neural model. The double pole double throw switch, S , is set to the plant when the CFM algorithm has solved for the best input, $u(n)$, that will minimize a specified cost function. Between samples, the switch is set to the plant neural model where the CFM algorithm uses this model to calculate the next control input, $u(n+1)$, from predictions response of neural model. Once the

cost function is minimized, this input is passed to the plant. The NGPC cost function can be represented by:

$$J(N_1, N_2, N_u) = \delta \sum_{j=N_1}^{N_2} (y_d(k+j) - \hat{y}(k+j))^2 + \lambda \sum_{j=1}^{N_u} (\Delta u(k+j-1))^2 \quad (1)$$

Where:

N_1 = Minimum Prediction Horizon ($N_1=d+1$).

d = Open loop Input/output process delay. In this paper $d=0$. N_2 = Maximum Prediction Horizon. This parameter has been taken large enough to absorb effect of variable time delay.

N_u = Maximum Control Horizon.

$\hat{y}(k)$ = Predicted Output from Neural Network at sample time (k). This is equivalent to y_n in Fig. 2.

$u(k)$ = Manipulated Control Input at sample time (k).

$\Delta u(k)$ = Variation in control input at sample time (k) and equals the difference between $u(k) - u(k-1)$.

$y_d(k)$ = Reference Trajectory at sample time (k). This is equivalent to y_m in Fig. 2.

δ and λ = Weighing Factor for tracking error and control input respectively. For simplicity we consider λ constant across all sample times and $\delta = 1$.

This cost function minimizes not only the mean squared error between the reference signal and the plant's model, but also the weighted squared rate of change of the control input. When this cost function is minimized, a control input that meets the constraints is generated that allows the plant to track the reference trajectory within minor tolerance rely on both configured search parameter (α) and non-configured neural network modeling error (ε) as in Table I.

III. LYAPUNOV STABILITY ANALYSIS

Assuming discrete nonlinear state space equations:

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k));$$

$$\mathbf{y}_p(k) = \mathbf{h}(\mathbf{x}(k), \mathbf{u}(k)) \quad (2)$$

Let us assume SISO NGPC as neural MPC controller has admissible control input $u(k)$. The admissible term here for $u(k)$ means that control input should verify stability according to Lyapunov theorem as well as minimizing predictive cost function of NGPC in (1). Putting (1) in vector notation lead to:

$$J(k) = [\mathbf{Y}_d(k) - \mathbf{Y}(k)]^T * [\mathbf{Y}_d(k) - \mathbf{Y}(k)] + \lambda \Delta \mathbf{U}(k)^T * \Delta \mathbf{U}(k) \quad (3)$$

Where:

- $\mathbf{Y}_d(k)$ is the reference trajectory at instant (k), $\mathbf{Y}_d(k) = [y_d(k+1) \ y_d(k+2) \dots y_d(k+N_2)]^T$

- $\mathbf{Y}(k)$ is the predicted horizon NN output at instant (k), $\mathbf{Y}(k) = [\hat{y}(k+1) \ \hat{y}(k+2) \dots \hat{y}(k+N_2)]^T$

- $\Delta U(k)$ is the control trajectory variation at instant (k),
 $\Delta U(k) = [\Delta u(k+1) \ \Delta u(k+2) \dots \Delta u(k+N_u)]^T$;
Such that: $\Delta u(k+1) = [u(k+1) - u(k)]$ and
 $\Delta u(k+N_u) = [u(k+N_u) - u(k+N_u-1)]$

- $U(k)$ is the predicted control input at instant (k),
 $U(k) = [u(k+1) \ u(k+2) \dots u(k+N_u)]^T$

- Control weighing factor $\lambda \geq 0$, $N_1=1$ and $N_2 \geq N_u$.

Also let us assume the Lyapunov candidate function as:

$$L(k) = [Y_d(k) - Y(k)]^T * [Y_d(k) - Y(k)]$$

$$L(k) = E^T(k) * E(k) \quad (4)$$

Where $E(k) = [e(k+1) \ e(k+2) \dots e(k+N_2)]$ and
 $e(k+1) = y_d(k+1) - y(k+1)$.

It is apparent that $L(k) > 0$ positive definite, which is the first necessary condition of Lyapunov stability for closed loop NGPC. The second necessary condition is $\Delta L(k) \leq 0$.

$$\Delta L(k) = L(k+1) - L(k) = 2\Delta E^T(k) * E(k) + \Delta E^T(k) * \Delta E(k) \quad (5)$$

Where $E(k) = [Y_d(k) - Y(k)]$; $E(k+1) = E(k) + \Delta E$, and,

$$\Delta E(k) = \frac{\partial E(k)}{\partial U(k)} * \Delta U(k) = -\frac{\partial Y(k)}{\partial U(k)} * \Delta U(k) = -G(k) * \Delta U(k) \quad (6)$$

Where $\frac{\partial Y(k)}{\partial U(k)} = G(k)$ represent controller gain matrix and has size $[N_2 \times N_u]$.

From matrix algebra controller gain can be represented as:

$$G(k) = \begin{pmatrix} \frac{\partial[y^{\wedge}(k+1)]}{\partial u(k+1)} & \dots & \frac{\partial[y^{\wedge}(k+1)]}{\partial u(k+N_u)} \\ \vdots & \ddots & \vdots \\ \frac{\partial[y^{\wedge}(k+N_2)]}{\partial u(k+1)} & \dots & \frac{\partial[y^{\wedge}(k+N_2)]}{\partial u(k+N_u)} \end{pmatrix} \quad (7)$$

According to Newton-Raphson CFM algorithm [8]

$$\Delta U(k) = -\eta * [H]^{-1} * \frac{\partial J(k)}{\partial U(k)} = -\eta * H_{inv} * \frac{\partial J(k)}{\partial U(k)} \quad (8)$$

Where:

- $\partial J/\partial U$: is the cost function partial differential with respect to control vector denoted by Jacobian and has size of $[N_u \times 1]$.
- H : is the cost function second partial differential with respect to control vector denoted by Hessian. The Hessian matrix is always symmetric and has size of $[N_u \times N_u]$.
- H_{inv} : is the inverse Hessian and also symmetric.
- η : Cost function line search optimization step parameter.

The cost function Jacobian can be expressed as:

$$\frac{\partial J(k)}{\partial U(k)} = -2 * \left[\frac{\partial Y(k)}{\partial U(k)} \right]^T * E(k) + 2 * \lambda * \frac{\partial \Delta U(k)}{\partial U(k)} * \Delta U(k) \quad (9)$$

$$\frac{\partial \Delta U(k)}{\partial U(k)} = \begin{pmatrix} \frac{\partial[u(k+1)-u(k)]}{\partial u(k+1)} & \dots & \frac{\partial[u(k+N_u)-u(k+N_u-1)]}{\partial u(k+1)} \\ \vdots & \ddots & \vdots \\ \frac{\partial[u(k+1)-u(k)]}{\partial u(k+N_u)} & \dots & \frac{\partial[u(k+N_u)-u(k+N_u-1)]}{\partial u(k+N_u)} \end{pmatrix} \quad (10)$$

As $\frac{\partial \Delta u(k+j)}{\partial u(k+h)} = \delta(h, j) - \delta(h, j-1)$, Then

$$\frac{\partial J(k)}{\partial U(k)} = -2 * G^T(k) * E(k) + 2 * \lambda * A * \Delta U(k) \quad (11)$$

A : is constant square matrix ($N_u \times N_u$) independent of sample time k .

Substituting Jacobian above in (8):

$$\Delta U(k) = 2\eta * H_{inv} * G^T(k) * E(k) - 2\eta\lambda * H_{inv} * A * \Delta U(k) \quad (12)$$

$$\Delta U(k) = 2\eta * V * H_{inv} * G^T(k) * E(k) \quad (13)$$

Where $V = [I + 2\eta\lambda * H_{inv} * A]^{-1}$ is square non-symmetric matrix with size $[N_u \times N_u]$.

Substituting $\Delta U(k)$ in (6):

$$\Delta E(k) = -2\eta * G(k) * V * H_{inv} * G^T(k) * E(k) \quad (14)$$

$$\Delta E^T(k) = -2\eta * E^T(k) * G(k) * H^T_{inv} * V^T * G^T(k) \quad (15)$$

Substituting $\Delta E(k)$ & $\Delta E^T(k)$ in (5) we can find:

$$\Delta L = -4\eta * E^T(k) * G(k) * H^T_{inv} * V^T * G^T(k) * E(k) + 4\eta^2 \{ E^T(k) * G(k) * H^T_{inv} * V^T * G^T(k) * G(k) * V * H_{inv} * G^T(k) * E(k) \}$$

$$\Delta L = -4\eta * E^T(k) * [G(k) * H^T_{inv} * V^T * G^T(k) * \{ I - \eta * G(k) * V * H_{inv} * G^T(k) \} * E(k)] \quad (16)$$

From matrix algebra if matrix P is symmetric positive definite then $A^T * P * A$ is positive function for all vectors $A \neq 0$. Also all eigenvalues of matrix P must be positive.

Therefore to obtain $\Delta L \leq 0$ and verify Lyapunov stability the following positive condition should verify:

$$\{ G(k) * H^T_{inv} * V^T * G^T(k) - \eta * G(k) * H^T_{inv} * V^T * G^T(k) * G(k) * V * H_{inv} * G^T(k) \} \geq 0 \quad (17)$$

The difficulty of above proof come from the fact that matrix V is not always symmetric. This is because, although H_{inv} matrix is always symmetric, the matrix $(H_{inv} * A)$ is non-symmetric and so matrix V generally non-symmetric. However there is a fact that in some cases where η is very small, V can be approximated to unity matrix.

Assumption 1: For the simplicity of calculation and not affecting the flow of derivation, when η is very small $\Delta U(k)$ is very small and $\{A * \Delta U(k)\} \approx \Delta U(k)$. Then

$$V \approx H_{inv} \approx I_{N_u \times N_u} \quad (18)$$

And Lyapunov equality in (17) can be simplified to:

$$\{ G(k) * G^T(k) - \eta * G(k) * G^T(k) * G(k) * G^T(k) \} \geq 0 \quad (19)$$

As left hand side of equality in (19) is always symmetric matrix, therefore to achieve positive definiteness of this matrix, the eigenvalues must be all positive.

$$\{ \text{eig}(G(k) * G^T(k)) - \eta * \text{eig}(G(k) * G^T(k) * G(k) * G^T(k)) \} \geq 0 \quad (20)$$

Where $\text{eig}(A)$ is eigenvalues of matrix A . From matrix algebra this requires:

$$\eta < \frac{1}{\sigma_{\max}^2}, \forall k > k_0 \quad (21)$$

Where $\sigma_{\max}$ is the maximum singular value decomposition of non square controller gain matrix $G(k)$.

The above approximation is the same as (Chi-Huang Lu) obtained in normal gradient descent algorithm using recurrent neural identification network model [14].

Therefore in case of validity of approximation made in (18), the second Lyapunov stability condition ($\Delta L \leq 0$) is valid if and only if NGPC optimum line search step “ η ” at each sample time verifies (21). This means to get Asymptotic Stability (AS) or Stable In Sense of Lyapunov (SISL) at any sample time (k), maximum singular value of controller gain matrix should verify:

$$\sigma_{\max}(k) < \frac{1}{\sqrt{\eta(k)}} \quad (22)$$

In case we add constraint $E(k) > 0$; This leads to the same Lyapunov theorem in [15] for Uniform Ultimately Bounded (UUB) Stability.

The mathematic challenge is raised when η is not very small and hence approximation made is no longer valid and sophisticated calculation of condition mentioned in (17) is required at each sample time to verify stability.

IV. SIMULATION RESULTS

In this section, we will show the advantage of using NGPC (with a nonlinear system identifier) over traditional GPC which utilizes a linear system identifier by showing the closed loop control results of both controllers against the same process.

The process model we will use in the simulations consists of a nonlinear non minimum phase system with variable dead time described by:

$$\ddot{y}(t) + \dot{y}(t) + y(t) + y^3(t) = 2u(t) - \dot{u}(t) \quad (23)$$

The controller objective is to stabilize this system over a wide range of reference step input given as:

$$y_d(t) = \begin{cases} 0.5, & t < 75 \\ 3, & t \geq 75 \end{cases} \quad (24)$$

A. Linear Predictor GPC Simulation

Traditional GPC algorithm requires an initial linear model of the process after which the algorithm will utilize a linear predictor to update this linear model. By using extended recursive least squares method on the system output, choosing suitable polynomial order, applying the same Pseudo Random Binary Sequence (PRBS) to the obtained model and finally verifying the obtained model using the un-

correlation test one can obtain the following results (sampling frequency = 100 Hz):

$$G_{process}(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})} \quad (25)$$

$$A(q^{-1}) = 1 - 1.884 q^{-1} + 1.381 q^{-2} - 0.3938 q^{-3} + 0.02195 q^{-4} - 0.02646 q^{-5}$$

$$B(q^{-1}) = 0.01024 q^{-1} \quad (26)$$

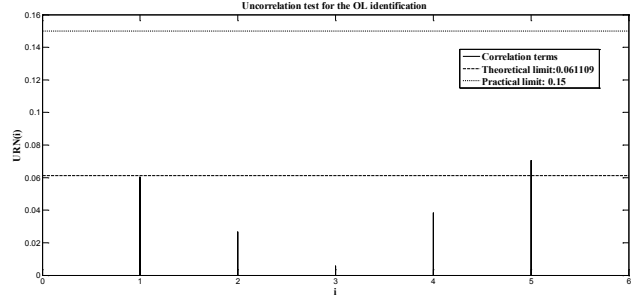


Fig. 3: Ordinary GPC un-correlation test for OL identification.

Fig. 3 shows the quality of the linearized model by applying the Un-correlation test on the output obtained from the actual system and the linearized model versus the theoretical and practical limits described in [17].

The linearized model is then utilized to design a traditional GPC controller. Traditional GPC suffers from obtaining correct prediction matrices due to the variable delay nature of this process. Accordingly, for a wide variety of prediction and control horizons; GPC is unable to stabilize system's output. Fig. 4 shows the response of the closed loop system against the desired input.

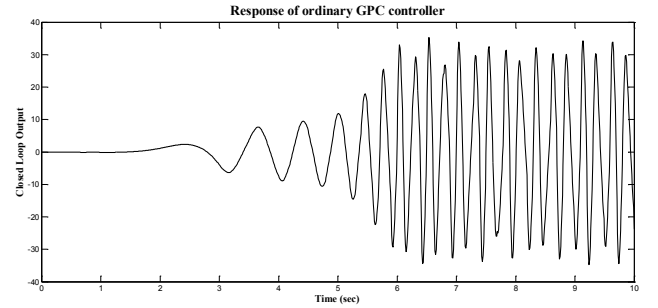


Fig. 4: Ordinary GPC closed loop response.

B. Neural GPC Simulation

In the simulation below, MATALB neural network and control toolboxes were used [16]. Fig. 5 shows the open loop response of the nonlinear process in (23) under step input of (24). It is worthy to show that this plant suffer from variable dead time due to non-symmetric behavior in Fig. 6. The NGPC neural network of Fig. 2 was trained offline on nonlinear plant using Levenberg-Marquardt algorithm. The LM offline training results are described in Fig. 7. In which optimum tuning parameters have been chosen as per Table I to obtain minimum neural network modeling error (ϵ) $\leq 1.5e-03$. The input training data for neural network is band limited white input (Power=100 and correlation time = 10 s) covers practical control input range (-6 to +8).

As seen in Fig. 8 the closed loop NGPC enhanced the tracking even if the reference is to some extent outside NN training range.

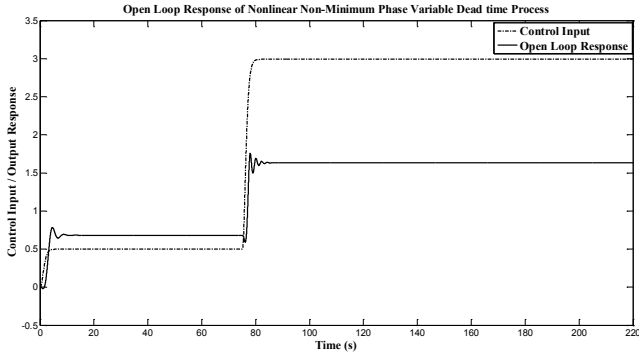


Fig. 5: Open loop response for nonlinear system under step input.

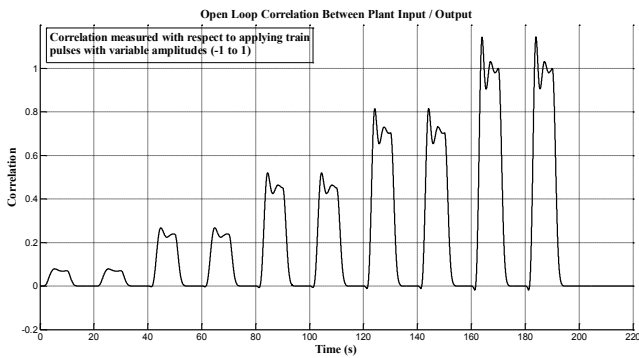


Fig. 6: Open Loop Correlation between Process Input/Output.

Fig. 8 shows that NGPC is able to remove open loop bad dynamics of non-minimum phase nonlinear process. Also in Fig. 9, for the set point (0.5) the NGPC control input is (0.3) whereas for set point (3) the control input is (14.7) outside training ranges.

Table I: Neural GPC Simulation Parameters.

NGPC Parameters	Simulation value
Sampling Time (T_s)	0.1s
Prediction Horizon (N_2)	40
Control Horizon (N_u)	5
Control Weighing Factor (λ)	0.3
Search Parameter (α)	0.1
Iterations per sample	2
Line search Parameter (η)	$0 < \eta \leq 1$ (to obtain stability)
Hidden Layer Neurons (L)	9
Delayed Plant Output (N_j)	4
Delayed Control Input (N_i)	4
Control Input Allowed	$-15 \leq u(t) \leq +15$
Training samples	4,000
Training Epochs	1,000
Minimization Routine	Newton-Raphson
Training Algorithm	Levenberg-Marquardt
NN Modeling Error (ϵ)	$\leq 1.5e-003$

Table II: NGPC stability performance.

NGPC Stability Performance for Non-Linear Process under study					
	Step Ref	Line search step (η)	Average $1/\sigma^2_{max}$	NGPC Stability & Performance Results	
Applying variable Step Reference	Ref (0.5)	$0.0091 \leq \eta \leq 0.0126$	0.02488	negative ΔL tends to zero	As $V \approx H_{inv} \approx 1$, therefore to get stability $\eta < 1/\sigma^2_{max}$ should verify
	Ref (3)	1	Not required as approx. is no longer valid and no direct relation between η & σ^2_{max}	negative ΔL tends to zero	As $V \neq H_{inv} \neq 1$, therefore to get stability ΔL should be negative.

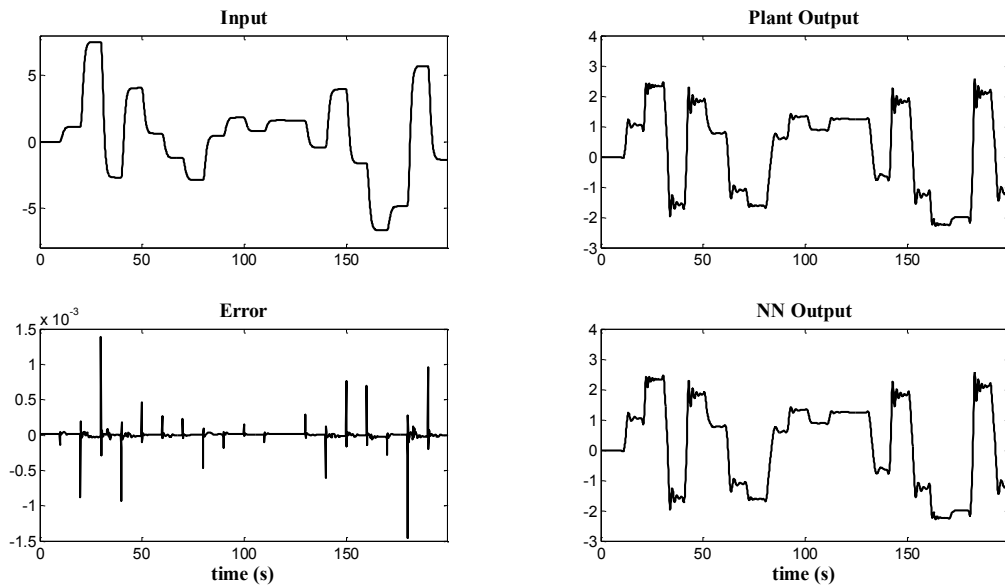


Fig. 7: NGPC neural identification using Levenberg-Marquardt training algorithm.

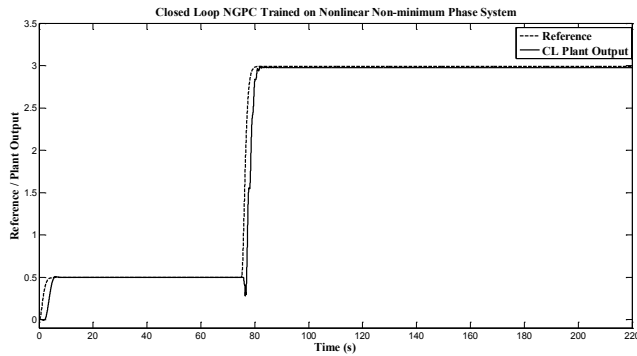


Fig. 8: Neural GPC closed loop response.

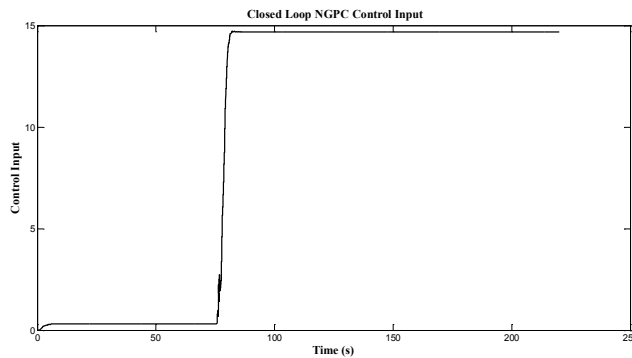


Fig. 9: Neural GPC control input.

In Table II, paper proved the relation between stability criteria in (17) and NGPC closed loop response. When the reference is (0.5), the approximation of (18) is valid and there is a mandatory Lyapunov stability condition as in (21). However when reference is (3), the approximation is no longer valid and sophisticated calculation of (17) is required.

V. CONCLUSION

This paper proved the high quality of stability characteristics when a neural network identifier is utilized with GPC in highly nonlinear systems. It is proved mathematically that NGPC satisfied Lyapunov stability conditions if proper CFM algorithm is used, well offline training is performed and line search parameter verifies derived conditions. Simulation results show the quality of the NGPC controller over the traditional GPC algorithm in terms of meeting the required closed-loop performance while stabilizing the system. The next work shall focus on integrating Internal Model Principle technique with NGPC to enhance both tracking and disturbance rejection performance of NGPC controlling severe nonlinear systems.

REFERENCES

- [1] R K Al Seyab; Y Cao: "Nonlinear System Identification for Predictive Control using Continuous Time Recurrent Neural Networks and Automatic Differentiation", Journal of Process Control, Volume 18, Issue 6, July 2008, Pages 568-581.
- [2] Dennis and Schnabel, "Numerical Methods for Unconstrained Optimization and Nonlinear Equations", 1983.

- [3] D. W. Clarke, C. Mohtadi and P. S. Tuffs: "Generalized predictive control-Parts I & II. The basic algorithm", Automatica, vol 23, pp. 137-163. 1987.
- [4] David T. Westwick and Robert E. Kearney: "Identification of Nonlinear Physiological Systems", Institute of Electrical and Electronics Engineers, 2003, chapters 6, 7 & 8.
- [5] K. S. Narendra, A. U. Levin: "Identification and Control of Dynamical Systems Using Neural Networks", IEEE Transaction on Neural Networks 1, No. 1, pp. 4-27, 1990.
- [6] Anna Vasičkaninová, Monika Bakošová: "Neural Network Predictive Control of a Chemical Reactor", Institute of Information Engineering, Automation and Mathematics, Faculty of Chemical and Food Technology, Acta Chimica Slovaca, Vol.2, No.2, 2009, pp 21 – 36.
- [7] Sadhana CHIDRAWAR, Balasaheb PATRE: "Generalized Predictive Control and Neural Generalized Predictive Control", Leonardo Journal of Sciences, Issue 13, July-December 2008, Pages 133-152.
- [8] Donald Soloway & Pamela J. Haley: "Neural generalized predictive control: A Newton-Raphson Implementation", NASA Technical Memorandum 110244, Feb 1997.
- [9] B. Durmuş, H. Temurtaş, N. Yumuşak, F. Temurtaş, R. Kazan: "The Cost Function Minimization for Predictive Control by Newton-Raphson Method", Proceedings of IMECS 2008, Vol. II.
- [10] S. Chen, S. Billings, P. Grant: "Non-linear System Identification Using Neural Networks", International Journal of Control 51, No. 6, pp. 1191-1214, 1990.
- [11] D.N.Rao, M.R.K.Murthy, S.R.M.Rao, D.N.Harshal: "Comparison of NGPC With Approximate and Nonlinear Predictive Control - A Simulation Study", ACSE Journal, Volume (6), Issue (1), January, 2006.
- [12] Michalska, H. Mayne, D.Q., "Robust Receding Horizon Control of Constrained Nonlinear Systems", IEEE Transactions on automatic control, No. 11, 38, 1993.
- [13] Jim Benjamin Luther, "Stability of a Neural Predictive Controller Scheme on a Neural Model", Department of automation (IAU), Technical university of Denmark (DTU), 1999.
- [14] Chi-Huang Lu, "Discrete Time Neural Predictive Controller Design", HSIUPING JOURNAL. Volume (18), pp 27-38, March 2009.
- [15] Jagannathan Sarangapani: "Neural Network Control of Nonlinear Discrete-Time Systems", University of Missouri, Rolla, Missouri, chapters 1, 3, 4 & 7, 2006.
- [16] MathWorks, "MATLAB 7.8 User Guide", Neural Network Toolbox, Neural Predictive Controller, Release R2009b.
- [17] Ioan D. Landau and Gianluca Zito, "Digital Control Systems; Design, Identification and Implementation", 1st. Edition, Springer Verlag, 2006.

Improving Web Clustering through a New Modeling for Web Documents

Ioan Agavriloaei¹, Adrian Alexandrescu² and Mitică Craus³

Abstract — The constant and rapid growth of the Web complexity and the Web size generates new challenges regarding the approaches in efficient processing of Web searched results. Due to the dynamic Web content and the huge amount of information returned by search engines, it is necessary to find new methods and ways for better organizing and modelling the information spread on the Web. In this paper, we propose the structuring of the Web content as a hierarchical environment, taking into account the site content and structure, the HTML document structure and the term importance. Furthermore, we propose an effective partitional clustering algorithm for a Web site. The preliminary results prove the effectiveness of the new Web content representation and the accuracy of the Web clustering algorithm.

Keywords — World Wide Web, Web document model, Web clustering, data mining, information retrieval.

I. INTRODUCTION

By exploring the Internet, each user can benefit from a large quantity of information available online. This information is provided in several standard formats, such as HTML, XML and XHTML for text pages and jpeg or tif for graphics and images. In addition, the multimedia information (audio and/or video) has an extraordinary growth on the Internet and requires new technologies for accessing and processing it.

According to Google [10], the size of World Wide Web (WWW) is infinite and this boom has led us to a new era of information, creating new opportunities like e-business, e-commerce, e-marketing, e-finance, e-learning and many social-networks. All efforts that have been made since the birth of Hyper Text Transfer Protocol (HTTP) and of Hyper Text Markup Language (HTML) make the Internet accessible for anyone who has a computer or a smart device. The Internet tends to take the place of the traditional libraries where the people could find the desired information accumulated in a single place. By means of the actual search environments, the information spread over the Internet is accessible such as the information in a library is available locally.

During time, the number of Web documents has increased, especially HTML pages. This growth has also led to necessity to form up the information and to build more structured containers, in this way allowing more thorough and personalized searches, directly correlated to the semantics of the document itself. Finally, the main effect of the Internet explosion in information processing was that the developers needed to create and provide more sophisticated search-tools systems and more computing power to continually process what is new and old on the Web.

Since the Web has become a global library, in order to represent such astronomical amount of interconnected documents, new or adapted Information Retrieval (IR) techniques have been proposed. These techniques were called Layout Representation [1]. In classic IR, the data needs to pass through a stage of pre-processing and a stage of refinement, which consists of four intermediate stages: data selection, filtering, cleaning and representation. While certain researchers have come with some proposals from classic IR [2, 3], others have researched for new strategies relying on the Web's hypertext features, such as page hyperlinks and HTML general tags [4, 5, 6]. The result was the birth of Web Information Retrieval (Web-IR) that was used to retrieve documents from the Web [7, 8].

In this paper we present a Web document model, based on tags and links, and a hierarchical structure of the Web. We propose a representation model of the Web document taking into account the HTML structure of the document and the fact that the Web pages are already grouped into sites over the WWW. Furthermore, for the document representation we are going to take into consideration the latest trends in Search Engine Optimization (SEO) to improve the IR accuracy and efficiency. With our model, we intend to build a representation of the Web documents, useful for all applications that are aimed at processing information automatically. Based on this new document representation and the hierarchical Web structure, a partitional site clustering algorithm is proposed.

This paper is organized as follows. In section 2 it is described our model of Web document representation. The partitional clustering algorithm based on our page representation model is proposed in section 3. Some primary evaluations of the proposed algorithm are given in section 4. In section 5, some related work and discussions are presented. Finally, section 6 gives the conclusions for this work and further research directions.

1. I. Agavriloaei is with Faculty of Automatic Control and Computer Engineering, "Gheorghe Asachi" Technical University of Iasi, Romania, ioan.agavriloaei@gmail.com
2. A. Alexandrescu is with Faculty of Automatic Control and Computer Engineering, "Gheorghe Asachi" Technical University of Iasi, Romania, avalexandrescu@gmail.com
3. M. Craus is with Faculty of Automatic Control and Computer Engineering, "Gheorghe Asachi" Technical University of Iasi, Romania craus@tuiasi.ro

II. TERM WEIGHTING AND DOCUMENT MODELLING

Consider that the Web documents are already grouped in sites and taking into account the HTML structure of the Web document, we present a new approach about Web content modelling. Each Web document from collection site (S) is represented by two vectors: a vector of terms weight obtained from its HTML content and a vector of links obtained from its hyperlinks. A vector of terms weight is used to build the *Index Term Database (ITD)* of the collection S . The list of links from each document is used to build a hierarchy structure among the documents within the specific obtained cluster from *ITD* matrix. The *ITD* matrix is the source content for the clusters. After modelling the Web content in such a manner, a Web site collection is represented as a group of structured clusters.

A. Term weighting

The weight of one term must ensure the needed discerning power for a correct representation and retrieval of documents containing it [3].

An appropriate term weighting can considerably improve the performance of the Vector Space Model (VSM) [20]. A weighting scheme is composed of three different types of term weighting: local weight, global weight and normalization factor. The complete term weight definition is given by:

$$w_t = w_{l_{td}} * w_{g_t} * w_{n_d} \quad (1)$$

where, $w_{l_{td}}$ is the local weight for the term t in the document d , w_{g_t} is the global weight for the term t in the collection, and w_{n_d} is the normalization factor.

In almost all IR techniques the local weight depends on how many times each term t appears in a document d and it is evaluated according to the terms in the given document. In this respect, the local weight $w_{l_{td}}$ for a term t is a function of the frequency of term t in document d :

$$w_{l_{td}} = f(t_{fd}) \quad (2)$$

Taking into consideration Cutler's approach [20], we can define a new local weight model based on his results while considering the latest trends in SEO to improve the IR. In his approach, Cutler represents a Web document by six tag classes: (*Plain Texts*, *Strong*s, *Lists*, *Headings*, *Anchor*s, *Titles*). The local term weight is defined as an inner product of the two vectors: *Class Importance Vector (CIV)* and *Term Frequency Vector (TFV)*. In his experimental results, Cutler found three different *CIV*s: (1, 8, 1, 8, 8, 2), (1, 8, 1, 7, 8, 2) and (1, 8, 1, 5, 8, 2). Consider that some tag classes have the same importance factor, we can represent a Web document only by a vector of four tag classes: (*Titles*, *Headings*, *Strong*s-*Anchor*s, *Plain Text*s). For *Headings* we take the average value from Cutler's results. The *CIV* vector will be (1, 8, 7, 2) (see Table I).

These four classes describe a Web document as follow:

- 1) *Titles*: terms given by `<title>` tag of the document.

TABLE I
THE CLASSES AND ASSOCIATED TAGS

Class Name	HTML tags or tag attribute	Importance factor
Title	<code><title></code>	1
Header	<code><h1, h2, h3, h4, h5, h6></code>	8
Strong	<code><strong, b, em, i, u, a, a:href, a:alt></code>	7
Plain Texts	None of the above	2

- 2) *Headings*: terms provided by headers that describe the main structure and subjects of a page.
- 3) *Strong*s-*Anchor*s: contains all terms that occurs in the tags `<strong>`, `<b>`, `<em>`, `<i>`, `<u>` and `<a>`. Here we will add some additional tag attributes from tag `<a>`, like `alt` and `href` attributes to increase the importance of these terms.
- 4) *Plain Text*s: includes the remaining terms.

To represent correctly a Web document, we need to parse each nonstop word of a document and stemm it, then we assign each stemmed word to one of this four classes and increment the class count. Now we can define the local term weight as follow:

$$w_{l_{td}} = CIV * TFV \quad (3)$$

where, $CIV = (1, 8, 7, 2)$ contains the weight of each class and $TFV = \{t_{fd1}, t_{fd2}, t_{fd3}, t_{fd4}\}$ contains the occurrence frequency of the term t in the document d for each tag class.

The global weight depends on how many times each term t appears in the entire collection S and tries to give a "discrimination value" to each term. Many global weights definitions are based on the idea that the less frequently a term appears in the whole collection, the more discriminating it is [8]. The most used global weight is the inverted document frequency (*idf*):

$$idf = \log \left(\frac{|D|}{df_{td}} \right) \quad (4)$$

where, $|D|$ is the cardinality of collection D and df_{td} is the number of documents from D where the t -th term occurs. In our case, the global weight is defined on the site collection S , so the global weight will be:

$$w_{g_t} = idf_S = \log \left(\frac{|S|}{df_{tS}} \right) \quad (5)$$

The normalization factor is a correction variable that must compensate the discrepancies among the documents lengths. It is calculated as soon as the local and global weighting is achieved. The idea is to normalize the document term vector so that documents are retrieved independent of their lengths. In this paper we use the cosine normalization factor, defined in (6), and the pivoted document length normalization factor, defined in (7), with slope 0.2 as a normalization factor, also called *Lnu* weighting in *Smart* [2].

$$wn_d = \frac{1}{\sqrt{\sum_{i=1}^m w_i^2}} \quad (6)$$

where, w_i is the rough weight ($w_{l_{td}} * w_{g_t}$) for the i -th term and m is the total number of terms in collection.

$$wn_d = \frac{\frac{1+\log(tf_{td})}{1+\log(\text{average } tf_{td})}}{(1.0-\text{slope})*\text{pivot}+\text{slope}*\# \text{ of unique terms}} \quad (7)$$

where, *pivot* is the average number of unique terms in a document (computed across the entire collection) and *slope* is a constant determined experimentally [16].

B. Document modelling

Considering the above term weight definition and the relationships among Web documents provided by their specific structure, we can create an abstract representation of the Web document, useful for all applications that aim at processing information automatically.

For a Web document d_j from collection S the model can be defined by an ensemble of two vectors:

$$d_j = \{(TWV_j, DLV_j), j = 1, \dots, n\} \quad (8)$$

where, n is the cardinality of S , $TWV_j = \{w_{ij}, i = 1, \dots, m\}$ represents the terms weight vector for j -th document that has m terms and, $DLV_j = \{l_k, k = 1, \dots, p\}$ is the vector that contains all links from document d_j .

Each link from a Web document aims to give semantics between two documents; it is created in order to be relevant to the related documents. This connection, once being revealed, could be used to find higher-level relationship among the Web documents, in our case a relationship within the cluster's documents.

For each document that belongs to collection S , all terms are extracted and the *ITD* is built: it consists of all index terms of all the documents belonging to the collection S . By creating the *ITD* matrix each document is represented as a vector of real numbers in an m -dimensional space, with $m = |ITD|$. Since the m dimension could be very high, even several thousands, a dimension reduction process over *ITD* matrix must be done, to avoid computational problems in computing the VSM. This process is performed ignoring the terms that are in the stopwords list, or the terms that have at least a required number of characters, or have a required term frequency in the Web document, or an inverted document frequency in the collection. Also, the dimension is further reduced after applying the stemming phase over the remaining terms.

III. SITE CLUSTERING ALGORITHM

The key for implementing an efficient web clustering algorithm is to find intrinsic relationships, such as similarities and dissimilarities among the pages [11].

With the above page model and the site group idea, a Web clustering algorithm can be established. Our clustering

algorithm consists of two phases. The first one is single site clustering, in which the all terms from site collection S , respectively the matrix *ITD*, are clustered at the same level without hierarchy. At this point, we will have the site represented as a collection of clusters, each cluster being composed by a group of terms. The second step phrase consists of a topological sort of all Web documents in a specific cluster taking into consideration the *in-site-links* list of each cluster's document. In Figure 1 it is depicted a Web clustering diagram. The role and the importance of each document in the cluster can be reflected by the document connections within the same cluster's documents or with the documents from others clusters or more, with the documents from the whole Internet. Each document is most likely to reflect certain topics associated to the query within the concerned document. If two documents are linked by or linked to some documents from the same cluster, these two documents will more likely be located in the same cluster and have a higher position in the cluster document hierarchy.

Initially, we cluster the site by its *ITD* matrix; therefore we can observe that each cluster can also be represented as a group of documents that contains all the terms from the corresponding cluster. This approach has an important advantage, namely that it can produce some overlapping clusters (a single document can be contained in many clusters). Based on the above page model and observations, the details of the site clustering algorithm are described as follows:

Phase 1: Single Site Clustering

[Input]: An *ITD* matrix with all site terms, $ITD = \{t_1, t_2, \dots, t_m\}$, parameter l - the number of clusters.

[Output]: A set of base clusters $CL = \{CL_j, j = 1, \dots, l\}$.

[Algorithm]: *SiteClustering*(*ITD*, l)

Step 1: Randomly select the initial centroid positions for the given number of clusters, $CE_j, j = 1, \dots, l$.

Step 2: For each term $t_i \in ITD$, calculate the distance between t_i and the centroid of each existing cluster, $dist(t_i, CE_j)$.

Step 3: If CL_k is the current cluster and if $dist(t_i, CE_k) = \min_{1 \leq j \leq l} (dist(t_i, CE_j))$, then add t_i to the cluster CL_k and recalculate the centroid CE_k of this cluster:

$$CE_k = \frac{1}{|CL_k|} \sum_{j \in CL_k} t_j,$$

where $|CL_k|$ is the number of terms in cluster CL_k .

Otherwise, t_i itself initiates a new cluster and it is the centroid of this new cluster.

Step 4: If there are still terms to be clustered (e.g. terms that have not been clustered or a term that is a cluster by itself) or the stop condition is not true, go back to step 2 until all clusters centroids no longer change or stop condition is satisfied.

Step 5: Return the set of base clusters

$$CL = \{CL_j, j = 1, \dots, l\},$$

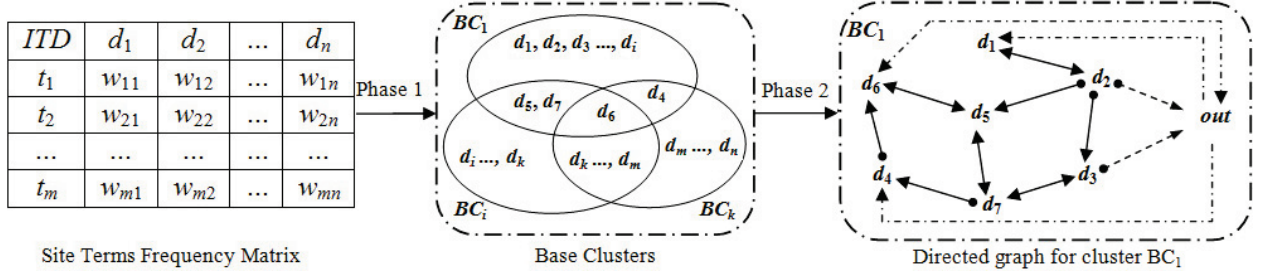


Fig. 1. Web Clustering Diagram

each base cluster CL_j being composed of all documents that contain the cluster's terms.

The first phase of the clustering algorithm produces a set of single site clusters called site base clusters. Taking into account the list of *in-site-links* for each cluster's document and that a document is more important in the cluster if it is highest-linked with the cluster's and the site's documents, each site base cluster will be represented as a hierarchical document structure. The procedure is described as the phase 2 of the clustering algorithm.

Phase 2: Cluster structuring

[**Input**]: A set of base clusters, $CL = \{CL_j, j = 1, \dots, l\}$.

[**Output**]: A set of structured clusters

$$SCL = \{SCL_j, j = 1, \dots, l\}.$$

[**Algorithm**]: *ClusterStructuring* (CL)

Step 1: For each base cluster $CL_j \in CL$ build the directed *in-site-links* graph $DG_j = (D_j, A_j)$ where the vertices D_j are the documents belonging to the cluster CL_j and the directed edges A_j are anchors from and to the documents of cluster CL_j . A link from a document $d \in D_j$ to a document that is not belonging to the cluster is represented by an arc (d, out) and a link to a document $d \in D_j$ from a document that is not belonging to the cluster CL_j is represented by an arc (out, d) . The node *out* is a special vertex in the digraph DG_j representing all the documents outside the cluster CL_j .

Step 2: If DG_j has at least one cycle, then remove a minimum number of edges, so that DG_j is acyclic again and go to step 3. Otherwise go to step 3.

Step 3: Do the topological sort of the documents in the digraph DG_j .

Step 4: Return the structured clusters

$$SCL = \{SCL_j, j = 1, \dots, l\}$$

where, SCL_j is defined as the topological sorted list of the documents that belong to the cluster CL_j .

The challenge of the second phase is to remove a number of edges from digraph $DG_j, j = 1, \dots, l$, which "contradict" the property of being acyclic. Intuitively, this set of edges should be minimal. Once the cycle is broken, we can sort the vertices by a criterion like "least number of *in-cluster* edges first". When we have two vertices with the same number of *in-cluster* edges, we define another order

criterion, like "least number of *out-cluster* edges first".

Since the page normalization is guaranteed in the term weighting phase, as in (6) and (7), we can observe that the above clustering algorithm is independent from the document order. It is not difficult to prove that the complexity of this algorithm is $O(l \cdot p \cdot \log p) + O(m + n)$, where l is the number of given clusters, p is the number of documents to be clustered, n is the number of vertices and m is the number of edges (arcs) from all $DG_j, j = 1, \dots, l$.

IV. EVALUATIONS

Primary clustering experiments were executed on two sets of data. In order to better understand the algorithm's phases and to manually process and sort the generated oriented graph from the second phase of our algorithm, the first set of documents was a small site (S_1) with a collection of 23 pages that has 867 different terms and 64 hyperlinks. The second data set was a big site (S_2). The total number of documents in the site was 3.240, and the number of hyperlinks in the document source was 14.291.

We named the clustering algorithm with cosine document normalization as $SCLA(C)$, and that with pivoted document length normalization as $SCLA(Lnu)$.

It is not a difficult task to measure the efficiency and the effectiveness of this kind of clustering algorithm. In this paper, we used two performance measurements: *recall* and *precision* functions to measure the accuracy of our method. These two functions can be applied on the resulted clusters from the first phase of the algorithm since the second phase does not affect the accuracy of the results. To reflect how the retrieved documents are relevant to the user's information need, we adapted the precision and the recall concept in IR [13]. Given a Web document, we note its real cluster as CLR_i and its obtained cluster as CLE_j . For an obtained cluster CLE_j we defined the precision and the recall function as follows:

$$Precision(CLE_j) = \frac{\max_{1 \leq i \leq l} (|CLE_j \cap CLR_i|)}{|CLE_j|}$$

$$Recall(CLE_j) = \frac{\max_{1 \leq i \leq l} (|CLE_j \cap CLR_i|)}{|CLR_i|}$$

where, $|CLE_j|$ is the number of documents in cluster CLE_j and $|CLR_i|$ is the number of documents in cluster CLR_i . As

a global functions, this two equations can be defined as the average values of each cluster values from collection S .

In our experiment for the site S_1 , we manually checked each term to be clustered and we decided manually upon the correct clusters according to our judgment. At this stage of the algorithm comparison it was reasonable to use it as a reference model or standard since we tried our best to objectively cluster the Web documents. The further experiments will be conducted in our future plans. For the first step of our algorithm, the number of clusters (parameter l) was given by the equation: $l \approx \sqrt{n/2}$, where n

is the total number of documents in the collection. Given the sets of clusters produced by the above $SCLA(C)$ and $SCLA(Lnu)$ algorithms over collection S_1 , we conducted an accuracy comparison. This evaluation was made after the first phase of the algorithm and it is represented as the average value of the accuracy in the base clusters. Site S_1 has 23 documents, so according to the previous equation, the number of clusters is $l = 4$. The result of this comparison is shown in Fig. 2. In the last four columns it is shown that the $SCLA(Lnu)$ algorithm has higher cluster accuracy than the $SCLA(C)$ algorithm. The below evaluation results prove the effectiveness of the new page model and the corresponding clustering algorithm. Results with the same accuracy were obtained for collection S_2 .

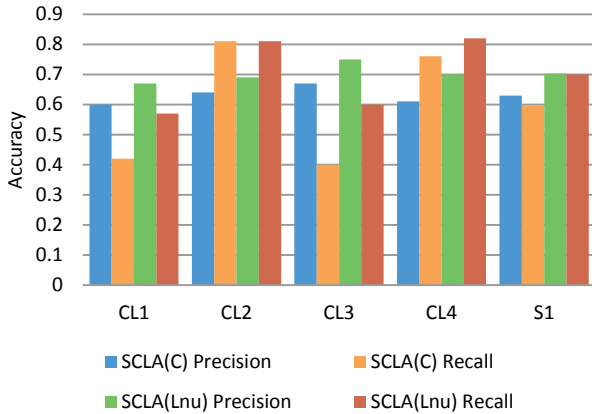


Fig. 2. The Base Clusters Accuracy for S_1 .

After the first phase of the algorithm the documents in the same cluster represent the same area of information and they have the same importance factor for the user's topic. Therefore, in the second phase, we highlighted the most important documents within a cluster by building a directed graph and by performing a topological sort over its documents taking into account the hyperlinks among the site's documents. The directed graph for the first base cluster from S_1 is illustrated in Fig. 3.

The numbers assigned to edges represent the order in which the edges are removed when we sort the documents. The less important documents are first in the list. As we can

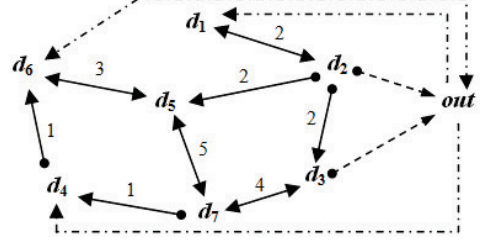


Fig. 3. The Directed Graph of cluster CL_1 from S_1 .

see, the digraph has a cycle represented by $\langle d_4, d_5, d_6, d_7 \rangle$ with the *in-degree* $\langle 2, 3, 3, 2 \rangle$. The topological sort cannot be applied on a digraph with cycles therefore we need to remove a minimum number of edges from the digraph so it will be acyclic again. To break the cycle we consider that the edges which are connected to the *out* vertex have less importance. Because the vertex d_4 has an edge from the *out* vertex, this vertex is less important than d_7 and, it is chosen to break the cycle by deleting the edge that comes from d_7 (e_{74}). Thus, the sorted list of documents in cluster is: $\langle d_4, d_2, d_1, d_6, d_3, d_7, d_5 \rangle$.

With the above sorted list we can rapidly offer the most relevant documents for the user's topic without an additional search among the cluster's documents. From the second phase of our algorithm, we could conclude that the right choice of the vertex that breaks the cycles is very important for the quality of final ranking. The clustering results are satisfactory as the documents in the same cluster have a well established hierarchy.

V. RELATED WORK AND DISCUSSIONS

Along with the progress in Web technologies and the development of the HTML language used to write the majority of Web documents, other document representations have been suggested; they are not only based on the single terms that made them up, but also on other typical features of the hypertext environment, expressed through hyperlinks and HTML tags. To model and to represent a Web document some research projects have taken into account the structure of a Web document to obtain a better representation. Other research projects have developed their own systems for IR.

Many proposals in literature have tried to improve the performance of IR. The idea consists in modifying the document pre-processing process, taking into account the HTML structure of the Web document as well. Some models based on HTML tags are found in [12, 13, 14, 21].

In IRIS System [14], Yang and Maglaughlin implement a Vector Space representation of a document where the term weight is calculated according to the tags within certain borders. The IRIS System uses these parts as an index term source. In the term weighting phase the terms are weighted differently, depending on their position within the document and the SMART *lnu* [15] weighting schema with slope 0.3 [16].

In [17, 18] the document is modeled in 12 classes of terms (*Title, Headings, Linking, Lists*, etc) which, together make up the document model after the parsing phase. The document is then represented by a vector $S = \{S_1, S_2, \dots, S_{12}\}$, where $S_i \geq 0$ represents the number of terms in the tag class C_i . The frequency of a term t is represented by a vector $T = \{T_1, T_2, \dots, T_n\}$, where $T_i \geq 0$ is the frequency of term t in the tag class C_i and $n = |ITD|$.

Another tag-based system, WEBOR System [12, 19], was proposed by Cutler, Shih and Meng. After some intermediate experiments and studies [19], a document model is composed only of six tag classes (*Titles, Headings, Lists, Strong, Anchors and Plain Texts*). In the first phase a document is divided in 6 parts, each one formed by the groups of terms in the document which are contained between each class tags. Then the document is modeled with the classic Vector Space Model. In term weighting step a vector $CIV = \{civ_1, civ_2, \dots, civ_6\}$ is composed, where $civ_i \geq 0$ represents a weight determined experimentally by means of genetic algorithm [19]. The experimental results regarding the best CIV s lead to the vectors: (1, 8, 1, 8, 8, 2), (1, 8, 1, 7, 8, 2), and (1, 8, 1, 5, 8, 2) for each tag class. Then, another vector is formed, $TFV = \{tif_1, tif_2, \dots, tif_6\}$, which contains the occurrence frequency of term t in document d for each tag class. The last step is the calculation of weight w_t to be associated with term t by the following formula:

$$w_t = (CIV * TFV) * idf,$$

where, idf is the inverse document frequency of term t in the documents of the collection D . By using these values the experiments indicate that the retrieval effectiveness can be improved with 39.6% or higher. The experiments show that the terms in the Strong and Anchor classes are the most useful for improving general retrieval effectiveness.

VI. CONCLUSIONS AND FUTURE WORK

This paper proposes a new Web page model and a new clustering algorithm applied over a limited collection of Web documents. The new page model is based on HTML tags and on the hyperlinks among the documents within the concerned site, while effectively incorporating terms importance and hyperlink analysis. Our clustering algorithm highlights the useful documents for the user's topic by clustering the weighted terms within the site and by topologically sorting the documents in the cluster.

Our tests showed that, when using the page model with Lnu as the factor of normalization, the clustering accuracy is improved. Furthermore, the document relevance is highlighted by taking into account the links among the documents. The clustering accuracy improvement shown in our experiments demonstrates the effectiveness of the proposed Web document model and the effectiveness of the proposed clustering algorithm as well.

The proposed clustering algorithm is intuitive but it

partially reveals only the semantics among the Web documents. A more careful analysis of the HTML tags position within the Web document and of the hyperlinks among the clusters might be another approach to increase the efficiency of the Web document clustering. Until then, some problems in clustering analysis must be solved, such as how to determine the optimum number of clusters for a site. More experiments need to be performed to demonstrate the feasibility of the proposed ideas. The research into these topics will be done in the near future.

ACKNOWLEDGMENT

This paper was realized with the support of EURODOC "Doctoral Scholarships for research performance at European level" project, financed by the European Social Fund and Romanian Government.

REFERENCES

- [1] P. Brusilovsky, A. Kobsa, and W. Nejdl, "The Adaptive Web, Methods and Strategies of Web Personalization", in Proc. The Adaptive Web, 2007.
- [2] G. Salton, "The Smart Retrieval System. Experiments in Automatic Document Processing", First ed. Prentice Hall, 1971.
- [3] G. Salton and M. McGill, "Introduction to Modern Information Retrieval", 1984.
- [4] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine", Computer Networks, 1998, pp.107-117.
- [5] J.M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment", in Proc. SODA, 1998, pp.668-677.
- [6] A. Molinari, R.A.M. Pereira, and G. Pasi, "An Indexing Model of HTML Documents", in Proc. SAC, 2003, pp.834-840.
- [7] M. Kobayashi, K. Takeda, "Information retrieval on the web", Journal ACM Computing Surveys, June 2000.
- [8] M.O. Nassar, G. Kanaan, H.A.H. Awad, "Comparison between Different Global Weighting Schemes", International Multi Conference of Engineers and Computer Scientists, Hong Kong, 2010.
- [9] R.A. Baeza-Yates and B.A. Ribeiro-Neto, "Modern Information Retrieval", 1999.
- [10] <http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html>.
- [11] J. Hou and Y. Zhang, "Utilizing Hyperlink Transitivity to Improve Web Page Clustering", in Proc. ADC, 2003, pp.49-57.
- [12] M. Cutler, Y. Shih, and W. Meng, "Using the Structure of HTML Documents to Improve Retrieval", in Proc. USENIX Symposium on Internet Technologies and Systems, 1997.
- [13] K. Yang, "Combining Text- and Link-based Retrieval Methods for Web IR", in Proc. TREC, 2001.
- [14] K. Yang and K. Maglaughlin, "IRIS at TREC-8", in Proc. TREC, 1999, pp. 645-656.
- [15] A. Singhal, C. Buckley, and M. Mitra, "Pivoted Document Length Normalization", in Proc. SIGIR, 1996, pp.21-29.
- [16] C. Buckley, A. Singhal, and M. Mitra, "New Retrieval Approaches Using SMART: TREC 4", in Proc. TREC, 1995.
- [17] A. Molinari, R.A.M. Pereira, and G. Pasi, "An Indexing Model of HTML Documents", in Proc. SAC, 2003, pp.834-840.
- [18] R.A.M. Pereira, A. Molinari, and G. Pasi, "Contextual weighted representations and indexing models for the retrieval of HTML documents", presented at Soft Comput., 2005, pp.481-492.
- [19] M. Cutler, H. Deng, and W. Meng, "A New Study on Using HTML Structures to Improve Retrieval", in Proc. ICTAI, 1999, pp.406-409.
- [20] E. Chisholm, and T. G. Kolda, "New term weighting formulas for the Vector Space model in information retrieval", 1999.
- [21] A. Markov, M. Last, and A. Kandel, "The hybrid representation model for web document classification", presented at Int. J. Intell. Syst., 2008, pp.654-679.

A Genetic Algorithm for Mapping Tasks in Heterogeneous Computing Systems

Adrian Alexandrescu, Ioan Agavriiloaei and Mitică Craus

Abstract—Heterogeneous computing systems require an efficient way of distributing tasks across processing nodes. The tasks have to be mapped to the processors which execute them in the shortest time possible, while keeping the processors at a similar load. Tests have shown that, in most cases, the genetic algorithm produces the best solution among all the mapping heuristics.

This paper presents a Genetic Algorithm with a 3-Step Mutation which significantly increases the solution's convergence rate by using a combination of methods to mutate a chromosome. Beside the standard random approach, we implemented a targeted mutation operator which lightens the load of the most occupied processors. We also focused on different fitness functions in order to improve both the makespan and the load balance. The mutation combinations and the fitness functions are then tested to see which ones perform better and in what cases.

Index Terms—Mapping heuristic, genetic algorithm, 3-step mutation, fitness function, heterogeneous computing system

I. INTRODUCTION

WITHIN heterogeneous computing systems, the mapping problem consists of assigning a set of tasks to processors. An acceptable solution to this problem is obtained if the tasks are mapped in such a way that all the processors have more or less the same load while the execution of all the assigned tasks is done in the shortest time possible.

There are two categories of mapping heuristics: the ones which assign one task at a time and the heuristics which map all the tasks at once [1]. The heuristics in the first category, such as Opportunistic Load Balancing, Minimum Execution Time, Minimum Completion Time or K-Percent Best, are characterized by a low algorithm complexity and a short mapping time, however, they also provide poorer results. On the other hand, mapping heuristics such as Min-Min, Max-Min, Simulated Annealing or Genetic Algorithms provide a much better solution, but they usually take longer to run.

Tests have shown that, when compared to other heuristics, the genetic algorithm provides the fastest time in which all the mapped tasks finish their execution, especially if a chromosome in the initial population represents the solution of another mapping algorithm [2].

A. Alexandrescu is with Faculty of Automatic Control and Computer Engineering, "Gheorghe Asachi" Technical University of Iasi, Romania
avalexandrescu@gmail.com

I. Agavriiloaei is with Faculty of Automatic Control and Computer Engineering, "Gheorghe Asachi" Technical University of Iasi, Romania
ioan.agavriiloaei@gmail.com

M. Craus is with Faculty of Automatic Control and Computer Engineering, "Gheorghe Asachi" Technical University of Iasi, Romania
craus@tuiasi.ro

There are also other algorithms that can produce a good solution to the mapping problem such as the particle swarm optimization [3], the Bayesian optimization algorithm [4], an iterated greedy algorithm [5], a swarm intelligence technique for maximizing the system reliability [6] or an algorithm which uses a task clustering method, two metrics and a refinement heuristic [7]. Another approach is in [8] where processors can handle only a limited number of tasks and the proposed solution uses a Hopfield neural network combined with a genetic algorithm and a simulated annealing.

Nonetheless, the genetic algorithm is considered to be a very good mapping algorithm because it obtains a good result in a short time. There are different forms of the genetic algorithm that can be used to obtain a better and a faster solution, from simple approaches to more complex methods. For instance, [9] uses an algorithm for guiding the search process and [10] also considers the data communication between tasks.

A genetic algorithm can generate a better solution by optimizing the evaluation process and by improving the evolution operators: selection, crossover and mutation.

The evaluation process uses a fitness function usually defined by the *makespan* [11], which represents the time it takes for all the processors to execute all the assigned tasks ([2], [12]). In [13] the genetic algorithm uses two fitness functions. The first one tries to minimize the makespan, while the second one focuses on satisfying the load balance between processors and represents the makespan divided by the average execution time over all processors. The goal is to obtain a solution with a low makespan and, if it is possible, to keep the processors well balanced. However, we do not need to use two fitness functions in order to achieve that goal; we can use a simple method of computing the fitness which gives good results by using the makespan and the difference between the time when the first and the last processor finish running all their assigned tasks. The evolution operators are in most cases roulette wheel selection, one-point crossover and random mutation. While there is little to improve in regards to the first operator, for the second and especially the third one certain optimizations can be done. We can speed the solution's convergence by using a targeted crossover and by employing more than one mutation method.

All the mapping heuristics use estimations of the execution time of each task on each processor. These times are kept in an *Expected Time to Compute* (ETC) matrix which includes on the rows, the tasks to be mapped and on the columns, the available processors. Taking into consideration that we have n tasks and m processors, a matrix cell (etc_{ij}) represents the

TABLE I
INCONSISTENT ETC MATRIX

	p_1	p_2	p_3	p_4
t_1	4	5	7	4
t_2	3	2	5	5
t_3	9	15	13	11
t_4	3	12	6	9
t_5	11	12	10	11
t_6	5	6	11	4
t_7	3	5	9	12

execution time of task t_p on processor p_q , where $i = 1, \dots, n$ and $j = 1, \dots, m$. The ETC matrix can be *consistent*, *inconsistent* or *partially-consistent* [2]. A matrix is *consistent* if, for each processor, the execution times of all the tasks are either all lower or all higher than the execution times on another processor. If no such order relationship can be made, then the matrix is *inconsistent* and if the inconsistent matrix includes a consistent submatrix, then the matrix is considered to be *partially-consistent*. For simplicity's sake, we considered only the inconsistent ETC matrix, because it provides a greater degree of processor heterogeneity. In Table I we have a matrix where, for example, tasks t_1, t_2, t_6, t_7 have a lower execution time on processor p_2 than on p_3 , while tasks t_3, t_4, t_5 have a higher execution time. Thus, no order relationship can be made. The values in the ETC matrix are expressed in generic *time units* (t.u.).

II. THE ALGORITHM

Our Genetic Algorithm with 3-Step Mutation (GA3SM) is represented by the following pseudocode:

- 1) Generate the initial population.
- 2) Compute the fitness of each chromosome.
- 3) While the stop condition is not met, apply the operators:
 - a) Selection,
 - b) Crossover,
 - c) Mutation,
 - d) Evaluation.
- 4) Output the best solution.

The population consists of multiple chromosomes and each chromosome is a possible mapping solution. A chromosome is represented by an array of numbers, where the element on the i^{th} position represents the processor to which task i is assigned. This type of chromosome does not take into account the order in which the tasks are scheduled to processors.

A. Initial Population

The *initial population* can be generated in various ways. The easiest one is to use all-random chromosomes, but, in order to speed up the solution's convergence, one or more chromosomes can be initialized with the solution of other mapping algorithm such as Min-Min, Max-Min or Duplex. In [14] 8 heuristics are used to initiate the population, but

the problem with using multiple heuristics is the extra time needed to run those algorithms. In most cases, to obtain a better solution it is enough for only one chromosome to be initialized with the result of the Min-Min heuristic. In order to increase the probability of selecting that chromosome, at least initially, a small percentage of the population can be initialized with the solution provided by the chosen heuristic. To further improve the algorithm, a mutation operator can be applied on those chromosomes before the iterations start.

Our algorithm uses *elitism* therefore the best chromosomes are being kept in the population with each iteration. The best 20% of chromosomes from the old population are copied in the new one.

B. Fitness Function

The *fitness function* of a genetic algorithm is usually considered to be the solution's makespan, but, in doing so, the load across the processors can be unbalanced. Another method involves minimizing the difference between the time in which the first and the last processor should finish running their tasks (Δ). However, this leads to another problem: in most cases, the tasks will take longer to run because they are assigned to processors that do not provide a fair execution time. The obvious solution is to choose a fitness function which considers both the makespan M and the time difference Δ . The selected fitness function is:

$$\frac{1}{aM + b(\Delta + 1)} \quad (1)$$

where a and b are constants between 0 and 1, and both cannot be 0. If we consider that a equals 0 and b equals 1 then the previous fitness function becomes:

$$\frac{1}{\Delta + 1} \quad (2)$$

If all the processors have a perfectly balanced load, the time difference Δ value is equal to 0, therefore the value of the fitness function from (2) is equal to 1. By adding a unit to the time difference Δ we prevent the situation where the fitness value is equal to infinity. If this were the case, then the chromosome with the fitness value of infinity would always have been chosen in the selection phase, thus stopping the search for a better solution. However, because the fitness value cannot be further improved, we can stop the algorithm, even if another solution with the same time difference Δ but with a lower makespan can be obtained.

In Fig. 1 we have three possible solutions to the problem of mapping 7 tasks to 4 processors with the ETC matrix described in Table I. Looking at Table II, we can observe that solution $S2$ provides the best makespan (11 t.u.), while solution $S1$ has both the worst makespan (16 t.u.) and the best time difference value Δ (2 t.u.). If we consider only the makespan when computing the fitness value of a possible solution, then solution $S2$ will have the best fitness, whereas if we take into account only the time difference value Δ then $S1$ will be the best solution. A more balanced fitness value is obtained if we consider both the makespan and the time difference value Δ . In this case, solution $S2$ is the best,

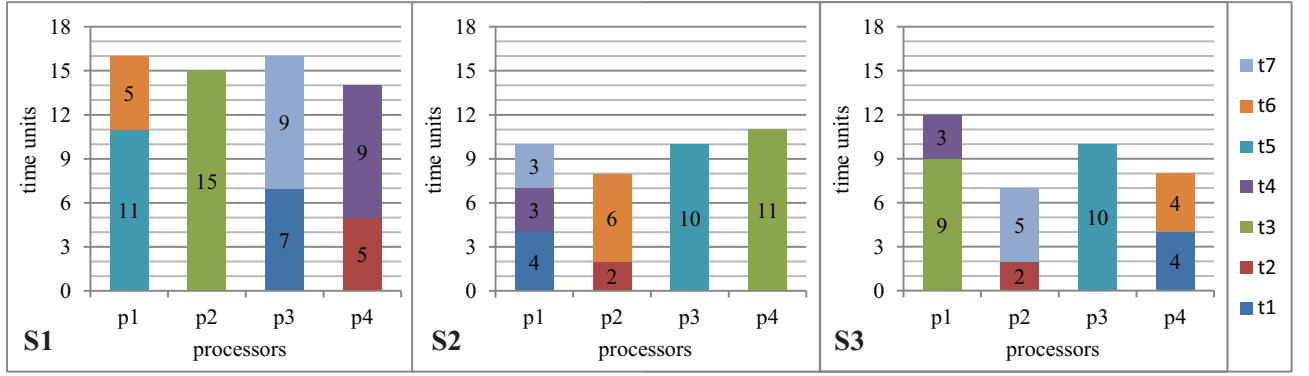


Fig. 1. Three possible task mapping solutions for the ETC matrix from Table I

TABLE II
THE MAKESPAN AND TIME DIFFERENCE Δ VALUES FOR THE THREE
SOLUTIONS FROM FIG. 1

	S1	S2	S3
M	16	11	12
$\Delta + 1$	3	4	5
$M + \Delta + 1$	19	15	17

followed by S3, even though the latter does not have the best makespan nor the best Δ among the three solutions.

C. Stop Condition

The *stop condition* is a combination of factors: a specified number of iterations have been registered, the makespan has not improved for a number of iterations and the best solution has an acceptable makespan or time difference Δ . The total number of iterations should be big enough in order to give the algorithm more time to improve the initial solution, but also limited because we want the tasks mapping process to be as fast as possible. The best chromosome must be improved at least once every few iterations. If a number of iterations have been performed and there is still no change in the makespan value of that chromosome, then it is safe to say that the solution will not improve too soon or, even if it does improve, it will not lead to any significant change. Choosing a proper makespan limit is very difficult, but instead we can use an acceptable difference between the initial makespan and the current one. A percentage of chromosomes from the initial population are initialized with the solution of another mapping algorithm, therefore the makespan of that heuristic is used as reference when computing the difference. If we have a poor solution with a low time difference Δ , then the algorithm must not be terminated. The third stop factor considers both the makespan and the time difference Δ ; the algorithm ends when the initial makespan has improved enough and the time difference Δ is under an imposed threshold.

D. Selection

With each iteration, four steps are taken in order to improve the solution. The first one is the *selection* process

where chromosomes are selected with a certain probability to be used when applying the crossover and mutation operators. Possible selection methods are: random selection, rank selection, tournament selection or roulette wheel selection. The last two methods are the most widely used; tournament selection uses a *selection pressure*, defined as the degree by which the better chromosomes have a better chance of being selected [15], whereas roulette wheel selection uses the fitness of each chromosome to determine their selection probability. Our algorithm uses the latter selection method because, while the best chromosomes are more likely to be selected, the worst chromosomes have also a small chance to be picked out.

E. Crossover

In order to expand the search space and to obtain possible better solutions, we use the *crossover operator*. This process implies selecting two parent chromosomes, then combining them by some means and producing two offspring chromosomes. The most common crossover methods are one-point crossover, two-point crossover and uniform crossover. We chose the former method because it is simpler and it has more or less the same effect as the others. In one-point crossover a child is generated by selecting a crossover point and copying the first part from a parent and the second part from the other parent. Usually, the probability of applying the crossover operator for each chromosome pair has to be high, therefore we choose a probability of 0.7.

F. Mutation

The *mutation operator* consists of moving task t_p from processor p_x to processor p_y . There are various methods of implementing the mutation operator, but the easiest one is the random mutation where a random task is assigned to a random processor. In [1] it is described a smart mutation operator in which a random task is moved to the processor which has the lowest sum of task execution times. A mutation can also try to lower the makespan by choosing a task from an overloaded processor and moving it to another random processor or, it can combine the previous two mutation methods and focus on speeding up the process of finding a solution with a lower makespan while trying to keep

the time difference Δ at a minimum. This mutation has a bigger computation cost and, therefore, it should have a lower probability of being applied to a selected chromosome when compared to the other mutation operators.

In our approach, the mutation can be applied randomly or it can take into account factors like the processors' load or the time difference Δ . The algorithm implies choosing a processor, then selecting a task from that processor and, finally, choosing a new processor where to move the selected task. Each of these three steps can be done in various ways resulting in many mutation possibilities.

With the first step, the mutation operator chooses one of three methods of selecting a processor p_x : randomly select the processor, choose the processor with the biggest total execution time of the allocated tasks, or choose a processor which is in the top percentage regarding the sum of the execution times of each mapped task. The goal is to level the load by moving tasks from the most loaded processors to the least loaded ones, especially when the time difference Δ is too significant.

The second step implies selecting a task from processor p_x . The most used method is to randomly pick the task from the selected processor. Another method is to choose the task with the longest execution time on that processor or one in the top percentage; this minimizes the processor's execution time and increases the chance of allocating a better task in the next iterations. The task with the shortest execution time can also be chosen. In some cases, this can lead to a lower makespan and a lower time difference Δ . As with the previous method, a task can be selected from the top percentage of the tasks with the shortest execution time. Finally, the last method is to choose the task t_p so

$$\Delta_p = \max_{q=1,\dots,n} (\Delta_q) \quad (3)$$

where

$$\Delta_q = etc_{qx} - \min_{z=1,\dots,m} (etc_{qz}) \quad (4)$$

and x is the index of the selected processor p_x . This value introduces a penalty for each task from processor p_x . This penalty equals to the difference between the task's expected time to compute on processor p_x and the lowest expected time of that same task among all the processors. The task with the biggest penalty or a task in the top percentage is then selected.

The final mutation step moves the task selected previously to a processor p_y . We propose three methods of choosing the destination processor. The first one is to randomly pick a processor and to move the selected task there. Another method is to select the processor that is the least occupied, which is the processor with the lowest sum of execution times for the mapped tasks. The third method is to choose the processor which executes the selected task the fastest. In this last case, the task t_p will be moved to processor p_y so

$$etc_{py} = \min_{z=1,\dots,m} (etc_{pz}) \quad (5)$$

This last method has the disadvantage of always choosing the same processor for the same task because it does not

depend on the current task mapping, the same as with the Minimum Execution Time heuristic. To some extent this can lead to a local maximum, where chromosomes represent a satisfying solution, even if not the optimal one, and any subsequent mutation will generate a poorer solution. However, due to the multiple mutation methods we implemented, the local maximum situation has a low probability of occurring.

III. EXPERIMENTAL RESULTS

In order to see how our GA3SM algorithm performs, we tested it along with two other mapping heuristics: a Genetic Algorithm (GA) with a simple random mutation and the Genetic Algorithm with Smart Mutation (GASM) from [1]. The tests were done using our task mapping simulator which, on each run, generates an Expected Time to Compute matrix and runs all the algorithms using that matrix. The simulator used an inconsistent Expected Time to Compute matrix with 10 processors and 500 tasks which ran the genetic algorithms for 1000 iterations 1000 times. The task execution times were values in the [1 - 150] t.u. interval and the maximum allowed difference between the worst and the best execution time of a task on all the processors was 50 time units.

An important element of our genetic algorithm is the fitness function. Depending on the chosen values for the constants a and b from (1), the resulted fitness function can have a different impact on the solution. Table III shows the results of running three algorithms, on the same configuration as the one previously described, with three different fitness functions: $f1$ considers only the makespan, $f2$ considers only the time difference Δ and $f3$ considers both the makespan and the Δ value. The figures included in Table III are expressed in time units and represent the means of the values obtained after running the algorithms 1000 times. For the first fitness function $f1$, all three algorithms obtain a lower value for the mean of the makespans. Also, by running the algorithms with the fitness functions $f2$ and $f3$ we obtain similar values for the time difference Δ . Although $f2$ produces better Δ values, the difference between the values obtained by $f2$ and $f3$ is insignificant compared to the difference between the ones obtained by $f1$ and $f2$.

Each mutation combination has a certain effect on the candidate solution and some combinations can improve the solution more frequently and with a greater makespan decrease. In order to determine which mutation combinations perform better we used the simulator with a standard fitness function which considers only the makespan.

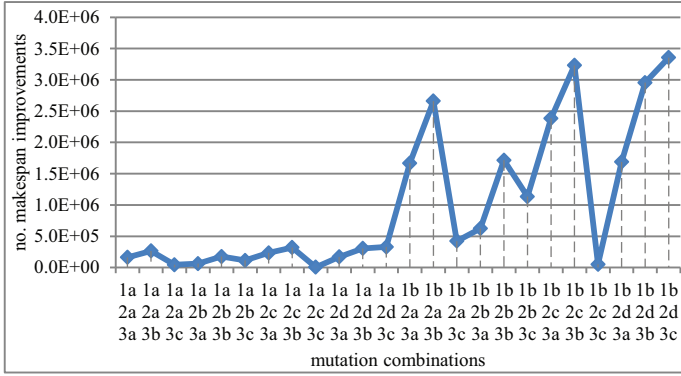
Our algorithm uses a three step mutation as follows:

- 1) Select a processor p_x
 - a) at random,
 - b) with the biggest total execution time.
- 2) Select a task t_p from processor p_x
 - a) at random,
 - b) with the longest execution time on p_x ,
 - c) with the shortest execution time on p_x ,
 - d) which satisfies (3).

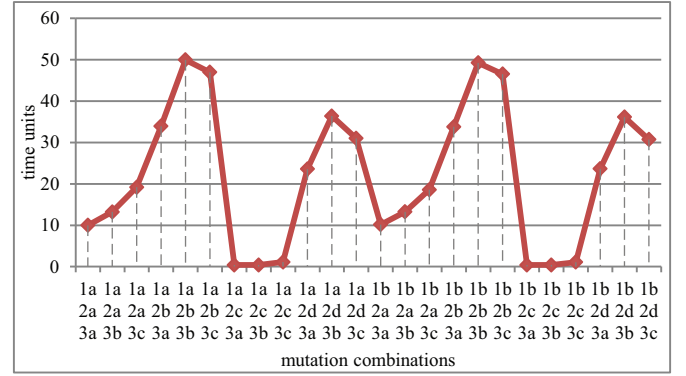
TABLE III

THE MEAN OF THE MINIMUM, MAXIMUM AND AVERAGE TOTAL EXECUTION TIMES AND THE TIME DIFFERENCE Δ FOR 3 MAPPING HEURISTICS WITH 3 DIFFERENT FITNESS FUNCTIONS

	Max Total Time	Min Total Time	Average Total Time	Δ
GA - f1	1076.196847	1046.135204	1066.329862	30.061643
GASM - f1	1064.521213	1061.273905	1062.881894	3.247308
GA3SM - f1	1054.394584	1049.586877	1052.780916	4.807707
GA - f2	1091.710901	1074.979748	1083.805942	16.731154
GASM - f2	1081.178122	1078.401073	1079.688022	2.777049
GA3SM - f2	1075.27971	1072.947209	1074.163424	2.332501
GA - f3	1083.905639	1067.601283	1076.40367	16.304356
GASM - f3	1070.165848	1067.368077	1068.714378	2.797771
GA3SM - f3	1065.808379	1063.430383	1064.699774	2.377996



(a) Number of makespan improvements for each mutation combination



(b) Average makespan improvement for each mutation combination

Fig. 2. Test results for 24 mutation combinations

3) Select a processor p_y and move task t_p to that chosen processor

- at random,
- with the lowest sum of execution times,
- which executes the selected task t_p the fastest.

The first mutation step can be done with two methods (1a and 1b), the second step with four methods (2a, 2b, 2c and 2d) and the third step with three methods (3a, 3b and 3c) as previously stated. For example, the mutation combination 1a 2a 3a means that all three steps (selecting processor p_x , then choosing a task t_p from that processor and moving it to processor p_y) are done randomly. Each chromosome from the population has a probability of 0.7 of entering in the crossover phase and a probability of 0.8 of being selected for mutation.

The test results are shown in Fig. 2, where, on the horizontal axis, we represent the mutation combinations and, on the vertical axis, we represent the number of makespan improvements (Fig. 2(a)) and the average makespan improvement values (Fig. 2(b)). The data in both figures was obtained by applying a randomly selected mutation combination for each selected chromosome.

Fig. 2(a) shows that, if processor p_x is selected randomly (1a), there will be a lower probability to produce a better solution from the mutation as opposed to choosing the pro-

cessor with the biggest total execution time (1b). Also, most of the times a candidate solution was improved it resulted from the combination 1b 2d 3c meaning that, moving the task which satisfies (3) (2d) from the processor with the biggest total execution time (1b) to the processor which executes the selected task t_p the fastest (3c) will have a high probability of improving the makespan. The combinations 1a 2c 3c and 1b 2c 3c apparently have no effect on the chromosome because a task is moved from the processor which executes it the fastest to a processor with the same property. But, if we take into account the fact that there can be more than one processor which provides the shortest execution time for that task or, if we consider the top percentage and not just the best processor for this situation, it is safe to say that this mutation combination can prove to be quite useful, especially when trying to obtain a more fine-grained solution.

The greatest average decrease of the makespan and, implicitly, the best candidate solution improvement, will result by moving the task with the longest execution time to either the processor with the lowest sum of execution times or to the processor which executes the selected task the fastest (combinations 1a 2b 3b, 1b 2b 3b, 1a 2b 3c and 1b 2b 3c) as presented in Fig. 2(b). On the other hand, the lowest average decrease of the makespan will be obtained with all the combinations which move the task with the shortest execution time (2c) from a processor to any other processor.

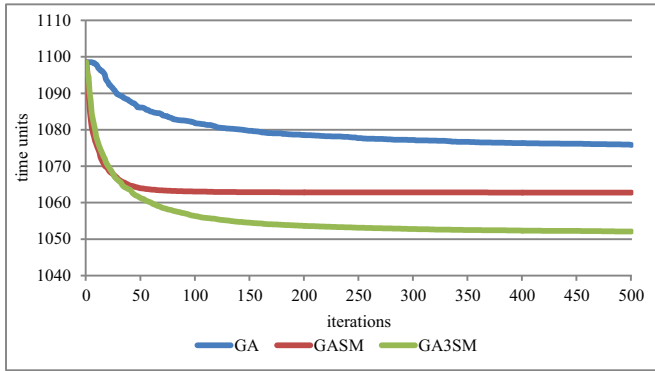


Fig. 3. The makespan comparison for 500 iterations over 1000 runs of the GA, GASM and GA3SM algorithms

Our algorithm, the Genetic Algorithm with 3-Step Mutation (GA3SM), obtains a much better solution than the standard Genetic Algorithm (GA) with simple random mutation and also a better solution than the Genetic Algorithm with Smart Mutation (GASM) from [1]. For the first iterations GASM provides the best solution, but, as seen in Fig. 3, after iteration number 23 GA3SM outperforms GASM by obtaining a better makespan and improving the solution with almost every iteration. All three algorithms start from the same best makespan of 1098.51 t.u. and, after 500 iterations, GA obtains the poorest solution (1075.83 t.u.), GASM comes in second (1062.78 t.u.) and GA3SM provides the best solution among the three (1052.06 t.u.).

IV. CONCLUSIONS AND FUTURE WORK

The task mapping problem is a difficult one because of the large solution search space. The genetic algorithm is well fitted to obtain an acceptable solution in a fair amount of time. Most likely, obtaining a good makespan is not enough, because the processors' load has to be balanced as well. Here comes into play the fitness function which has to consider both the makespan and the difference between the time in which the first and the last processor finish running their tasks.

Our tests have shown that using only the makespan in the fitness function may lead to an unacceptable load imbalance whereas when considering the time difference the load is much better balanced at the expense of a slightly higher makespan value. Our proposed heuristic, the Genetic Algorithm with 3-Step Mutation, proved to generate the best solution for all three tested fitness functions compared to GA and GASM.

The mutation operator of GA3SM uses a three step mutation to speed up the convergence process. The different targeted mutation combinations produce a better solution compared to GA and GASM by moving tasks from the most occupied processors to the least occupied ones or to the processors which execute the selected tasks the fastest.

According to our testing, certain mutation combinations improve the fitness more than others and, taking into account that each combination has a certain probability of occurring,

we intend to search for a method to integrate another genetic algorithm in GA3SM in order to determine the optimal probabilities for the mutation combinations. To further increase the algorithm's performance we also consider using adaptive mutation probabilities because some combinations may improve the solution more significantly in the first iterations, while other mutation combinations are better at obtaining a fine-grained solution.

ACKNOWLEDGEMENT

This paper was realised with the support of EURODOC "Doctoral Scholarships for research performance at European level" project, financed by the European Social Fund and the Romanian Government.

REFERENCES

- [1] A. Alexandrescu and M. Craus, "Improving mapping heuristics in heterogeneous computing," in *Proceedings ECIT2010 6th European Conference on Intelligent Systems and Technologies, October 7-9, Iasi, 2010*, pp. 1–12.
- [2] T. D. Braun, H. J. Siegel, N. Beck, L. L. Böllöni, M. Maheswaran, A. I. Reuther, J. P. Robertson, M. D. Theys, B. Yao, D. Hensgen, and R. F. Freund, "A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems," *J. Parallel Distrib. Comput.*, vol. 61, no. 6, pp. 810–837, June 2001.
- [3] Q. Kang and H. He, "A novel discrete particle swarm optimization algorithm for meta-task assignment in heterogeneous computing systems," *Microprocessors and Microsystems*, vol. 35, no. 1, pp. 10 – 17, 2011.
- [4] J. Yang, H. Xu, L. Pan, P. Jia, F. Long, and M. Jie, "Task scheduling using bayesian optimization algorithm for heterogeneous computing environments," *Applied Soft Computing*, vol. 11, no. 4, pp. 3297 – 3310, 2011.
- [5] Q. Kang, H. He, and H. Song, "Task assignment in heterogeneous computing systems using an effective iterated greedy algorithm," *Journal of Systems and Software*, vol. 84, no. 6, pp. 985 – 992, 2011.
- [6] Q.-M. Kang, H. He, H.-M. Song, and R. Deng, "Task allocation for maximizing reliability of distributed computing systems using honeybee mating optimization," *Journal of Systems and Software*, June 2010.
- [7] B. Ucar, C. Aykanat, K. Kaya, and M. İkinci, "Task assignment in heterogeneous computing systems," *Journal of Parallel and Distributed Computing*, vol. 66, no. 1, pp. 32–46, Jan. 2006.
- [8] S. Salcedo-Sanz, Y. Xu, and X. Yao, "Hybrid meta-heuristics algorithms for task assignment in heterogeneous computing systems," *Comput. Oper. Res.*, vol. 33, pp. 820–835, March 2006.
- [9] Y. Wen, H. Xu, and J. Yang, "A heuristic-based hybrid Genetic-Variable neighborhood search algorithm for task scheduling in heterogeneous multiprocessor system," *Information Sciences*, Oct. 2010.
- [10] S. Dey and S. Majumder, "Task allocation in heterogeneous computing environment by genetic algorithm," in *Proceedings of the 4th International Workshop on Distributed Computing, Mobile and Wireless Computing*, ser. IWDC '02. London, UK, UK: Springer-Verlag, 2002, pp. 348–352.
- [11] M. Pinedo, *Scheduling: Theory, Algorithms and Systems*, 2nd ed. Prentice Hall, 2006.
- [12] O. Engin, G. Ceran, and M. K. Yilmaz, "An efficient genetic algorithm for hybrid flow shop scheduling with multiprocessor task problems," *Appl. Soft Comput.*, vol. 11, pp. 3056–3065, April 2011.
- [13] F. A. Omara and M. M. Arafa, "Genetic algorithms for task scheduling problem," *Journal of Parallel and Distributed Computing*, Oct. 2009.
- [14] A. J. Page, T. M. Keane, and T. J. Naughton, "Multi-heuristic dynamic task allocation using genetic algorithms in a heterogeneous distributed system," *Journal of Parallel and Distributed Computing*, vol. Article in Press, 2010.
- [15] B. L. Miller, B. L. Miller, D. E. Goldberg, and D. E. Goldberg, "Genetic algorithms, tournament selection, and the effects of noise," *Complex Systems*, vol. 9, pp. 193–212, 1995.

Parallel Quantum Computer Simulation on the GPU

Andrei Amariutei and Simona Caraiman

Abstract—Simulation of quantum computers using classical computers is a hard problem with high memory and computational requirements. Parallelization can alleviate this problem, allowing the simulation of more qubits at the same time or the same number of qubits to be simulated in less time. A promising approach is to exploit the high performance computing capabilities provided by the latest graphical processing units. In this paper we present a parallel implementation of the QC-lib quantum computer simulator on the GPU using the CUDA programming model. The proposed scheme for partitioning the terms that describe the state of a quantum register takes advantage of the specific characteristics of the CUDA memory spaces and allows for an efficient parallelization of the general single qubit operator. Experimental results indicate that very good speed-ups can be obtained in contrast with the sequential implementation.

I. INTRODUCTION

The research in quantum informatics has gained an immense interest due to the remarkable results obtained for the factorization [1] and search [2] problems. These results prove the huge computational power of a quantum machine with respect to the classical computers. However, building quantum computers represents an immense technological challenge and, at present, the quantum hardware is only available in research labs. Under these circumstances quantum simulators have become valuable instruments in developing and testing quantum algorithms and in the simulation of physical models used in the implementation of a quantum processor.

Simulating a quantum computer on a classical computer is a computationally hard problem. According to Feynman's paper [3] classic computers will never be able to perform simulations of full behavior of a quantum system in a polynomial time. Because of the exponential behavior of quantum systems, simulating them on conventional computers requires an exponential amount of operations and storage. Parallelization alleviates this problem, allowing the simulation of more qubits at the same time or the same number of qubits to be simulated in less time.

In this paper we describe the parallelization of an open source quantum computer simulation library, QC-lib, by employing parallel computing on the Graphical Processing Unit (GPU) based on the CUDA technology. The paper is organized as follows. In section 2 we give a short introduction to quantum computing. Section 3 shortly describes the concept of quantum computer simulation and classifies the

existing simulators, while in section 4 some basic elements of the CUDA technology are mentioned. The main features of the Quantum Computation Library (QC-lib), the back end of an interpreter for the QCL Quantum Computation Language [4], are presented in section 5, followed by a detailed description of the simulation design based on the CUDA programming model. In section 5 we also discuss the performance measurements of our implementation and we draw our conclusions in section 6 along with sketching the future development directions.

II. QUANTUM COMPUTATION - BASIC CONCEPTS

Just like a classical computer, a quantum computer is built out of three main parts: processor, memory and input/output. A quantum computer can be formally described by $M = (H, O, T, \delta, \beta)$, where H - represents the states space (C^{2^n} Hilbert space) of the quantum system, O - the set of unitary transformations, T - the set of measurement commands, δ - is an initialisation operator and β - describes the final measurement. The quantum analogous of the classical bit is the qubit. A qubit is a quantum system whose states can be completely described by the superposition of two orthonormal basis states, labelled $|0\rangle$ and $|1\rangle$ (in a Hilbert space $H = C^2$, $|0\rangle = \begin{pmatrix} 1 & 0 \end{pmatrix}^T$, $|1\rangle = \begin{pmatrix} 0 & 1 \end{pmatrix}^T$). Any state $|\psi\rangle$ can be described by:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, |\alpha|^2 + |\beta|^2 = 1 \quad (1)$$

The orthonormal system $|0\rangle, |1\rangle$ is called *computational basis*. The machine state $|\psi\rangle$ of an n -qubit quantum computer is an unit vector in Hilbert space $H = C^{2^n}$:

$$|\psi\rangle = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle, \quad (2)$$

where $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$ and $|\alpha_i|^2$ represents the probability of getting value $|i\rangle$ when measuring the register.

A quantum register s is represented by a sequence of qubits. If s is an n -qubit quantum register and U is an operator in the states space H , then the operator $U(s)$ applied to a register is called a *quantum gate*.

The quantum analogous of the classical NOT gate is labeled X and can be defined such that $X|0\rangle = |1\rangle$ and $X|1\rangle = |0\rangle$. The quantum NOT gate act similarly with its classical counterpart, although, unlike in the classical case, its action is linear: state $\alpha|0\rangle + \beta|1\rangle$ is transformed in a corresponding state $\beta|0\rangle + \alpha|1\rangle$. A convenient way of representing the action of the quantum NOT gate is in matrix form:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \quad (3)$$

A. Amariutei is with Faculty of Automatic Control and Computer Engineering, Technical University of Iasi, Romania a.amariutei@gmail.com

S. Caraiman is with Faculty of Automatic Control and Computer Engineering, Technical University of Iasi, Romania sarustei@cs.tuiasi.ro

Hadamard gate, given by the following matrix form

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad (4)$$

is sometimes described as "square root of NOT" because when applying it to any of the $|0\rangle$ or $|1\rangle$ basis states produces an equal mixture of both of them:

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad \text{and} \quad H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle). \quad (5)$$

III. SIMULATION OF QUANTUM COMPUTERS

Quantum computer simulators come in different flavours and have different levels of complexity [5], [6]. Understanding the functions and characteristics of existing quantum computer simulators is essential for the researchers in the field in order to avoid effort duplication and to reuse the already achieved results as foundation for future work. We classify existing quantum computing simulation software according to the purpose of our present work in two categories: sequential simulators and simulators that use parallel computing. The former category has a quite good representation and contains different types of simulators [6]: quantum programming languages (QCL [4], Q language [7], [8], Quantum Superpositions [9], QuBit [10], Q-gol [11], QDD [12], Quantum Lambda Calculus [13]), quantum compilers (Qubiter [14], GQC [15]), quantum circuits simulators (QCAD [16], QuaSi [17], Libquantum [18], QCS [19]), quantum hardware emulators (QCE [20], QSS [21]) and purely pedagogical software (quantum Turing machine simulator, Quantum Search Simulator, Shor's algorithm simulator).

The latter category of simulators emerged due to the exponential nature of quantum computation and the inability of classical computing device to simulate it efficiently.

The first such simulator was developed by Obenland and Despain in 1998 [22]. It is written in C and runs on distributed memory parallel systems. The drawback of this simulator is the fact that it is based on the physical model of a certain experimental implementation of a quantum computer. A quantum simulator independent from a physical model for the quantum computer is the one introduced by J. Niwa in 2002 [23]. It is developed in C and runs in shared memory parallel systems thus eliminating the need for distributing and collecting the states of a quantum register. The Fraunhofer Computing Portal [24] is a computing service and a collaborative work-space for the simulation and investigation of quantum algorithms. Although not available anymore, the purpose of this project was to provide researchers with a simulation engine based on several computation nodes through the means of an easily accessible web interface. In [25] authors present a software package available in C and Fortran 90, developed for the simulation of an ideal quantum computer. The software is portable, running on different computer architectures, from PCs to distributed and/or shared memory parallel machines. The maximum number of qubits is given by the available memory on the machine it runs on. 37 qubits simulations are described using

3 TB of memory. Another simulation software with parallel processing capabilities is QCMP [26]. It extends the QDENSITY package for Mathematica by implementing parallel processing using MPI (Message Passing Interface). Potential applications of this software tool include the stability study and error correction of the quantum computation using the Hamiltonian dynamics.

In [27] and [28] authors present a parallel version of the QC-lib, the simulation environment for the execution of QCL programs in the absence of quantum resources. The authors describe the MPI parallelization of the general single qubit operator and the C-NOT operator. The speed-ups obtained executing the Hadamard transform and Grover's search algorithm are promising, up to 17.7 on 16 processors, for the largest problem size, 29 qubits.

A quantum register contains a number of basevectors, each with a corresponding amplitude. When the quantum register finds itself in a superposition of states, the number of vectors rises exponentially. For example, a quantum register of only 10 qubits contains 1024 basevectors. Any operation applied to the quantum state can be represented by a square matrix with the dimension equal to the number of basevectors. The operation applied is in essence the multiplication of this matrix with the vector of amplitudes of the quantum state. As the dimension of the quantum register gets larger, the necessary resources to apply the matrix-vector product rises exponentially making it impossible to execute it on a PC. A natural approach for solving this problem would be to parallelize the multiplication operation. In [29] it is proved that this approach is not well suited for distributed memory simulation environments because of high inter-process communication costs. [29] suggests the description of quantum algorithms using the tensor product operator. The effected tests proved the efficiency of this approach, but its major disadvantage is that not all quantum computing problems allow for such a parallelization.

A new approach for the simulation of quantum computation emerged recently along with the general purpose GPU computing. Due to the massive computing power of recent GPUs, these have become very popular for solving general purpose computations. The quantum simulation problem is well suited for the parallel SIMT execution on the GPU. Such an approach is described in [30].

IV. CUDA - COMPUTE UNIFIED DEVICE ARCHITECTURE

CUDA (*Compute Unified Device Architecture*) technology, developed by NVIDIA, is a hardware and software architecture that is able to exploit the massive parallel processing power of the recent NVIDIA graphic cards. In order to develop GPU-enabled applications, programmers can make use of various programming languages: C/C++ for CUDA, OpenCL, Fortran or DirectCompute.

From the point of view of the hardware, the GPU consists of several multiprocessors working in a SIMT (Single Instruction Multiple Threads) fashion, each of them containing a certain number of streaming processors, as shown in Figure 1 (left). Several memory spaces are available on the GPU:

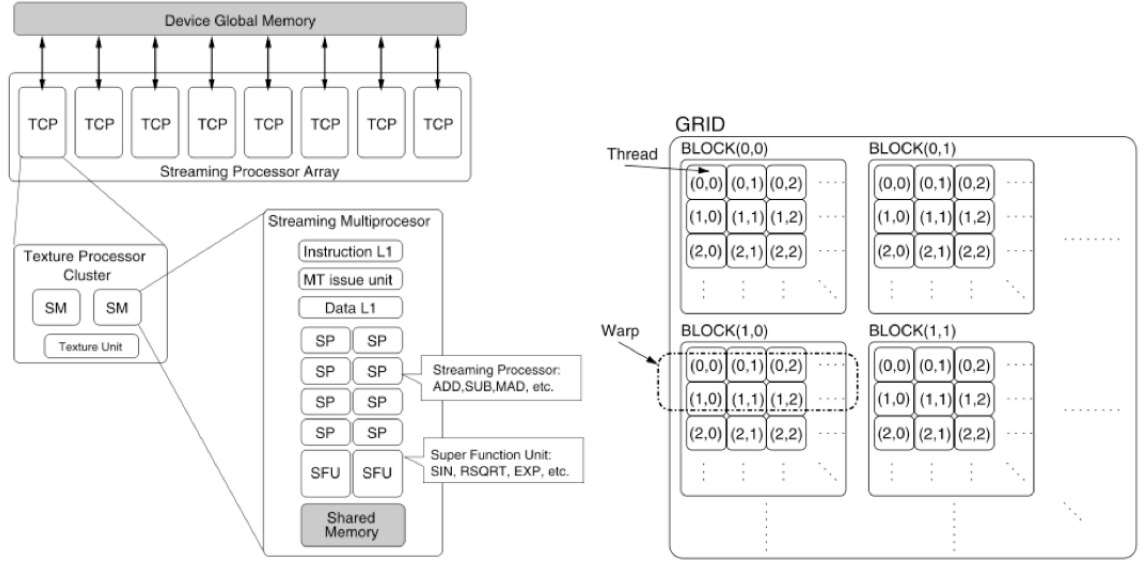


Fig. 1. The CUDA architecture: hierarchy of memory spaces (left), execution model (right).

(1) *Global memory*, the default memory space that can be accessed by each SM. The host (CPU) has R/W access to this memory space. (2) Moreover, each SM owns a private on-chip memory space representing the *shared memory* which is smaller than the global memory but with smaller access times. (3) There are also other addressing spaces, omitted in the figure, namely the *texture memory* and *constant memory*.

The CUDA execution model is based on an hierarchy of abstracting levels: grids, blocks, warps and threads, as shown in Figure 1 (right). A thread is the fundamental execution unit associated to a SM. A block is a group of threads that are executed on a multiprocessor, and a grid is formed out of several blocks. When the number of blocks is greater than the number of multiprocessors, the blocks inside a grid are distributed between multiprocessors. A warp is a group of threads is executed in SIMT fashion, allowing for threads in the same warp to execute the same instruction simultaneously. There are two types of code sequences: the ones executed on the CPU (*host*), and the ones executed on the GPU (*device*), called kernel functions. The CPU is responsible with both transferring the data between host and device memory spaces and with the invocation of the kernels and setup of the execution parameters (block and grid dimensions). The kernels are executed in a SIMT manner by the SMs.

The programmer is responsible with defining the number of blocks, threads per block and their distribution (1D, 2D, 3D). He/she is also responsible with applying coding strategies in order to provide a good distribution and balancing of the tasks. Other aspects that need to be considered are the memory access and thread synchronization. Global memory accesses are grouped in a single instruction for the threads in the same warp, given that consecutive memory locations are accessed. This provides a better bandwidth. CUDA provides a synchronization mechanism for the threads belonging to

the same block, but no mechanism is provided for global synchronization between blocks of threads. Thus, synchronization between blocks must be explicitly implemented by the programmer through consecutive invocations of the kernels.

V. GPU-ENABLED QUANTUM SIMULATION

A. Quantum Computation Language (QCL) and the Quantum Computer Simulator Library (QC-lib)

QCL [4] is a high level quantum programming language written in C++. It uses a complex numbers for the representation of quantum states. Other simulators use Bayesian networks (Quantum Fog, Qubiter) or BDDs (QDD) to represent the quantum state. Although the use of binary decision diagrams allows modelling quite large quantum states [12], this facility imposes the representation of a "digital" quantum computer versus the "analog" model which can be manipulated by QCL.

In contrast with other quantum programming languages, QCL is architecture independent and it is designed to work on any qubit based computer architecture.

The elementary datatype in QCL is the quantum register, *qureg*. It is a reference to one or more qubits in a so-called *quantum heap*. Elementary operations can be applied to these registers: initialization, unitary transformations and measurement. Nevertheless, a special feature of QCL is the possibility to write complex quantum operators, besides the general unitarian ones. These operators are defined like classic procedures and functions (in fact reversible pseudo-classic gates - *qufunct*), but they have one or more quantum registers as input. Variants of the basetype register are also defined and although they have the same physical interpretation, the different types impose different constraints on the allowed operations. Constant registers (*quconst*),

void registers (*quvoid*) and scratch registers (*quscratch*) are defined.

The classical extension of the language allows integration of classic pre- and post-processing and presents facilities for functions, flow control, interactive I/O and different classical datatypes (*int*, *real*, *complex*, *boolean*, *string*).

QCL is an universal language, allowing the implementation of all known quantum algorithms. The ultimate purpose of QCL is to provide the means for specifying algorithms executed using quantum resources (or a combination of quantum and classical resources). Nevertheless, the author also provides a simulation environment, QC-lib, in order to allow QCL programs to be executed in absence of quantum hardware.

QC-lib is a C++ library for simulation of quantum computers at an abstract functional level [31]. It means that even though simulation of certain hardware architecture (for example the linear ion trap) is also possible, the main purpose is the verification of quantum algorithms at high level, especially in the field of number theory. The main features of this simulator are: basevectors of arbitrary length (not limited by the length of the word for a certain hardware implementation), efficient representation of quantum states using hash tables, nesting of substates and arbitrary combination of qubits, composition and tensor product of quantum operators (quantum gates), easy extendibility by class inheritance. See [32] and [31] for a detailed description of the class hierarchy in QC-lib.

In order to simulate a problem using QC-lib it is necessary to describe the algorithm using the formalism of quantum gates. An example of such an analysis is given in [33] where this formalism is used to allow for the decomposition of the computation in elementary operations in the case of the Bernstein-Vazirani problem.

B. Single Qubit Operators

The application of a single qubit operator on a quantum register implies computing the output state of the register,

$$|\psi^{out}\rangle = \sum_{i=0}^{2^n-1} \alpha_i^{out} |i\rangle \quad (6)$$

given the state

$$|\psi^{in}\rangle = \sum_{i=0}^{2^n-1} \alpha_i^{in} |i\rangle \quad (7)$$

and a 2x2 transform U ($|\psi^{out}\rangle = U^{\otimes n} |\psi^{in}\rangle$, where $\otimes$ denotes the tensor product). This implies computing the values of coefficients α_i^{out} , $i = 0, n-1$ based on coefficients α_i^{in} , $i = 0, n-1$.

Due to the exponential computing power of a quantum computer compared to a classical one, the quantum simulation requires exponentially larger memory space and computation resources. In order to represent the state of a n -qubit register a 2^n vector of complex numbers is used to store the amplitudes α_i from Equation 2. Thus, in order to represent the state of a n -qubit quantum register using

a conventional (classic) computer, at least 2^{n+3} bytes are needed. The required memory needed to simulate a quantum computer with n qubits grows exponentially with the number n of qubits.

When applying a quantum operator on a register in QC-lib two lists of terms are generated: one for storing the amplitudes of the current state and one to accumulate the resulting amplitudes. Thus, for example, for $n = 25$ ($n = 34$) 2 GB (1 TB) of memory is needed to store a single arbitrary state $|\psi\rangle$.

It is obvious that the simulation of quantum problems of interesting sizes enforces the use of high performance computing devices.

Any operation which is applied to a quantum state can be represented as a matrix. The matrix to be applied to any particular quantum state is a square matrix of the same dimension as the number of basevectors. Applying a matrix to a large quantum state takes a very long time on one computer.

In order to avoid the representation of the whole $U^{\otimes n}$ matrix of dimension $2^n \times 2^n$, the 2x2 transform, U , can be applied on each qubit of the register. For example, applying operator U on term $|10\rangle$ of a 2-qubit register gives

$$U^{\otimes n} |10\rangle = U^{\otimes n} (|1\rangle \otimes |0\rangle) = (U|1\rangle) \otimes (U|0\rangle). \quad (8)$$

Figure 2 shows the application of the general single qubit operator, U , on a 2-qubit register, where U is in the form

$$U = \begin{pmatrix} w & x \\ y & z \end{pmatrix}. \quad (9)$$

The state vector is divided in two vectors on which 2x2 transforms are applied. The generated terms form a new state vector. This procedure is applied on vectors of any dimensions and thus a $2^n \times 2^n$ transform applied to a 2^n vector represents a sequence of 2^{n-1} independent transforms of size 2x2.

C. Implementation Using the CUDA Programming Model

The simulation of a quantum operator on the GPU implies partitioning the state vector in sets of fixed dimensions and assigning them to CUDA blocks, where the sub-vectors are processed in parallel on a SIMT (single Instruction Multiple Thread) fashion. The general simulation scheme is presented in Figure 3. The outer loop of the code executed by the host (CPU) invokes the kernel executed on the device (GPU) and computes the resulting amplitudes in parallel.

The steps executed by the kernel are: copying the set of amplitudes from global to shared memory, SIMT parallel processing of the amplitudes inside each block and finally copying the amplitudes of the generated terms from shared to global memory on their original positions.

Similar to the sequential case, the transform is applied to groups of two terms chosen such that in their binary codification only the j th position differs, where j is the qubit is the current qubit acted upon by the transform. The problem consists in finding appropriate mapping functions between shared and global memory relative to the thread index.

$$U^{\otimes 2} \times \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} w & x \\ y & z \end{bmatrix} \begin{bmatrix} w & x \\ y & z \end{bmatrix} \times \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \Rightarrow \begin{cases} \begin{bmatrix} w & x \\ y & z \end{bmatrix} \times \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} a'_0 \\ a'_1 \end{bmatrix} \\ \begin{bmatrix} w & x \\ y & z \end{bmatrix} \times \begin{bmatrix} a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} a'_2 \\ a'_3 \end{bmatrix} \end{cases} \Rightarrow \begin{bmatrix} a'_0 \\ a'_1 \\ a'_2 \\ a'_3 \end{bmatrix}$$

Fig. 2. A 2x2 quantum transform applied to a 4 state vector

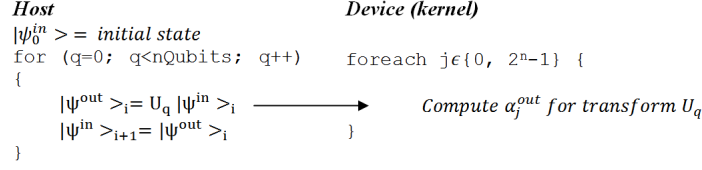


Fig. 3. Simulation scheme (invocation of the kernels executed in parallel on the GPU)

For a qubit j , applying the operator on a term of the quantum state modifies both the amplitude of the current term and of its pair (the term with flipped bit in position j). The newly generated amplitudes are stored in temporary variables which will replace the original values after the application of the operator on both the terms that form the pair.

Each thread in a block is assigned a pair of two terms and thus the nt threads of a block can process the $2nt$ terms assigned to a block in parallel.

The application of the operator on the entire register implies two distinct phases:

(1) For the first $\log_2 2nt$ qubits, the quantum operator can be applied in-place, without the need to access other terms from global memory as they are assigned to the same block (first $2nt$ terms are assigned to block 0, next $2nt$ terms are assigned to block 1 and so on).

(2) For the rest of the qubits, starting with index $\log_2 2nt$, the kernel will be applied on a single qubit. This is necessary because the qubits forming the pair do not belong to the same block anymore. In this case a different memory mapping is necessary:

- a mapping function assigns terms from global memory to a certain block, in groups of $2nt$ terms, and copies them in the shared memory of each block based on the index of the qubit that is being operated on;
- the operator is applied on the terms in shared memory in parallel;
- the resulted amplitudes are copied in the global memory and the kernel execution is ended, thus obtaining a synchronization between blocks as no other mechanism for block synchronization is offered by CUDA;
- the actual mapping of terms from global to block memory is accomplished with the following principle: the pairs of the $2nt$ terms from the current block will be found in a pair block, thus in order to apply the operator, one block of threads will actually receive the first halves of both blocks of terms and the pair block of threads will receive the second halves of both blocks of terms.

D. Experimental Results

The proposed parallelization of the single qubit operator was integrated in the QClib implementation and thus the QCL statements need no modifications. We used the Hadamard transform as a test case. This gate is useful because applying it to either of the basis states $|0\rangle$ and $|1\rangle$ produces an equal mixture of both of them and thus is able to generate all possible states of a quantum register (see Equation 4).

The simulation was run on an NVIDIA Quadro Plex Model IV, with two FX5600 on SLI and CUDA compute capability 1.0. The maximum size of the simulated quantum register is 25 qubits due to the limitations introduced by the memory allocation scheme in QC-lib and the available memory on the host (2 GB).

Figure 4 shows the execution times recorded for both the sequential and parallel simulations, while Figure 5 presents the speedup of the GPU implementation computed as t_{CPU}/t_{GPU} . It can be observed that the sequential execution time increases by a factor of approx. 2 for each extra qubit as each extra qubit represents a doubling of the problem size. For a small number of qubits, the GPU implementation is more costly as it requires memory transfers between CPU and GPU as well as between global and shared memory on the GPU. Nevertheless, the speed-up of the parallel implementation increases with the number of qubits and a value of 384 is obtained for the maximum size of the simulated quantum registry.

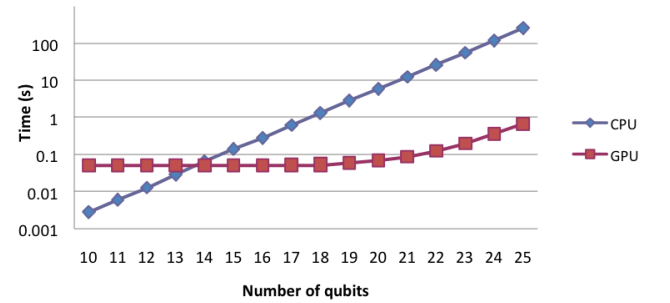


Fig. 4. Execution time for the Hadamard transform

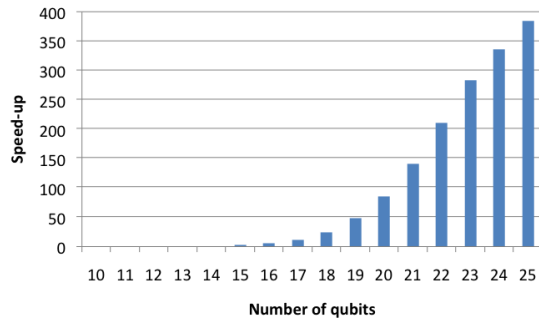


Fig. 5. Speed-up of the GPU execution over the sequential implementation, in the case of the Hadamard transform

VI. CONCLUSIONS AND FUTURE WORK

Simulation of quantum computation using classic machines is a hard problem because representing the memory of a quantum computer requires allocating space for an exponential number of states and an exponential number of operations are executed on these states. These limitations make quantum simulation a good candidate for parallel execution. This paper presented the parallel implementation of the QcLib simulator on the GPU using the CUDA programming model. The experimental results prove that the proposed parallelization scheme delivers significant speed-ups compared to the sequential execution, being almost 400 times faster for only 25 qubits.

The obtained results justify further development which will address the efficient implementation of more operators and algorithms and investigating their performances. The two qubit CNOT gate will be implemented next, which together with the single qubit operator are universal for quantum computation. Although the CNOT gate is a two qubit gate, it only modifies a single qubit so we envision that a similar parallelization scheme could be used.

VII. ACKNOWLEDGMENTS

This paper was supported by the project PERFORM-ERA "Postdoctoral Performance for Integration in the European Research Area" (ID-57649), financed by the European Social Fund and the Romanian Government.

REFERENCES

- [1] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *SFCS '94: Proceedings of the 35th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, 1994, pp. 124–134.
- [2] L. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of 28th ACM Annual STOC*, 1996, pp. 212–219.
- [3] R. Feynman, "Simulating physics with computers," *International Journal of Theoretical Physics*, vol. 21, no. 6, pp. 467–488, 1982.
- [4] B. Ömer, "Structured quantum programming," Ph.D. dissertation, TU Vienna, 2003, available online at <http://tph.tuwien.ac.at/oemer/doc/structqprog.pdf>.
- [5] J. Wallace, "Quantum computer simulators," *International Journal of Computing Anticipatory System*, vol. 10, p. 230?245, 2001.
- [6] H. D. Raedt and K. Michielsen, *Handbook of Theoretical and Computational Nanotechnology*. Los Angeles: American Scientific Publisher, 2006, vol. 3, ch. Computational Methods for Simulating Quantum Computers, arXiv:quant-ph/0406210.

- [7] S. Bettelli, "Toward an architecture for quantum programming," Ph.D. dissertation, Università di Trento, Italy, 2002.
- [8] S. Bettelli, T. Calarco, and L. Serafini, "Toward an architecture for quantum programming," *The European Physical Journal D - Atomic, Molecular, Optical and Plasma Physics*, vol. 25, no. 2, pp. 181–200, 2004.
- [9] S. Lembark, "Quantum-superpositions-2.02," <http://search.cpan.org/~lembark/Quantum-Superpositions-2.02/lib/Quantum/Superpositions.pm>.
- [10] "Qubit," <http://www.bluedust.com/qubit/>.
- [11] G. Baker, "Qgol. a system for simulating quantum computations: Theory, implementation and insights," 1996, honours thesis.
- [12] D. Greeve, "Qdd : A quantum computer emulation library," <http://thegreves.com/david/QDD/qdd.html>.
- [13] "Quantum lambda calculus," <http://www.het.brown.edu/people/andre/qlambda/>.
- [14] R. Tucci, "Qubiter," <http://www.ar-tiste.com/qubiter.html>.
- [15] M. Bremner, C. Dawson, J. Dodd, A. Gilchrist, A. Harrow, D. Mortimer, M. Nielsen, and T. Osborne, "Practical scheme for quantum computation with any two-qubit entangling gate," *Phys. Rev. Lett.*, vol. 89, no. 24, 2002.
- [16] H. Watanabe, "Qcad," <http://apollon.cc.u-tokyo.ac.jp/watanabe/qcad/>.
- [17] M. Eck, P. Wocjan, and R. Zeier, "Quasi - quantum circuit simulator," <http://iaks-www.ira.uka.de/QIV/QuaSi/aboutquasi.html>.
- [18] H. Weimer, "libquantum: the c library for quantum computing and quantum simulation," <http://www.libquantum.de/>.
- [19] A. Nagai and Y. Tokunaga, "Qcs: Quantum computer simulator," <http://www-imai.is.s.u-tokyo.ac.jp/tokunaga/QCS/simulator.html>.
- [20] K. Michielsen, H. D. Raedt, and K. D. Raedt, "A simulator for quantum computer hardware," *Nanotechnology*, vol. 13, no. 1, pp. 23–28, 2002.
- [21] S. Schneider, "Quantum system simulator," Master's thesis, Massachusetts Institute of Technology, 2000, <http://web.mit.edu/scottsch/www/qc/>.
- [22] K. Obenland and A. Despain, "A parallel quantum computer simulator," 1998, <http://arXiv.org/abs/quant-ph/9804039>.
- [23] J. Niwa, K. Matsumoto, and H. Imai, "General-purpose parallel simulator for quantum computing," in *UMC '02: Proceedings of the Third International Conference on Unconventional Models of Computation*. London, UK: Springer-Verlag, 2002, pp. 230–251.
- [24] H. Rosé, T. Aßelmeyer-Maluga, M. Kolbe, F. Niehörster, and A. Schramm, "The fraunhofer quantum computing portal - www.qc.fraunhofer.de - a web-based simulator of quantum computing processes," Fraunhofer FIRST, Berlin, techreport quant-ph/0406089, 2004.
- [25] K. D. Raedt, K. Michielsen, H. D. Raedt, B. Trieuc, G. Arnoldc, M. Richterc, T. Lippertc, H. Watanabed, and N. Itoe, "Massively parallel quantum computer simulator," *Computer Physics Communications*, vol. 176, no. 2, pp. 121–136, 2007.
- [26] B. J.-D. F. Tabakin, "Qcmpi: A parallel environment for quantum computing," *Computer Physics Communications*, vol. 180, p. 948?964, 2009.
- [27] S. Caraiman, A. Archip, and V. Manta, "A grid enabled quantum computer simulator," in *Proceedings of the 2009 11th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, ser. SYNASC '09. IEEE Computer Society, 2009, pp. 189–196.
- [28] S. Caraiman and V. Manta, "Parallel simulation of quantum search," *International Journal of Computers Communications and Control*, vol. 5, no. 5, pp. 634–641, 2010.
- [29] G. Patz, "A parallel environment for simulating quantum computation," Master's thesis, Massachusetts Institute of Technology, 2003.
- [30] E. Gutierrez, S. Romero, M. Trenas, and E. Zapata, "Parallel quantum computer simulation on the cuda architecture," in *Computational Science - ICCS 2008*. Berlin: Springer, 2008, pp. 700–7709.
- [31] I. Glendinning and B. mer, "Parallelization of the qc-lib quantum computer simulator library," *Lecture Notes in Computer Science*, vol. 3019, pp. 461–468, 2004.
- [32] B. Ömer, "Simulation of quantum computers," 1996, <http://tph.tuwien.ac.at/oemer/doc/qcsim.ps>.
- [33] S. Arustei and V. Manta, "Qcl implementation of the bernstein-vazirani algorithm," *Buletinul Institutului Politehnic din Iasi, Automatic Control and Computer Science Section*, vol. LVIII, no. LIII, pp. 23–32, 2008.

Linear diffusion for generating plant distribution in natural environment simulation

Bogdan Apostol and Vasile Manta

Abstract—Seamless simulation and rendering of realistic natural environments has the goals of modeling the terrain and placing large groups of plants in a manner that would generate a scene similar to what a viewer would see in real life. The placement of plants is derived automatically from real land cover classification (LCC) data which describes each area on the map as a single vegetation type. This work presents a method for solving the problem of plant distribution in transition areas between different vegetation classes by using the concept of linear diffusion in an arbitrary direction. In practice this direction can be determined by different climatic factors. Our proposed implementation can accurately process a high number of neighboring areas in real-time by using parallel processing on the graphics processing unit.

I. INTRODUCTION

Virtual environments and three-dimensional visualization are essential aspects for computer graphics applications that try to simulate a real or imaginary world. The rendering of natural scenes has interested the scientific community due to its significant contributions in computer assisted analysis. In areas such as educational system and topographic analysis the three-dimensional exploration of geographical information is often critical for a better understanding of the overall problem.

The issue of natural environment simulation can be divided into the problem of generating a realistic terrain model and the problem of distributing the plants from an ecosystem on top of this model. In the field of artificial terrain generation the scientific community has focused on the acceleration of large land rendering in an acceptable number of frames per second.

A wide range of techniques have been proposed to solve the problem of generating artificial terrain [1] with or without a priori geometrical information. These include the technique based on fractals implementation [2] which is the most common method for generating artificial terrain, especially in video games, and detail reduction techniques such as Level Of Detail (LOD) and Chunked Level Of Detail (CLOD) [3]. Less complex methods allow the generation of a small number of terrain types that lack local surface characteristics. To achieve a natural look a secondary algorithm [4] usually needs to be applied.

Manuscript received June 18, 2011.

B. Apostol is with the Faculty of Automatic Control and Computer Engineering, Technical University "Gheorghe Asachi" Iasi, Str. Prof. dr. doc. Dimitrie Mangeron, nr. 27, Iasi, cod 700050, Romania apostolbgdn@yahoo.com

V. Manta is with the Faculty of Automatic Control and Computer Engineering, Technical University "Gheorghe Asachi" Iasi, Str. Prof. dr. doc. Dimitrie Mangeron, nr. 27, Iasi, cod 700050, Romania vmanta@cs.tuiasi.ro

Ecosystem simulation presents a real problem because of the large number of plants that need to be rendered and due to the increased amount of memory needed for storing and loading them. Given the variety of natural environments, solving the problem of positioning plants in an area of land is generally difficult, and the location of each plant is rarely known. However, the exact location of a single plant does not add realism to the environment being visualized, as what is important for an ecotype are the group characteristics such as the coverage area and density, which are characteristics that remain constant.

In this paper, we generate improved natural environments by using altitude information for modeling the terrain and vegetation information for placing plants in the ecosystem. We turn our attention mainly to plant distribution in transition areas between different vegetation classes, where sudden changes in plant models make the scene look unnatural. The all or nothing nature of LCC data make each area on the field have only a single LCC class, without the possibility of overlapping between classes. An example of the LCC data taken from the National Land Cover Dataset 2001 provided by the online service USGS Seamless Data Warehouse (<http://seamless.usgs.gov/>), where each pixel in the GeoTIFF image represents a surface of 1 arc-seconds (about 30 meters) length with a single vegetation class, is shown in figure (Fig.1).

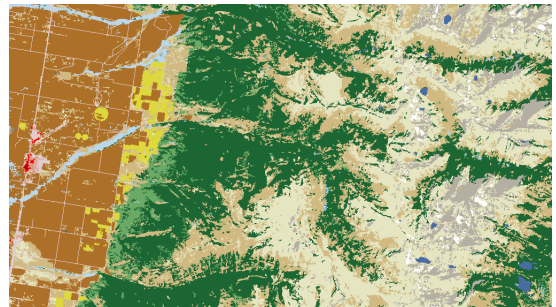


Fig. 1. Area from NLCD 2001 dataset.

This problem is also addressed in [5] where probability map images are generated for each LCC class and then smoothed using the "third nearest neighbor" weighting scheme to create the transition effect. However this generates an area partial class and does not solve the problem of a smooth transition at subarea level.

We solve this problem by using the concept of linear diffusion for each area represented by a LCC class. The direction of the diffusion can be arbitrary and can also be

chosen based on various local factors like terrain slope and slope orientation, based on prevailing wind direction or direct exposure to sunlight. We then split the area in different concentration subareas and generate random positions for plants within each area. This makes a smooth transition at a finer level that can be influenced by any given ecosystem parameter. For real-time rendering of the virtual environment we chose to partially implement the rendering process on the graphics processing unit and also make all necessary calculations for determining the values of plant distribution and final positions.

The outline of this paper is as follows: Section II describes the use of linear diffusion in the process of vegetation placement; Section III explains the details of the plant distribution algorithm; Section IV describes the parallel implementation of the diffusion algorithm on the graphics processing unit. Section V outlines the experimental results of the method and Section VI concludes with a summary and discussion of further development possibilities.

II. LINEAR DIFFUSION

The physical process of diffusion describes the spread of particles through random motions without creating or destroying mass. For the phenomenon to occur, there must be a difference in concentration (*concentration gradient*) between the two regions. The diffusion process takes place from the region of higher concentration to the region of smaller concentrations, in the direction of the concentration decrease.

For simplicity we accept that the diffusion takes place only in one direction, Ox . Volume concentration of particles, n , will then be only a function of x :

$$n = n(x). \quad (1)$$

The diffusion will occur if on the Ox axis there is a concentration gradient, in other words $\nabla n = \frac{\partial n}{\partial x} \neq 0$.

The diffusion phenomenon is characterized by a physical measuring unit, j_N , called the *flow particle number density*. This is numerically equal to the number of particles in the unit time passing through the unit area that is perpendicular on the direction in which the diffusion occurs. If we consider an area dS , that is normal to the diffusion direction, then in a timespan of dt the number of particles that will pass through the area is given by:

$$dN = j_N dS dt \quad (2)$$

The equilibration property in (3) is expressed by *Ficks law*:

$$j_N = -D \nabla n \quad (3)$$

where D is the *diffusion tensor* and j_N and ∇n are parallel or *isotropic*. The “-” sign shows that the diffusion takes place in the direction of the concentration decrease.

A particular case of diffusion, important from the practical point of view, is the *stationary diffusion*. This is characterized by the fact that volumetric concentration n has the same

value at a given fluid point at any point in time. This is equivalent to $\frac{\partial n}{\partial t} = 0$.

and from (2) after a first integration:

$$\frac{\partial n}{\partial x} = -\frac{j_N}{D} = 0 \quad (4)$$

Integrating (4) again we get:

$$\int_{n_0}^n dn = -\frac{j_N}{D} \int_x^D dx \quad (5)$$

and using (3)

$$n(x) = n_0 - \frac{j_N}{D} x = n_0 + x \nabla n \quad (6)$$

For each pair of neighboring surfaces that have different vegetation classes, and thus different plant concentrations, we consider a transition surface that has the area equal to the area of one of the surfaces. This is shown in figure (Fig.2) where n_1 is the concentration of the left surface, n_2 is the concentration of the right surface and the transition area is represented by the green rectangle.

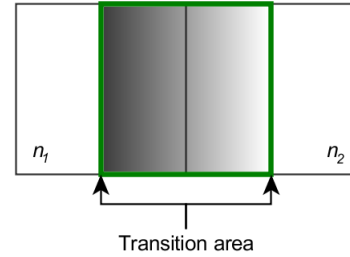


Fig. 2. Transition area for adjacent surfaces with n_1 and n_2 concentrations and concentration decreasing from n_1 towards 0.

We have two diffusion directions for each transition surface considered, one that will start from n_1 towards 0 and one that will start from n_2 towards 0. For simplicity we will take the general case of n_i to 0 diffusion where the diffusion direction is oriented and $i \in \{1, 2\}$. Using (6) on an arbitrary direction we obtain:

$$n_i(x) = n_i(0) \left(1 - \frac{x}{L}\right) \quad (7)$$

where L is the maximum distance in the diffusion direction.

This is depicted in figure (Fig.3) where an arbitrary direction is chosen for the dispersion and P_i is the intersection on the transition area border.

Thus the concentration in 0 is $n_i(0) = n_i$ and the concentration in P_i is $n_i(L) = 0$, where L is the length of the segment $[0, P_i]$. The concentration in an arbitrary point on $\mathbf{P}_i$ can then be calculated using (7).

III. PLANT DISTRIBUTION

This section will present the implementation details for the process of finding the position and number of plants in transition areas.

For the simplicity of finding the diffusion in any intermediate point on $\mathbf{P}_i$ and as we will see later on for minimizing

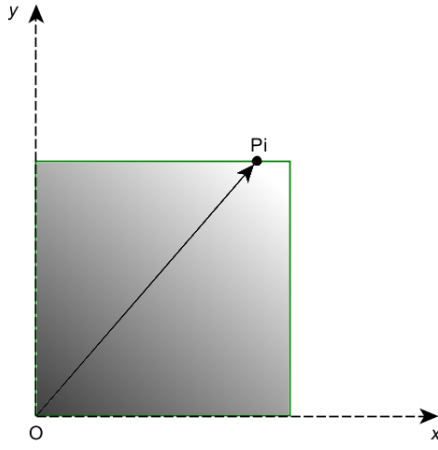


Fig. 3. Direction of the decrease of concentration in a transition area.

the number of false positives in the position generation, we consider a rotated coordinate system $x'Oy'$ where the direction of Ox' is the same as for P_i . This is exemplified in (Fig.4)

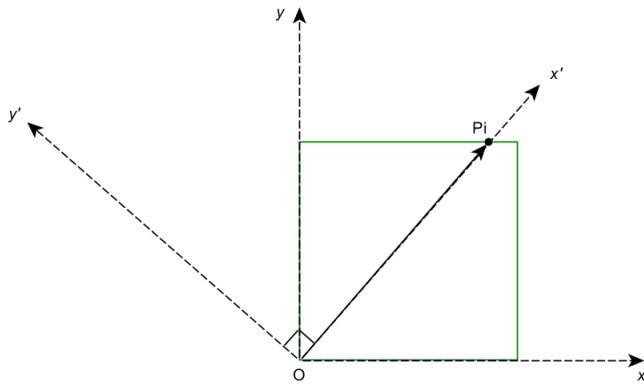


Fig. 4. xOy and rotated $x'Oy'$ coordinate systems.

We then split the segment $[0, P_i]$ in a number of m equal intervals, like it is shown in figure (Fig.5). For each of this intervals we define subsurfaces of the transition area between two vegetation types by tracing perpendiculars for each interval limit.

For each polygon thus determined we assign a concentration for that surface that is calculated in the middle of each interval using (7). Considering (2) we determine the number of plants instances that we need for a singular subsurface of the transition area:

$$N(x) = n_i(x)S \quad (8)$$

where $N(x)$ is the number of plants in the polygon centered in x , $n_i(x)$ is the concentration in that point and S is the area of the polygon. The area S of each polygon is easy to compute as the polygons are either triangles or can be split in trapezoids.

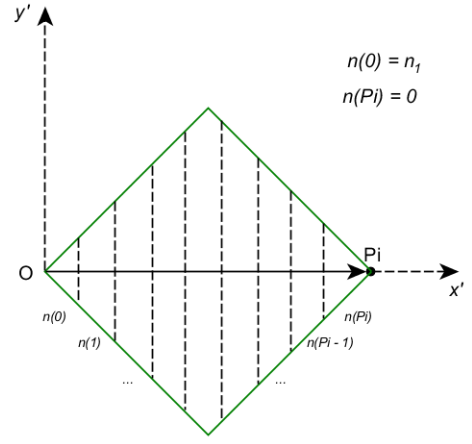


Fig. 5. Subsurfaces of the transition area.

Let us take a single complex polygon that the split generates and analyze the way in which we can generate a number of $N(x)$ random positions within the boundaries of the polygon. For this we introduce the concept of *Minimum Bounding Rectangle*, also known as bounding box or envelope, which is an expression of the maximum extents of a two-dimensional object within its bidimensional (x, y) coordinate system. This bounding box is determined by finding $\min(x)$, $\max(x)$, $\min(y)$, $\max(y)$ of the polygon considered as shown in figure (Fig.6).

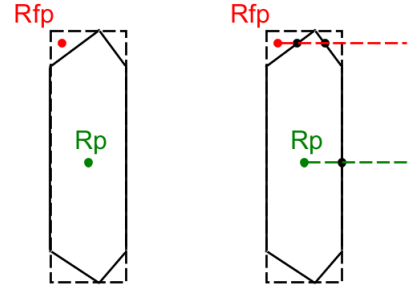


Fig. 6. Random point in polygon generation method.

We generate the random positions by a simple three steps algorithm:

- generate a bounding box for each polygon;
- generate random positions within the bounding box;
- test whether this point is in the interior of the polygon surface.

The second step introduces false positives, because we generate random points in a rectangle that can be inside the bounded polygon or outside of it. In figure (Fig.6) we show a false positive point in red and an inside point in green. Because of the random sampling nature of the second step, each generated point must be tested in the third step and if the point is outside resample the position until the test passes.

The performance characteristics of this random generation increases as the area where false positive might occur is

smaller. By rotating the $x'Oy'$ coordinate system we have already reduced the potential false positive area, as we have two polygon lines that are parallel and are also part of the bounding box limits. Without this rotation the result would be an extended false positive area as the polygon would be almost diagonal.

We test if a given point in the bounding box lies inside or outside the polygon by using a *point-in-polygon (PIP)* problem.

One simple way of finding whether the point is inside or outside the polygon is to test how many times a ray starting from the point and parallel to the Ox' axes intersects the edges of the polygon. If the random generated point is outside then the number of intersections is an even number, and if the point lies inside the polygon then the number of intersections is an odd number. We exemplify this in figure (Fig.6) for both an inside and outside point where the number of intersection points can be one or two. This algorithm is sometimes also known as the crossing number algorithm or the even-odd rule algorithm.

For each point generated in the $x'Oy'$ coordinate system that is not a false positive we find the corresponding coordinates in the xOy coordinate system by resolving (9)

$$R'(\theta) = R(-\theta) = [R(\theta)]^{-1} \quad (9)$$

where θ is the angle between the Ox and Ox' axes and $R'(\theta)$ is defined by the matrix in (10)

$$R'(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (10)$$

This points are then filtered based on parameters of altitude, relative altitude, slope and slope orientation, as defined in [6]. Similar to [5], for each random generated initial position of a plant we define a map of probabilities based on the above influencing parameters.

The problem of competition for natural resources between members of a vegetation class is treated using an evolutionary algorithm [7], [8], [9]. This algorithm uses a local-global approach L-systems to define the interaction between plants. We set a few simple rules: each plant is represented by a growing circle, plants develop at the same pace, a plant more adapted will dominate a less adapted plant and a plant is considered to have reached maturity when it has reached the maximum size for that vegetation class. This last step determines the size of the plants to be rendered in that area.

IV. PARALLEL COMPUTATION

Real-time rendering of densely populated areas of vegetation is one the true challenge for computer graphics and virtual reality systems. The complexity of the plant models and the number of plants included in a single scene exceed the rendering computational capacity, memory consumption of common hardware systems.

In this paper we use a simplification technique [10], [11] to increase the speed of rendering large communities of

plants and partial implementation of the rendering process in parallel on the graphics processing unit.

To maintain the impression of movement to the human eye it is necessary to generate at least 20 images per second. An image consists of about one million pixels, which means that for each pixel position and color need to be calculated in less than 50 nano-seconds. In such a short time the system processor (CPU) can run only a dozen instructions and this is not enough to solve complex problems. Also identifying each visible point as a pixel requires processing of the whole scene, which in our case consists of thousands of objects.

The computational power needed can be provided by parallel hardware that is specifically dedicated to the particular problem of rendering. This hardware is the graphics processor unit (GPU), which contains a large number of states of the execution flow and which breaks this flow in several parallel branches that are independent [12]. The result is a parallel architecture which can in terms of floating point process ten times the equivalent of current CPU processors.

Parallel GPU execution is effective only if the interdependency of the elements processed is minimal. In our case we can split the overall load of plants in equal areas that have a single LCC class value and can be processed in parallel because of their independence. This is also the case of texturing the terrain polygons.

We can interfere in the rendering execution stream in two programmable stages that are called *vertex shader* and *fragment/pixel shader*. A graphics card has an input buffer consisting of vertices and an output buffer consisting of colors, but implementing the shader stages we can use the graphics processing unit as a general purpose processing unit which can process a high number of operations.

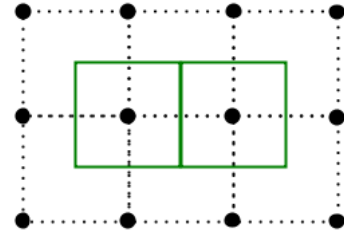


Fig. 7. Top view of transition areas considered around each vertex of the terrain model grid.

Thus we send for each of the terrain vertices, which are organized in an equally spaced grid, additional information about the surrounding area as it is shown in figure (Fig.7). This area can be considered as the transition area and thus for each vertex we have to send two LCC class values, n_1 and n_2 concentrations and the direction that the diffusion will take in consideration.

We use (9) to first rotate the local coordinate system of each transition area to match the direction of the diffusion. Then using (7) and (8) we calculate the partial concentrations for the considered area and split it in subareas. We generate the random point inside each subsurface based on the partial concentrations and then rotate using (9) again to find out

the final position of each plant. Finally we place a billboard [11] for each generated position and size it with the growing algorithm.

For this complex calculations the implementation was done in the vertex shader and the input data was given as a custom format vertex type that contains all the necessary data for that transition area. Textures for plants and for the ground are also sent to the GPU memory for processing. The billboard shader will generate four vertices for each plant position, thus generating the geometry needed for the pixel shader.

V. EXPERIMENTAL RESULTS

The implementation described in this paper has been tested using raster files in GeoTIFF format supplied by the U.S. Geological Survey (USGS) organization for the elevation model (National Elevation Dataset) and for the vegetation model (National Land Cover Dataset). These files contain information about the altitude of the ground and the vegetation cover of the geographically referenced land.

This section will present some of the qualities of the visualization system that we implemented, such as the ability to consistently recreate the same ecosystem and the more natural look of transition between different vegetation types. We will also make a study of the performance obtained by the algorithm in real-time on a standard hardware system, using only in GPU parallel implementation of plant positions diffusion. Finally we make a comparative study that shows how varying environmental parameters like density threshold limits and wind/sun direction affects the natural rendered environment.

For the GPU programming we use a high-level programming language. Of these languages the High Level Shader Language (HLSL) for Direct3D and C for Graphics (Cg) for OpenGL have become the most popular [13]. With these languages we can change the internal implementation of the rendering process on the graphics processor using its parallel processing power. We choose a solution based on HLSL programming as we also propose an implementation based on XNA3.1 Framework which can import HLSL language directly as effects.

In the first step we present a qualitative evaluation for the method of generating the terrain model and for the vegetation distribution based on the NED and NLCD data. Two locations were established for a fixed observer and for each observer's location a view direction was set. We generated the environment four times in a row without varying any parameter. It has been observed that the overall characteristics of the land were preserved (the shape and the texturing of the terrain) while local details change (the individual position of each plant changes) but the rendered plant density is constant. Furthermore, the general vegetation classes limits were kept between these renderings.

In the second step of evaluating the algorithm we made a comparative study between using the diffusion in transition areas and not using it. Figure (Fig.8) and figure (Fig.9) show the results obtained by using the diffusion for plant

placement and the occurrence of transition areas. The top images exemplify a rendering using only the LLC data and the bottom exemplify the use of the diffusion algorithm to generate a more natural feel.

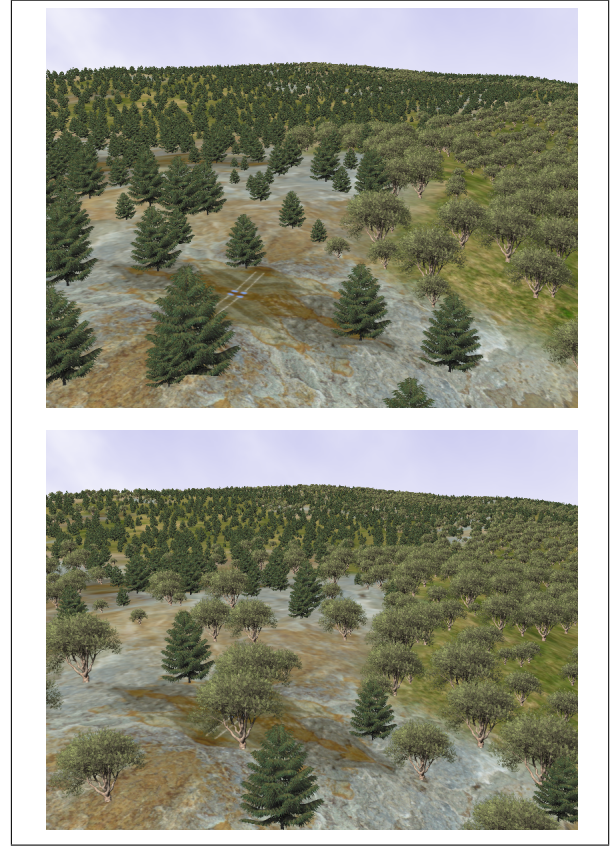


Fig. 8. Transition area without diffusion (top) and with diffusion (bottom).

We will present below a set of experimental results of the evaluation processing time during program execution for the implementation proposed by us. Results are obtained by running the program on a standard hardware system and using input data in GeoTIFF format with size between 60MB-90MB previously downloaded and stored to disc. The raster space resolution used as input altitude data is of about 9000x5000 pixels and the resolution of the raster data used as input information for the vegetation classes is of about 5000x4000 pixels. All images used for rendering plant billboards are in RGBA format and all images used for texturing are in RGB format.

TABLE I
THE SPEED IN FRAMES PER SECOND INFLUENCED BY THE NUMBER OF
RENDERED OBJECTS

Rendered object number	Rendered vertices number	FPS
147287	1133722	10
71365	557790	20
10702	131812	36
2708	41848	60

Table I is a qualitative assessment of the scene rendering speed depending on the number of vegetation objects placed

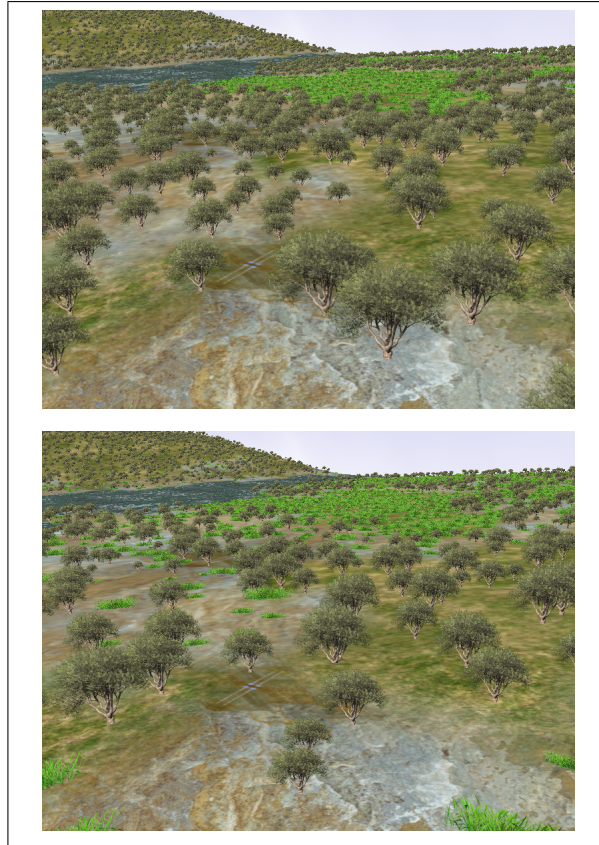


Fig. 9. Transition area without diffusion (top) and with diffusion (bottom).

in the scene. For these tests the scene dimensions were kept constant and plant concentration was varied. Also for this tests the viewing window and also the resolution were kept the same size. We will measure the rendering speed of our implementation depending on the maximum number of frames per second (FPS) that can be rendered by the graphics card. We mention that for the human eye to perceive a sequence of images as being continuous it needs to reach a minimum of 20 FPS. This shows a decrease in speed as we increase the number of plants to be rendered. Each rendered plant uses a set of six vertices.

In general, the assessment of a scene is done depending on the total number of vertices and because of this we added column for the number of vertices besides the number of objects. It can be easily observed that the number of vertices is greater than three times the number of vertices because besides the vertices used for plants rendering there are also the number of vertices used for rendering the terrain surface. Note that the last row in Table I determines a number of frames per second less than the imposed limit of 20fps, hence we conclude that a high plant concentration require reducing the visualized land surface.

In terms of CPU performance and system memory load we included a report on the resources load depending on the number of objects in the scene and according to motion versus static. Memory consumption and processor load are calculated by making an average of three consecutive runnings of the algorithm and noted in Table II. The average

time for preloading resources is 4.67 seconds and the data acquisition was achieved after the application has loaded.

TABLE II
MEMORY CONSUMPTION AND PROCESSOR LOAD IN RESPECT TO THE
NUMBER OF OBJECTS IN THE SCENE

Rendered object number	Memory load	CPU load
243091	508,924K	0.390
147287	409,820K	0.267
82498	314,984K	0.213

VI. CONCLUSIONS AND FUTURE WORKS

In this paper, we have proposed a method for solving the problem of plant distribution in transition areas between different vegetation classes by using the concept of linear diffusion in an arbitrary direction. The type of plants to be rendered in each area are derived automatically from real LCC data which describes each area on the map as a single vegetation type. For each pair of neighboring LCC areas that have different vegetation classes and thus different plant models we consider a transition area with two directions for the direction of the concentration decrease. This direction can be chosen a priori or can be derived from climatic factors, such as predominant wind direction or sunlight direction, or from local factors, such as slope, slope orientation or altitude. However each vegetation class has a predefined ability to withstand the steep slopes, or cannot be present over a certain altitude limit. To process a high number of neighboring areas we have implemented the diffusion process and vegetation placement using parallel processing on the graphics processing unit. Thus, we obtain an enhanced natural environment simulation that can be that can easily be rendered with more than 20 frames per second using a standard hardware system. Further algorithm development primarily involves varying the diffusion tensor in dependency to each vegetation characteristic proprieties and using a non-linear diffusion model for transition areas.

REFERENCES

- [1] Grumet Matthias. Terrain modeling. Unpublished computer science report, Institute of Computer Graphics and Algorithms, Vienna University of Technology, 2004.
- [2] Miguel Frade, F. Fernandez de Vega, and Carlos Cotta. Genetic terrain programming - an aesthetic approach to terrain generation. In *Computer Games and Allied Technology 08*, pages 1-8, Singapore, 2008.
- [3] A. Balogh. Real-Time Visualization of Detailed Terrain. Unpublished computer science masters thesis, Budapest University of Technology and Economics, Budapest, Hungary, 2003.
- [4] Miguel Frade, F. Fernandez de Vega, and Carlos Cotta. Breeding terrains with genetic terrain programming: The evolution of terrain generators. In *International Journal of Computer Games Technology Volume 2009*, Singapore, 2009.
- [5] William D. Wells. Generating enhanced natural environments and terrain for interactive combat simulations (genetics). In *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST 05, pages 184-191, New York, NY, USA, 2005. ACM.
- [6] Johan Hammes. Modeling of ecosystems as a data source for real-time terrain rendering. In *Proceedings of the First International Symposium on Digital Earth Moving*, DEM 01, pages 98-111, London, UK, 2001. Springer-Verlag.

- [7] Brendan Lane and Przemyslaw Prusinkiewicz. Generating spatial distributions for multilevel models of plant communities. In *Proceedings of Graphics Interface*, pages 69-80, Calgary, Alberta, May 27-29, 2002.
- [8] Oliver Deussen, Pat Hanrahan, Bernd Lintermann, Radomir Mech, Matt Pharr, and Przemyslaw Prusinkiewicz. Realistic modeling and rendering of plant ecosystems. *Proceedings of SIGGRAPH 98* (Orlando, Florida, July 19-24, 1998). In *Computer Graphics Proceedings, Annual Conference Series*, 1998, ACM SIGGRAPH, pp. 275–286.
- [9] M. Alsweis, O. Deussen. Modeling and visualization of symmetric and asymmetric plant competition. *Natural Phenomena 2005*, page 83, AK Peters, Ltd. 2005
- [10] Yi-Kuan Zhang, Olivier Teboul, Xiao-Peng Zhang, and Qing-Qiong Deng. Image based real-time and realistic forest rendering and forest growth simulation. In *Proceedings of the 2006 International Symposium on Plant Growth Modeling, Simulation, Visualization and Applications*, pages 323-327, Washington, DC, USA, 2006. IEEE Computer Society.
- [11] Guanbo Bao, Xiaopeng Zhang, Wujun Che, and Marc Jaeger. Billboards for tree simplification and real-time forest rendering. In *Proceedings of the 2009 Plant Growth Modeling, Simulation, Visualization, and Applications*, PMA 09, pages 433-440, Washington, DC, USA, 2009. IEEE Computer Society.
- [12] László Szirmay-Kalos, László Szécsi, and Mateu Sbert. *GPU-Based Techniques for Global Illumination Effects*. Synthesis Lectures on Computer Graphics and Animation. Morgan Claypool Publishers, 2008.
- [13] Sebastien St-Laurent. *The COMPLETE Effect and HLSL Guide*. Paradoxal Press, 2005.

The Fast Itemset Miner: A Detailed Analysis of the Candidate Generation Stages

Alexandru Archip and Mitică Craus

Abstract—Association Rule Mining (ARM) represents an important tool for Data Mining techniques. A key step in ARM is to determine the frequent itemsets present within the analyzed data. Recent algorithms addressing this problem focus on identifying frequent itemsets without candidate generation steps. This paper details a new algorithm – Fast Itemset Miner (FIM for short) – that still relies on candidate generation stages, but has a different approach from the standard Apriori algorithm. This approach focuses on increasing size of the interval that candidates belong to, rather than increasing the size of the candidates by one with each corresponding iteration. By comparing the Apriori and FIM candidate generation stages, we show that this second approach uses a faster and more efficient method of determining candidate itemsets. Moreover, such an approach also favors a better support counting method, which greatly impacts on the overall time response of the FIM algorithm.

Index Terms—Data Mining, Association Rule Discovery, Frequent Itemset Mining, Candidate Based Algorithms

I. INTRODUCTION

Association Rule Mining (ARM) refers the problem of identifying all relations between the elements of a given data set so that the presence of a given element/set of elements implies the presence of another element/set of elements [1], [2], [3], [4]. This process usually involves two distinct stages: the first one involves determining all frequent itemsets present within the collection of transactions, while the second stage aims to find strong association rules based on the previously identified itemsets [1], [5]. The first stage is generally considered to be the most important one with respect to the overall efficiency of the ARM algorithms.

The first notable algorithm to efficiently solve the issue of frequent itemset mining is the Apriori algorithm (1994) [6], [1], [5]. The algorithm performs multiple scans over the target transaction database as follows. The first such scan determines the frequent items with respect to the minimum support limit given by the user. All subsequent scans involve two stages: the first one – the so called *join stage* – is to identify all possible k -itemset candidates based on the frequent $(k-1)$ -itemsets found in the corresponding previous scan; the support for each candidate is then computed in the second stage and all candidates that do not meet the minimum support value are removed – this stage is known as the *pruning stage*. The key aspect of this algorithm is

the efficient generation of candidates. An optimal approach must consider the *Apriori Property* [6]: *If an itemset is frequent, then all of its subsets must also be frequent*. Should this restriction be met, Agrawal et al. state that the Apriori algorithm achieves a linear time complexity with respect to the size of the transaction database [6].

Further analyses performed using the Apriori algorithm have shown that its main advantage of counting the support value only for the determined candidates could also prove to be its greatest weakness. The amount of candidate itemsets that is determined in the *join step* could be quite large and this could negatively influence the overall performance [7], [4]. Moreover, in such cases, methods that offer an efficient candidate selection may consume considerable more resources [1], [5].

Taking this into consideration, ulterior attempts to develop better and faster algorithms follow two different approaches. The first one is to completely eliminate the candidate itemsets. One such remarkable algorithm is FP-Growth (first presented in [7] and revised in [8]). The algorithm builds prefix trees from the target transaction database by performing exactly two scans of the database: one to find the frequent items and one to actually build the prefix tree. In order to determine all frequent itemsets, the prefix tree is traversed in a bottom-up manner [7], [8]. A much more recent algorithm – called And Code (AC for short) – is presented in [4]. The main idea behind AC is to binary code itemsets and transactions.

The second line of approaches to the itemset mining problem has been to improve the *standard* candidate generation techniques for the Apriori algorithm. Attempts have been made to reduce the overall number of I/O operations involved [9] or to use different data structures in order to represent the itemsets and the candidates [10].

It can be noticed that almost no effort had been directed to investigating other means of generating candidate itemsets. The present paper details a new method for identifying valid candidates for a relatively new algorithm – Fast Itemset Miner (FIM for short) [3], [2]. In our previous work (papers [2] and [3]) little or no detail is given for the candidate generation stages used in FIM. The main goal of this paper is to show how simple and efficient is this new method of determining valid candidates. Section II describes the base sequential algorithm and the simple candidate generation stages for this algorithm. This section also includes a proof of correctness for this particular method of identifying candidate itemsets. Section III includes a presentation of the generalized form for both the algorithm and the candidate

A. Archip is with The Faculty of Automatic Control and Computer Engineering, The “Gheorghe Asachi” Technical University of Iași alexandru.archip@cs.tuiasi.ro

M. Craus is with The Faculty of Automatic Control and Computer Engineering, The “Gheorghe Asachi” Technical University of Iași craus@tuiasi.ro

generation and a few details regarding the parallel version of the algorithm. Section IV will present a first set of comparative time results between FIM and Apriori, while the last section of the paper is dedicated to some final remarks and conclusions, as well as a few details on how this new algorithm could evolve.

II. THE BASE FIM ALGORITHM AND CANDIDATE GENERATION METHOD

The fundamental idea behind FIM is to determine new frequent itemsets by gradually increasing the size of the interval that includes the items contained within an itemset [2]. Let n be the total number of frequent items. The algorithm begins with n intervals, each such interval including a single frequent item. At a given stage k , new candidates will be generated from items belonging to the interval $[i, i+k]$ (having $i \in \{1, 2, \dots, n-k\}$), while for the subsequent stage $k+1$ new candidates will be determined by combining items belonging to $[i, i+k+1]$ (having $i \in \{1, 2, \dots, n-k-1\}$). The following notations will be used [3], [2]:

$$\begin{aligned} F_{i,j} &\text{--the set of all frequent itemsets that include} \\ &\text{items belonging to interval } [i, j] \\ C_{i,j} &\text{--the set of all candidates that include} \\ &\text{items belonging to interval } [i, j] \end{aligned} \quad (1)$$

In [2] it is shown that new frequent itemsets in $F_{i,j}$ contain both items i and j . As a direct consequence, the new candidates for interval $[i, j]$ must contain both items i and j . Therefore, new candidates will be generated according to equation (2) [2], [3].

$$C_{i,j} = \{X \cup Y \mid (X \in F_{i,j-1} \wedge i \in X) \wedge (Y \in F_{i+1,j} \wedge j \in X)\} \quad (2)$$

The base FIM algorithm is given in Algorithm 1 [2], [3].

Algorithm 1 works as follows. A preprocessing stage is required to determine all frequent items, trim transactions by discarding non-frequent items and sort the new transaction dataset descending with respect to an items support threshold. Assuming n frequent items, in the first stage of the algorithm, new candidates will be generated based on the following subsets: $\{1, 2\}, \{2, 3\}, \dots, \{n-1, n\}$. For the next stage, candidates will be determined from items belonging to $\{1, 2, 3\}, \{2, 3, 4\}, \dots, \{n-2, n-1, n\}$. This same procedure continues until, finally, new candidate itemsets will include items from the interval $\{1, 2, \dots, n\}$. One of the key advantages of this approach is that the FIM algorithm does not need to scan the whole transaction database in order to determine the support for any given valid candidate. In order to determine the support for all candidates included in $C_{i,j}$, the algorithm must scan only the transactions that include item i .

Since the FIM algorithm is a candidate based approach to the frequent itemset mining problem, a key issue that needs to be solved is to efficiently determine the new candidates for a given stage (line 4 in Algorithm 1). In order to better

Algorithm 1 Fast Itemset Miner Base Algorithm

Require: $F_{i,j}$ = set of frequent itemsets belonging to $[i, j]$
Require: D = transaction set, $t \subset D$

```

1: for  $k = 1$  to  $n - 1$  do
2:   for all  $i, j : i \in 1, \dots, n - k; j = i + k$  do
3:      $F_{i,j} \leftarrow F_{i,j-1} \cup F_{i+1,j}$ 
4:      $C_{i,j} = \{X \cup Y \mid (X \in F_{i,j-1} \wedge i \in X) \wedge (Y \in F_{i+1,j} \wedge j \in X)\}$ 
5:     for all transactions  $t \subset D_i$  do
6:        $\{D_i$  denotes the subset of transactions in  $D$  that include frequent item  $i\}$ 
7:        $C_t \leftarrow \text{subset}(C_{i,j}, t)$  {determine all candidates in  $C_{i,j}$  that are include in transaction  $t$ }
8:       for all candidates  $c \in C_t$  do
9:          $c.\text{support}++$ 
10:      end for
11:    end for
12:     $F_{i,j} = F_{i,j} \cup \{c \in C_{i,j} \mid c.\text{support} \geq \text{minsupp}\}$ 
13:     $F_k \leftarrow F_k \cup F_{i,j}$ 
14:  end for
15: end for
```

understand the solution we submit for this step of the base FIM algorithm, the following notations will be used for an itemset $X \in F_{i,j}$:

$$\begin{aligned} &\text{the maximal itemset } N \text{ such that } N \subset X \text{ and} \\ &N \cap \{i\} = N \cap \{j\} = \emptyset - \text{the core itemset of } X \end{aligned} \quad (3)$$

$$\begin{aligned} &\text{suffix}(X) = X \setminus \{i\} - \text{the suffix itemset of } X \\ &\text{prefix}(X) = X \setminus \{j\} - \text{the prefix itemset of } X. \end{aligned} \quad (4)$$

Also, our solution relies on the following lemma, which is a direct consequence of the *Apriori Property* [6].

Lemma 1: Any given itemset X is not frequent if its core itemset (equation (3)) is not frequent.

Using Lemma 1 and considering line 4 in Algorithm 1, we can say that a candidate itemset is valid if its core is a frequent set. Moreover, each subset of a valid candidate $T \in C_{i,j}$ that does not simultaneously contain items i and j must be a frequent itemset. The new method we use to identify valid candidates is based on the following theoretical result.

Lemma 2: Let T be a candidate itemset in $C_{i,j}$. T is a valid candidate if and only if it is obtained through joining two frequent itemsets $X \in F_{i,j-1}$ and $Y \in F_{i+1,j}$ that simultaneously meet the following conditions:

$$i \in X \text{ \textbf{si} } j \in Y \quad (5)$$

$$\text{suffix}(X) = \text{prefix}(Y) \quad (6)$$

Proof: Lets assume that there is a valid candidate $T \in C_{i,j}$ that may be obtained from joining itemset $X \in F_{i,j-1}$ and $Y \in F_{i+1,j}$ such that:

$$i \in X \text{ \textbf{si} } j \in Y \quad (7)$$

$$\text{suffix}(X) \neq \text{prefix}(Y) \quad (8)$$

$$\text{and } T = X \cup Y \quad (9)$$

According to equation (3), the core itemset of T is:

$$N_T = T \setminus \{i, j\} \quad (10)$$

According to equations (10), (1) and (8), candidate T may be written as:

$$T = \{i\} \cup N_T \cup \{j\} \quad (11)$$

If T is a valid candidate, then its core N_T must be a frequent itemset (Lemma 1). Therefore, considering Lemma 1 and the *Apriori Property*, it results that:

$$\text{itemset } X' = \{i\} \cup N_T \text{ is frequent} \quad (12)$$

$$\text{itemset } Y' = N_T \cup \{j\} \text{ is frequent} \quad (13)$$

Considering equations (12), (13) and (1), it results that:

$$X' \in F_{i,j-1} \quad (14)$$

$$Y' \in F_{i+1,j} \quad (15)$$

and that:

$$T = X' \cup Y' \quad (16)$$

Furthermore, considering equations (12), (13) and (3) it results that:

$$\text{suffix}(X') = X' \setminus \{i\} = N_T = \text{prefix}(Y') = Y' \setminus \{j\} \quad (17)$$

Equation (17) contradicts the assumption of equation (8).

Reciprocal, let $X \in F_{i,j-1}$, $i \in X$ and $Y \in F_{i+1,j}$, $j \in Y$ that conform to equation (6). Let $T = X \cup Y \in C_{i,j}$ be the candidate itemset that may be obtained by joining itemsets X and Y . T may be written as:

$$T = \{i\} \cup \text{suffix}(X) \cup \{j\} = \{i\} \cup \text{prefix}(Y) \cup \{j\} \quad (18)$$

According to equations (3) and (4), the core itemset of T is N_T :

$$N_T = \text{suffix}(X) = \text{prefix}(Y) \quad (19)$$

Since X and Y are frequent itemsets, then all their subsets are frequent itemsets (*Apriori Principle* [6]). This implies that itemset N_T is frequent. Therefore, all subsets of candidate T that do not simultaneously include items i and j are frequent. Therefore T is a valid candidate. ■

Algorithm 2 depicts the new base method for generating all candidate itemsets in $C_{i,j}$ for the FIM base algorithm.

Algorithm 2 The candidate generation method for the FIM base algorithm

Require: $F_{i,j-1}$, $F_{i+1,j}$

```

1: for all itemset  $X \in F_{i,j-1}$  do
2:   for all itemset  $Y \in F_{i+1,j}$  do
3:     if  $\text{suffix}(X) = \text{prefix}(Y)$  then
4:       candidate  $\leftarrow X \cup Y$ 
5:       add candidate in  $C_{i,j}$ 
6:     end if
7:   end for
8: end for
```

In order to compare this new method against the standard Apriori method, we must first analyze the basic AprioriGen method given in Algorithm 3.

Algorithm 3 The standard Apriori candidate generation method

Require: L_{k-1} {the set of all frequent $k-1$ -itemsets}

```

1: for all pairs (s.a, s.b)  $\in L_{k-1} \times L_{k-1}$  such that  $a < b$  do
2:   candidate = s.a.b;
3:   if all  $k-1$  subsets of candidate are frequent then
4:     add candidate to candidate list
5:   end if
6: end for
```

Lets assume that the current stage for the Apriori algorithm is stage $k = 5$, i.e. all valid candidates of size $k = 5$ must be determined. Lets assume the target candidate is $\{1, 2, 3, 4, 5\}$. According to Algorithm 3, this candidate is obtained by joining itemsets $\{1, 2, 3, 4\}$ and $\{1, 2, 3, 5\}$. In order to validate this candidate, all its $k = 4$ subsets must be generated and all of them must be frequent 4-itemsets. Identifying all 4-itemsets include in $\{1, 2, 3, 4, 5\}$ requires 5 steps. Assuming n_4 to be the total number of frequent 4-itemsets, we can state that validating candidate $\{1, 2, 3, 4, 5\}$ requires $(5 \cdot \log n_4)$ steps to complete.

When analyzing our method for generating the same candidate (Algorithm 2), it can be observed that no supplemental steps are required for validating candidate $\{1, 2, 3, 4, 5\}$. This result is proven to be correct and is enforced by Lemma 2. This example clearly outlines the simplicity of our method when compared against the standard Apriori approach.

III. THE GENERAL FIM ALGORITHM AND CANDIDATE GENERATION METHOD

In the previous section, the base FIM algorithm has been briefly presented. As it can be observed from Algorithm 1, the simple model assumes that initial intervals include only a single frequent item. If we consider n to be the total number of frequent items, it is clear that FIM requires n stages to completely determine all frequent itemsets [2], [3]. This implies a total of n partial scans over the transaction database. The general model of the FIM algorithm aims to reduce this number of scans. In order to achieve this goal, it is necessary to group the frequent items in m larger intervals (obviously, $m < n$). The first stage of the generalized FIM algorithm implies applying Algorithm 1 over each interval of frequent items I_i ($1 \leq i \leq m$) in order to identify all frequent itemsets that belong to the target interval. The following notations will be used [3]:

F_{I_i, I_j} —the set of all frequent itemsets in

$$I_{i,j} = I_i \cup I_{i+1} \cup \dots \cup I_j \quad (20)$$

C_{I_i, I_j} —the set of all candidate itemsets in $I_{i,j}$

In [3] it is shown that a frequent itemset $X \in F_{I_i, I_j}$ must simultaneously include itemsets belonging to I_i and I_j . Therefore, new candidates belonging to C_{I_i, I_j} must simultaneously include itemsets belonging to I_i and I_j . This implies that new candidates are determined according to (21) (a general

approach to equation (2)):

$$C_{I_i, I_j} = \{X \cup Y | (X \in F_{I_i, I_{j-1}} \wedge I_i \cap X \neq \emptyset) \wedge (Y \in F_{I_{i+1}, I_j} \wedge I_j \cap Y \neq \emptyset)\} \quad (21)$$

The main difference from the base model is that the general FIM algorithm will consider for each stage a new interval I_i ($1 \leq i \leq m$) rather than a new frequent item i ($1 \leq i \leq n$). Algorithm 4 presents this general approach.

Algorithm 4 The general model for the FIM algorithm

Require: $F_{I_i, I_i} = \{\text{set of all frequent itemsets in } I_i\}$
Require: $D_i = \text{set of transactions grouped by intervals}$ (subsets of the original D transaction list that include items for interval I_i)

```

1: for  $k = 1$  to  $m - 1$  do
2:   for all  $I_i : i = 1, \dots, m - k$  do
3:      $I_j = I_{i+k}$ 
4:      $F_{I_i, I_j} \leftarrow F_{I_i, I_{j-1}} \cup F_{I_{i+1}, I_j}$ 
5:      $C_{I_i, I_j} = \{X \cup Y | (X \in F_{I_i, I_{j-1}} \wedge I_i \cap X \neq \emptyset) \wedge (Y \in F_{I_{i+1}, I_j} \wedge I_j \cap Y \neq \emptyset)\}$ 
6:     for all transactions  $t \in D_i$  do
7:        $C_t \leftarrow \text{subset}(C_{I_i, I_j}, t)$  {determine candidates in  $C_{I_i, I_j}$  included in  $t$ }
8:       for all candidates  $c \in C_t$  do
9:          $c.\text{support}++$ 
10:      end for
11:    end for
12:     $F_{I_i, I_j} = F_{I_i, I_j} \cup \{c \in C_{I_i, I_j} | c.\text{support} \geq \text{minsupp}\}$ 
13:  end for
14: end for
```

In order to efficiently generate all valid candidates for a single stage, we must adapt Algorithm 2. For a given itemset (either frequent or candidate) $X \in F_{I_i, I_j}$, the following notations will be used (the general form of equation (4)):

$$\begin{aligned} \text{suffix}_G(X) &= X \setminus \{k | k \in I_i\} \\ \text{prefix}_G(X) &= X \setminus \{k' | k' \in I_j\}. \end{aligned} \quad (22)$$

Using equation (22), the corresponding method for determining valid candidates represents a generalization of Algorithm 2 and is presented in Algorithm 5.

Algorithm 5 Generalized approach to the candidate generation stage for Algorithm 4

Require: $F_{I_i, I_{j-1}}, F_{I_{i+1}, I_j}$

```

1: for all itemset  $X \in F_{I_i, I_{j-1}}$  do
2:   for all itemset  $Y \in F_{I_{i+1}, I_j}$  do
3:     if  $\text{suffix}_G(X) = \text{prefix}_G(Y)$  then
4:       candidate  $\leftarrow X \cup Y$ 
5:       add candidate in  $C_{I_i, I_j}$ 
6:     end if
7:   end for
8: end for
```

Lemma 3 proves that Algorithm 5 generates all possible valid candidates.

Lemma 3: Let T be a candidate itemset in C_{I_i, I_j} . T is a valid candidate if and only if it is obtained through joining two frequent itemsets $X \in F_{I_i, I_{j-1}}$ and $Y \in F_{I_{i+1}, I_j}$ that simultaneously meet the following conditions:

$$X \setminus \text{suffix}_G(X) \subset I_i \quad (23)$$

$$Y \setminus \text{prefix}_G(Y) \subset I_j \quad (24)$$

$$\text{suffix}_G(X) = \text{prefix}_G(Y) \quad (25)$$

Lemma 3 represents the general form for Lemma 2, therefore its proof is similar.

The parallel model for the general FIM algorithm is straight forward since processing a set of intervals may be performed independently once the required values have been transferred accordingly. The initial number of intervals is equal to the number of parallel working units (either processes or threads, depending on the underlying parallel architecture). Each working unit would apply Algorithm 1 for its given interval. After local processing is finished, it can easily be observed the **for all** loop on line 2 in Algorithm 4 may be run in parallel. A detailed presentation of this parallel model is given in [3]. The important thing to notice is that this approach does not affect at all the candidate generation method presented in Algorithm 5. Once each active working unit receives its corresponding new data, no supplemental communications are required in order to determine all possible candidates and all computation is performed locally. This is a major advantage for this new algorithm over almost all parallelizations of the Apriori based approaches.

IV. PRELIMINARY TEST RESULTS

A preliminary series of tests have been performed in order to compare the overall performance of the FIM algorithm against the Apriori standard algorithm. For both algorithms a binary approach was used in order to code transactions and itemsets. Both algorithms were run on a synthetic dataset (T10I4D100K), having 100000 transactions and 1000 individual items. For the FIM algorithm, the number of intervals has been modified such that both Algorithm 1 and Algorithm 4 could be tested (with their corresponding candidate generation methods depicted in this paper). Table I presents the results we obtained for a minimum support limit of 500 transactions (resulting in a total of 569 frequent items).

The first important note is that FIM outperformed Apriori every time, even for the last test for FIM (1 interval with 569 items). This last test is extremely relevant, since in this particular case the FIM algorithm needed to fully scan the transaction database for each stage. Since the standard Apriori algorithm behaves in a similar manner, this last test outlines the efficiency of the candidate generation methods that have been presented in this paper (Algorithms 2 and 5): a difference of 800+ seconds. The second important note is that the minimum execution time for FIM is 517.19 seconds (60 intervals, with 9/10 items per interval – even split) – more than 5 times faster than Apriori. This result is explained

TABLE I

FIM VS. APRIORI - TRANSACTION SET *T10I4D100K*, MINIMUM SUPPORT
THRESHOLD 500 TRANSACTIONS

Apriori Algorithm		
Frequent items		Time (s)
569		2845.20
FIM Algorithm		
Interval count	number of frequent items per interval	Time (s)
569	1	2029.24
300	2	999.91
200	3	730.19
150	4	642.68
100	6	539.54
90	7	533.90
80	8	524.44
70	9	520.34
60	10	517.19
50	12	556.67
40	15	581.98
30	19	590.57
20	29	634.84
10	57	1025.86
1	569	2026.04

by both our methods of generating candidates and by the reduction in database scans explained in [3].

V. CONCLUSIONS AND FURTHER WORK

This paper presented new and efficient means of determining candidate itemsets for a particular algorithm – The Fast Itemset Miner (FIM). These methods exploit both the *Apriori Property* and the particularities of FIM algorithm to avoid any unnecessary candidate validation steps and to avoid generating too many candidates. Also, these methods do not impose any supplemental restrictions on the parallelization of the base and general FIM algorithms. Although limited tests have been performed until now, their results underline the potential of this novel approach to the frequent itemset mining problem.

Our immediate priority is to properly test this new approach, not only against Apriori, but also against faster algorithms such as FP-Growth. Favorable results would underline the fact that candidate based approaches should still be considered in addressing the frequent itemset mining problem. If properly handled, such approaches do have the advantage of eliminating unnecessary test cases. A second line of tests must address the parallel FIM algorithm. This new algorithm implies a simple, straight forward parallelization that may be easily adapted to either shared memory parallel machines, either message passing ones.

REFERENCES

- [1] M. J. Zaki, "Parallel and distributed association mining: A survey," *IEEE Concurrency*, vol. 7, no. 4, pp. 14 – 25, 1999.
- [2] M. Craus, "A new algorithm for association rule discovery," in *Proceedings of the 9th International Symposium on Automatic Control and Computer Science*, 2007.
- [3] M. Craus and A. Archip, "A generalized parallel algorithm for frequent itemset mining," in *ICCOMP'08: Proceedings of the 12th WSEAS international conference on Computers*. World Scientific and Engineering Academy and Society (WSEAS), 2008, pp. 520–523.
- [4] H. Zhou and J. Liu, "A novel algorithm for association rule mining without candidate," in *Proceedings of the 2009 International Joint Conference on Artificial Intelligence*, ser. JCAI '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 116–119.
- [5] J. Han and M. Kamber, *Data Mining - Concepts and Techniques*. Morgan Kaufman Publications, 2006.
- [6] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, ser. VLDB '94. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, pp. 487–499.
- [7] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *SIGMOD 00: Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000.
- [8] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *Data Min. Knowl. Discov.*, vol. 8, pp. 53–87, January 2004.
- [9] C. lap Yip, K. K. Loo, B. Kao, D. Cheung, and C. K. Cheng, "Lgen — a lattice-based candidate set generation algorithm for efficient association rule mining," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 1999, pp. 54–63.
- [10] N. Malhis, A. Ruttan, and H. H. Refai, "An efficient approach for candidate set generation," *Journal of Information & Knowledge Management (JIKM)*, vol. 4, pp. 287–291, 2005.

Multi-Threading System for Human Behavior Simulation based on Constraints Propagation

Ovidiu Aritioni and Viorel Negru

Abstract—In this paper, we propose a software prototype to simulate the human behavior in a domestic space. Our simulation takes into consideration the household structure and data about its residents lifestyle. The proposed system uses a multi-thread architecture, for each household's resident we deploy a single thread. The household resources management is based on the constraints propagation. We develop an ontology of activities of daily-living, and a special language based on this ontology: Human Behavior Scenarios Description Language. The context based reasoning is developed in JESS (Java Expert System Shell) language. The communication between threads is used in order to exchange information and for the constraints propagation.

Index Terms—constraints, human behavior, activities of daily living (ADL), human behavior, simulation.

I. INTRODUCTION

The European Commission goal in the energetic field is to reduce the CO_2 emissions by 2020 with more than 20% [2]. The CO_2 emissions due to the households represent approximately 30% of total, with an increasing trend [1]. The human behavior has a great impact on the household's energy consumption [5]. An energetic audit, about students who live in an university campus, proves that the same room has different levels of energy consumption which depend on the student behavior [8].

In this context the human behavior and the relation with energy consumption became an important research subject. Our paper present a simulator of human behavior with respect to electronic device usage primarily. Our simulation area is represented by houses, apartments and households in general. The simulation takes as input a lot of information about the household and the inhabitants and generates scenarios which describe the human behavior. To describe the human behavior we define a special language: Human Behavior Scenarios Description Language (HBSDL). Our simulator can be used for the energy consumption prediction or as a tool to help us to increase the energy efficiency [3].

Other models for human behavior [4], [6] take into consideration people in isolated spaces, without interaction with other persons, and in non-domestic spaces (offices, industry, etc). The simulations based on static profiles and on isolated residents are limited to be extended for the real-life. The 5W1H model [7] or the Brahms model [9]

This work was partially supported by the FP7 ICT project No 224609/1.06.2008 (DEHEMS) and by the Romanian projects: RO-PNII-20-EU/11.06.09 (DEHEMS-RO) and PNII nr.12-122/2008 (ASISTSYS).

Ovidiu Aritioni and Viorel Negru, Computer Science Department, Faculty of Mathematics and Computer Science, West University of Timișoara, 4 Blv. Vasile Pârvan, Timișoara 300223, Romania {oaritioni, vnegru}@info.uvt.ro

take into consideration the interaction between the household residents. Our simulation model is based on scenarios, the HBSDL language, and households profiles.

The paper presents a human behavior simulator with details about the scenarios description, and we define all the information that we need about the household and its residents behavior. We concentrate on the next section to describe the HBSDL vocabulary and syntax. The third section presents what kind of information needs the simulation process: the household residential profile, behavioral profile and energy-efficiency awareness profile. After we outline the simulator architecture: how we process the input data, how we build the behavior prototype, the rule-based systems and constraints and finally the context management. We conclude the paper with discussions about the results, the related work and with the important fields of the future work.

II. HUMAN BEHAVIOR SCENARIOS DESCRIPTION LANGUAGE

Our goal is to describe the human behavior during a large period of time. We use in this scope scenarios. A scenario is a linked list of actions which happen across a period of time. An action can be understood as a human activity which occurs in a time interval and in a specific place. Also, we use the term of action to describe weather conditions or events. These actions describe the interactions between the subject and the environment. The environment represents the resident living space, the place where the action happens. Our goal is to simulate the energy usage in a household and we are interested to describe the interactions which implies devices and appliances.

To describe a scenario we have defined a special language: Human Behavior Scenarios Description (HBSDL) Language. HBSDL allows us to introduce scenarios using a vocabulary and a syntax, very close to the natural language. The HBSDL language vocabulary is based on the English language.

Based on the human behavior simulation we can simulate the energy consumption and data-stream of other sensors. The information and the statistics about the human behavior help us to obtain a good overview of the energy usage in a household. In order to obtain accurate scenarios we have defined a compiler to check if the simulator output respects the syntax.

Our vocabulary contains 41 terms, one command and variables. The table I contains the HBSDL vocabulary. The HBSDL language uses the command `#day` to indicate the date for the scenario. We use this command in order to allow to introduce a scenario for a long period of time. We have

used variables to indicate the name of the persons from the household or to indicate the rooms where the actions occur. We use a special notation to indicate the action location, because in many situations the house has more than one bedroom or kitchen or living room. The used notation is $x \# \text{type_of_room}$ where x indicates the room number, and must to be an integer. We extract the name for all the variables, which represent the house residents and the rooms, from the household description profile (a XML file which contains the household profile).

Grammatical Category	Term
Predicate	turns everything off, uses, goes to bed, comes home, is resting, takes a bath, turns on, turns off, sets
Direct Object	gas cooker, vacuum cleaner, refrigerator, electric plate, air conditioning, coffee filter, radiator, the light, computer, hair dryer, toaster, heating, stove, sink, television, toilet microwave, dish washer
Modal Adjunct	low, medium, high
Temporal Adjunct	at X, from X to Y, at X for Y minutes

TABLE I

HBSDL LANGUAGE TERMS

The HBSDL grammar follows the English language grammar. A scenario is composed by instructions. Each instruction is a sentence written in the English language. It is mandatory that each sentence has to contain one subject and one predicate. Other categories such as direct object, modal and time complements are not mandatory. The subject of the "sentence" is the house resident which has an interaction with some devices, and the predicate describes this interaction. The HBSDL grammar can be described using a deterministic finite state-machine (see figure 1).

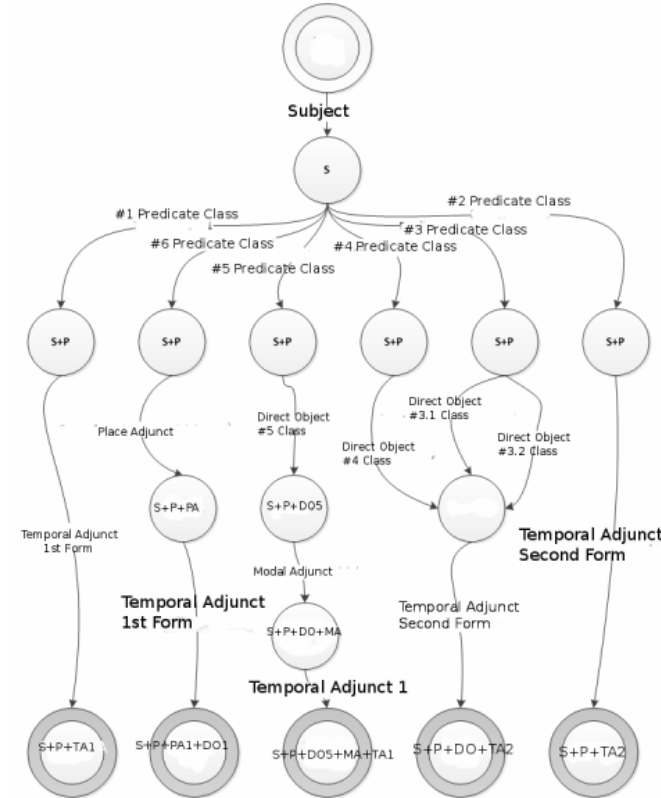


Fig. 1. The Finite State Machine for HBSDL

Predicate Category	Predicates
First Category	drinks, turns everything off, opens window, closes windows, comes home, go to bed
Second Category	eats, is resting, takes a bath, showers
Third Category	washes, uses
4th Category	turn on, turn off
5th Category	sets
6th Category	wakes up in, enters

TABLE II

PREDICATES CATEGORIES

The predicate is the most important. We define six predicates categories as you can see in the table II. These categories indicate the sentence syntax. For example if we have a predicate of the first class the sentence syntax is: subject + predicate + temporal adjunct.

The sentence which accepts direct objects must also respect some important conditions. For each predicate class we have defined a list of direct objects allowed. We have defined three categories of direct objects :

- Direct Objects for the third class predicates: sink, vacuum cleaner, toilet, refrigerator, hair dryer, gas cooker.
- Direct Objects for the fourth class predicates: television, air conditioning, computer, radiator, the light, heating, stove, electric plate, coffee filter, toaster, microwave, washing machine, dish washer.
- Direct Objects for the fifth class predicates: refrigerator, stove, heating, radiator.

For the temporal adjunct we accept only two forms:

- The first form, which is used for the instant actions: at X.
- The second form, which is used for the long-time actions: from X to Y, at X for Y minutes.

We have three categories of actions:

- instant-actions (the person spent less than one minute to do that action), such as: wake up, goes to bed, turns on, etc.
- long time exclusive actions (the person can do at a certain moment only that action, which is a long time action), such as: uses toilet, uses sink, etc.
- long time non-exclusive actions (the person can do at a certain moment that action and also other actions), such as: watching TV, uses washing-machine, etc.

The place of the action must be a room from the household, when we use the notation $x \# \text{type_of_room}$.

III. HOUSEHOLD DESCRIPTION USING PROFILES

Our goal is to simulate the behavior of people from a specific household, we do not want to obtain a general simulation of the human behavior. So, we have to take into consideration a lot of information about the house structure and the behavior of residents, in order to obtain a good simulation. To describe a household we use a form which contains a lot of questions. We can collect data through web-pages, to obtain in this way the following information:

- **The Household Residential Profile** describes the physical structure: number of household rooms, how many appliances are in each room, a list of all the appliances, etc. For each room we store data about: type (kitchen /

living-room / bedroom / etc), room-name, width, length, height and a list of all the objects from that room. For each object we must know the object-type (appliances which use electricity / static object / object which use gas / etc) and the object-name.

- **The Household Behavioral Profile** contains the household inhabitants number, their profession, their age and sex, data about each subject life style, inhabitants habits, etc. For each household resident who works or is a student we must know the weekly schedule. Also, we must know for each person all its preferences .
- **The Household Energy-Efficiency Awareness Profile** explains us the level of house residents attention about the energy - efficiency: how many "ecological" appliances are in the house, habits about interactions with devices, etc.

We associate for each household an unique residential profile, a behavioral profile and an energy-efficiency awareness profile.

We have developed a study over more than 800 households. For this study we have defined a questionnaire which follows a decision tree schema. Using this statistical study we found 110 profiles. All these profiles are stored in a database.

We use these profiles in the process of the human behavior simulation. Firstly the house structure is very important: in each room the person can perform only some activities and there are some activities which are not allowed. Also the appliances and their "map" which indicates their position in the household, allows the resident to perform or not some activities. We use also the geometrical description of the household for developing the movement equations for each person. Data about behavior, the person age, the weekly-schedule, the lifestyle are also very important for us. If we know for example that the person is a man which goes to his job between 9 a.m. and 18 p.m. it is mandatory that in this time interval the scenario will be empty, without any action. The profile about awareness is used to indicate the frequency of some "Eco-friendly" actions such as "turns off" or "turns everything off". Our goal is to translate these profiles into a set of constraints which will lead the simulation process.

The subject's age and sex have an important role in the simulation process [10]. Also we develop a taxonomy which describes the features with great impact in the human behavior. Based on this taxonomy we define 80 human behavior prototypes, such as: married woman, no job, with children who lives in urban area; married old-man, part-time job, with nephews, who lives in rural area, etc.

The section IV will explain how these profiles are used as input in the simulation process.

IV. THE SIMULATION PROCESS

We will present in this section the architecture of the software prototype, developed using JESS and Java. We chose Java because it provides support for multi-threading programming. Our simulator uses a multi-threading architecture: each household resident behavior is simulated using an unique thread. JESS provides support to "implement" the

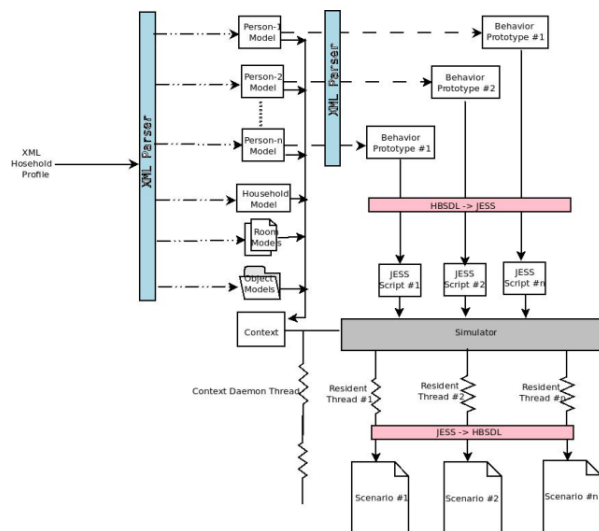


Fig. 2. The human behavior simulation process description

constraints and good communication with Java.

The simulation process consists of the following stages:

- Profile Understanding: we analyze in this stage the XML files which describe the household profile and we build: the household model, the room models, the object models and a model for each household resident - the person model.
- Building the behavioral prototype. Using the model for each subject, the household model, the data about the rooms and objects we build for each household resident the behavioral prototype. A behavior prototype contains a general scenario, a set of constraints and the associated decision-trees. This scenario is an one day scenario, which contains all the HBSDL instructions without specific time details. This means that for each action the time moment is a variable, such as in the following example:

Alice_wakes_up_in_0#Bedroom_at_t0
 Alice_turns_on_light_at_t1
 Alice_turns_on_television_at_t2
 Alice_enters_2#Bathroom_at_t3

 Alice_goes_to_bed_at_tn

The behavior prototype contains a set of constraints over the variables $t_0, t_1, t_2, \dots, t_n$, such as $t_0 \leq t_i, i = \overline{1..n}$. Constraints about the variables t_i are also developed only if we have habits definitions in the person model. We insert in the behavior prototype HBSDL instructions for all the possible actions.

- JESS script building process. On this stage we translate all the above HBSDL scripts into JESS scripts. For each subject we have as input a behavioral prototype and we obtain a JESS script. We translate the constraints set and the decision-tree about that subject.
- The simulation process continues with the context initialization using all the data from the all the models (person models, household models, rooms models, objects models).
- Threads building process. The simulator build the context daemon process and a thread for each resident. The

simulator load on each resident thread a particular JESS script. For example if in our household we have four residents - Alice, Bob, David and Eve - the generator will load the JESS scripts Alice.clp, Bob.clp, David.clp and Eve.clp on the threads Alice, Bob, David and Eve.

- Each thread runs and changes messages with the context daemon thread, which manages the context (working memory). The daemon thread sends a message about the current-time to all the resident-thread, and select the first thread which will run (the selection is based on a random number). The first thread selects an action and sends a message to the daemon thread about that. The daemon thread updates the context and chooses the next thread which will run (the selection is based on a random number). When all the threads finish the process of selecting an action, the daemon thread "increments" the time.
- After each thread finishes all its jobs, the daemon thread send a "kill message". When a resident-thread receives a "kill message" it translates all the JESS facts which describe the scenario into HBSDDL instructions and writes them into the scenario file.

The most important model is the **Person Model**, which stores data about:

- name, age, sex, weekly schedule.
- a list of all the objects which are allowed for that person to use them.
- a list of all the allowed actions for that person. For each action we store also the minimum/maximum number of occurrences, and the optimal frequency.
- a list of habits ("pattern" of actions) and minimum/maximum number of occurrences and the optimal frequency.

Using this model we build for each person a behavior prototype.

Our simulator deploys a thread for each household resident. Inside each thread we run a JESS script. We call that **resident-thread**, because it simulates the behavior of one household resident. These threads do not run independently. For example we can have a woman who opens a window. Outside the house is very cold, and a child from the same room does not like to be cold in that room. In these situation we have a communication between the threads, which proves that our simulator does not use independent threads: the woman-thread changes the window states, and the context daemon thread notifies the others threads and the children-thread takes the decision to close the window.

A. Rules and Constraints

Our prototype uses a set of constraints in order to obtain real-life scenarios. These constraints are related to the resources management, user location, the time, the subject, etc. For all these constraints we define JESS rules to check them.

We have chosen each new action using a decision-tree and a set of constraints. Firstly, we investigate the historic of the household and of the resident life. After we check all the constraints to obtain a set of actions that the resident-thread

can perform. We generate a random number, and using it we will choose the next action.

Each **action** has a **priority**. For example at 8:30 PM Alice can: (a1) turn on the light, (a2) use the vacuum cleaner or (a3) the toaster. The most probable action is a1, after a2 and the less probable action is a3. In our JESS script the salience mechanism is used in order to choose the next action. If we have two or more actions which have the same priority we generate a random number, and we use it to choose the next action.

To choose an action means to set the time-slot of the actions which we have chosen. Do not forget that the behavior prototype contains all the actions that a person can do, without indicating the time.

To check all the constraints we define rules. We have the following categories of constraints:

- **Constraints related to the user-location.** For example in the bathroom we can't watch TV.
- **Constraints related with time.** For the instant-actions and long-time exclusive actions we define one rule and for the long-time non-exclusive action we define two rules: one for the start of action and another for the ending of action. For the long-time action the time spent by the subject is randomly generated and must respect some constraints about the minimum / maximum period of time for that action. For example, the minimum time to take a shower is 5 minutes.
- **Constraints related with the subject.** There are some actions which are not allowed for all the people. For example a children can not use the washing machine.
- **Constraints for resources management.** There are some resources which allow access only for one subject: the microwave for example.
- **Logic Constraints**, which check the logic of the scenarios. For example, if a person has stopped the TV, now it is forbidden that other person will start the TV. There are situations when an action can be stopped and started after less than one minute, which is without any logical reason.
- **Social constraints**, such as: at the same time it is not allowed for two person to use the bathroom. The social rules are defined to respect the social laws about living in the same space.
- **Constraints related with the user preferences.** For example a man does not like cooking, and we must introduce the constraints that won't allow the introduction of this action in the scenario.
- **Historic Constraints.** These constraints are related to the scenario and the last actions: for example if a subject that sleep usually eight hours wake up at 7.30 AM he it is not allowed going to sleep from 8 AM to 11 AM.

Separately, **the lights management** is based on the energy efficiency awareness profile. When leaving a room, each resident-thread checks if some other resident is in the same room. If not, the resident-thread turns off the lights. When entering a new room, the resident-thread checks if the lights

are turned on. If not, it turns on the light in that room.

B. The simulation context

The life of one threads can be summarized as follows: first, the thread is initialized, having a name, a behavior prototype and list of preferences regarding the activities it wants to do. The state of a thread is based on a private field which contains the state of an thread at a given moment. Each activity has a given timespan. The resident-threads registers itself to the context, which keeps track of all resident-threads from the house. If the time associated to an activity elapsed, then the thread sends a Change Activity message to the context. The context-replies each household resident-thread state. The resident-thread who initiated the Change Activity gathered all the states and adds then to a list. The resident-thread then sends that list together with a list of its preferences to the working memory of the JESS engine which returns a new activity for the resident-thread with a margin error of ± 15 minutes or null. If the returned value is null, the resident-thread randomly picks a room and a devices in that room and asks the context if it is available and can be used. If not, it repeated picking randomly a room and a device until a free device in some room is found. The new state of the thread will be either the returned value of JESS, if it is not null, or an activity chosen randomly as showed before.

Between the resident-threads and the daemon-context thread is a direct communication. The context thread mediates the interactions between threads: there is any direct interaction between the resident-threads. The figure 3 explains the interaction between threads in our prototype. We use the Active Object design pattern.

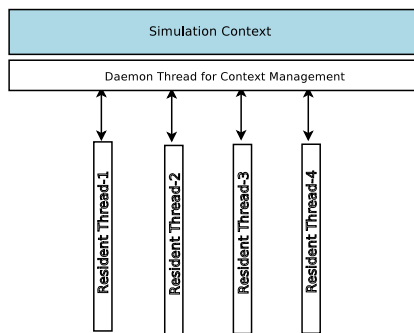


Fig. 3. Threads and context interaction

The context keeps track of all the resources from the household and stores information about the weather. De facto, the context is the working memory of our prototype. The context is represented by a working memory and a daemon-thread which reads and/or writes this memory and validates it.

Firstly, we "generate" the weather scenario, because it has a great impact over the human behavior. Using the household residential profile we develop a graph resources. This graph will be updated by the context daemon-thread. The graph explains the relations between the resources and the relations between the resources and the household residents. The working memory (context) stores this resources graph. Our

resources are: the household, the rooms and all the objects.

The reason to use a central-context instead to use a notification mechanism between threads is that we must to validate each time all the information about the household. The context has a good overview about what happens in each room, with each resources and with each resident. The context uses a validation mechanism in order to prevent having inconsistent scenarios. The validation can be view as a supplementary check-service.

Time is simulated in the context daemon thread using a **Timer Event** containing time information. The context daemon thread is put to sleep for one second. The resident-threads are reactivated and are able to interact and change states during this time. When the daemon thread wakes up, the timer is incremented with one minute after which the main thread is put back to sleep.

The constraints propagation mechanism is related to the resources management. A resident can use at a certain time one or more resources, or can change the state of resources. When a resident-thread changes the state of one resource it sends a notification to the context. The context updates the resources graph and notifies all the others resident threads not to use this resource(s). In this way, we propagate the constraint about not using a resource from a resident-thread to the others. Our propagation mechanism uses the Active Object design pattern for multi-threading programming. The social constraints are activated and all the others threads receive a notification. The constraint propagation mechanism consists of a set of notifications:

(i) a notification between the source resident-thread and the context. The source resident-thread is the thread which locks a resource.

(ii) notifications of all the others resident-threads from the context.

In conclusion, the context uses a working-memory to store data about the household (rooms, objects, residents) and provides services to update and to validate this information.

V. RESULTS AND DISCUSSIONS

Our software prototype allows to simulate the human behavior using as input data about the household structure and residents. The user can define the household profile or can load a profile from a repository. The figure 4 and 5 provide an overview about the simulator's user-interface.

The test case contains a house with seven rooms including the bathrooms each containing at least two devices. Four threads are used to represent a family living in that house. Each house resident has its own preferences regarding the activity it wants to do at a given moment. All the data regarding the house and the household people is extracted from a XML file. An example output of the application is presented in the following snippet.

```

Alice_wakes_up_in_0#Bedroom_at_06:30:20
Alice_turns_on_light_at_06:30:25
Alice_turns_on_television_at_06:31:00
Alice_enters_2#Bathroom_at_06:50:00
.....
Alice_enters_0#Bedroom_at_21:31:00
Alice_turns_on_light_at_21:31:30
Alice_turns_off_light_at_22:00:00
Alice_goes_to_bed_at_22:10:00

```

We generate scenarios for a four person family for one week. We define a typical family with two married people

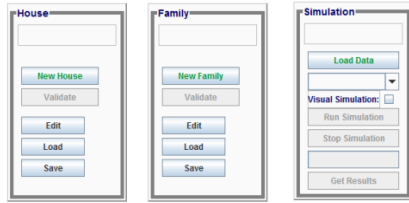


Fig. 4. The main menu of the scenarios generator application.

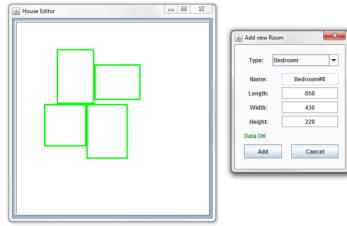


Fig. 5. The House Editor editor allows to describe the architectural structure of the user house.

and two children who live in a four room house. The man works from 8 AM to 20-21 PM and his wife works from 9 AM to 15 PM, from Monday to Friday. The children, Alice and Bob, have the same school program from 8 AM to 14 PM. The figure 6 show the length of the scenarios for each subject from Monday to Sunday. The Bob scenario is described using 164 sentences, and for each sentence the main attributes are the action, the start-time and end-time. For each action we associate a color: dark blue for wakes up, red for goes, blue for turn on / uses an appliance, gray for washes, pink for turns off, light green for exit, yellow for comes, purple for goes to bed, etc. For each day of the week you can see in figure 7 a column which describes what actions are done in that day.

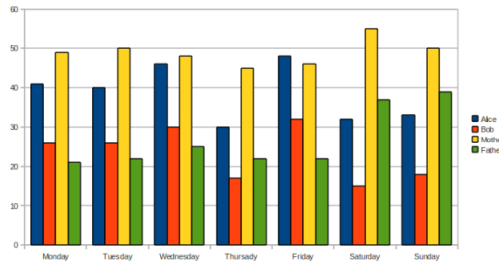


Fig. 6. Scenario length variation for a week

VI. CONCLUSIONS AND FUTURE WORKS

The simulation problem in ambient intelligent systems is an important subject. In this paper we have presented

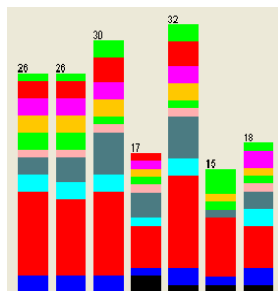


Fig. 7. Bob scenario summary for one week

a human behavior simulator for households. Our simulator provides support for domestic resources consumption forecasting (electricity, water consumption, gas). Also it can be used as a tool for researching. It provides support for testing and debugging ambient intelligent applications.

An important contribution of our research is the HBSDL language, used for the scenarios formalization. This language is based on an ADL ontology. A compiler for this language was developed in order to check the HBSDL scripts. The simulation output is written to a file called "logfile.sce". This language is extensible and allows the introduction of new terms, new actions or new objects. Also we have defined a framework to describe the households structure.

An important research was developed to define the households profiles. We have used in our research more than 800 households from the Banat Romanian region. Based on this statistical study we have defined more than 110 profiles for households.

The process of building behavioral prototypes is based on a large set of 80 predefined behavioral schema. The constraints and rules are developed using data about the household. To achieve this goal we have defined a set of meta-rules. An important contribution is the propagation mechanism using the context.

As a future work we will develop a sensor-data stream simulator based on scenarios. Using as input data about a household and its inhabitants our generator will develop a set of scenarios. The simulator of sensor-data streams takes as input this set of scenarios and will produce sensor-data streams. The simulator of sensor-data streams based on scenarios is a work-in-progress project.

REFERENCES

- [1] "Climate change programme review," 2005.
- [2] "European commission press," Mars 2007.
- [3] O. Artoni and V. Negru, *A multi-agent recommendation system for energy efficiency improvement*, ser. Communications in Computer and Information Science, 2011, vol. 171 CCIS, pp. 156 – 170.
- [4] D. Bourgeois, C. Reinhart, and I. Macdonald, "Adding advanced behavioural models in whole building energy simulation: A study on the total energy impact of manual and automated lighting control," *Energy and Buildings*, vol. 38, no. 7, pp. 814 – 823, 2006, special Issue on Daylighting Buildings.
- [5] J. Desmedt, J. V. Bael, and G. Vekemans, "Influencing households energy behaviour, an in-situ measuring campaign," Flemish Institute for Technological Research, VITO, Mol, Belgium, Tech. Rep., 2009.
- [6] B. Dong and B. Andrews, "Sensor-based occupancy behavioral pattern recognition for energy and comfort management in intelligent buildings center for building performance and diagnostics," *Design*, pp. 1444 – 1451, 2009.
- [7] A. Kashif, X. H. B. Le, J. Dugdale, and S. Ploix, "Agent based framework to simulate inhabitants' behaviour in domestic settings for energy management," in *ICAART (2)*, 2011, pp. 190 – 199.
- [8] J. Seryak and K. Kisson, *Occupancy and behavioral affects on residential energy use*. American solar energy society, 2003, pp. 717 – 722.
- [9] M. Sierhuis, W. J. Clancey, and R. J. J. V. Hoof, "Brahms: a multi-agent modelling environment for simulating work processes and practices," *International Journal of Simulation and Process Modelling*, vol. 3, no. 3, pp. 134 – 152, 2007.
- [10] L. M. Verbrugge, A. L. Gruber-Baldini, and J. L. Forzard, "Age differences and age changes in activities: Baltimore longitudinal study of aging," *J Gerontol B Psychol Sci Soc Sci*, vol. 51, no. 1, pp. 30 – 41, 1996.

On the Boundedness of Timed Continuous Petri Nets

Timotei Asaftei, and Mihaela-Hanako Matcovschi, *Member IEEE*

Abstract— For a timed continuous Petri net (TCPN), the paper introduces a new concept of boundedness based on the existence of (positive) invariant sets with respect to the marking of the net, sets that are defined by weighted Hölder p -norms. For join-free TCPNs, this type of boundedness is characterized by inequalities satisfied by the measure of the matrix that defines the dynamics of the marking. Some connections with the structural properties of the net topology are also delivered.

I. INTRODUCTION

PETRI nets (PNs) were first introduced as a graphical and mathematical tool for the modeling, analysis and synthesis of discrete event systems [1]. In *discrete* PNs the marking of each place in the net is restricted to be a non-negative integer. Based on the idea of the fluidization of models for the performance evaluation of production lines, an extension of the discrete PNs was introduced in [2] allowing the transition firing process to be continuous and the marking to be represented by non-negative real numbers instead of integers. Works such as [3]–[5] presented different ways for timing continuous PNs, and introduced *finite server semantics* (constant speed) or *infinite server semantics* (variable speed) to allow describing the token-flow through transitions. Paper [6] shows that, for a wide class of systems, *timed continuous Petri Nets* operating with infinite server semantics provide a better approximation of the steady state throughput than when using finite server semantics.

The model of a *timed continuous Petri net* (TCPN) is represented by a differential equation. For that reason, a large number of analytical techniques can be used for investigating the properties of TCPNs. For TCPNs, papers [7]–[11] studied the properties of controllability, observability, optimal control, predictive control and deadlock freeness. Applications of TCPNs include traffic systems [11], manufacturing systems [12], or biochemical networks [13].

The current paper introduces a new type of boundedness for TCPNs that generalizes the classical k -boundedness of discrete Petri nets. This type of boundedness is based on the existence of sets which are (positively) invariant with respect to the trajectories of the TCPN, i.e. any trajectory initiated inside that set does not leave it any longer. The study of invariant sets emerged as a research trend in mathematics at the middle of the 20-th century, as pointed out by the monograph [14]. The main theoretical results

were applied in control engineering in the 1980s and exploited in analysis and design techniques. The latest monograph [15] details the progress in the area by summarizing more than 300 works representing noticeable contributions to this research trend. The theory of invariance was applied for the qualitative analysis of TCPNs in paper [16]. The present paper extends the framework created in paper [16] for constant sets by considering sets defined by weighted Hölder p -norms; thus, the study covered in paper [16] corresponds to the particular case when $p = \infty$.

The paper is organized as follows. Section II presents the basic concepts in the theory of timed continuous Petri nets with infinite server semantics. Section III introduces a new concept of boundedness based on the existence of invariant sets with respect to the marking of a net. The case of join-free Petri nets and some connections with the structural properties of discrete Petri nets are detailed in Sections IV and V, respectively.

II. PRELIMINARIES

The topology $\mathcal{N} = \langle \mathcal{P}, \mathcal{T}, \mathbf{C}^+, \mathbf{C}^- \rangle$ of a continuous PN is a directed, weighted, bipartite graph defined by the disjoint sets of places, $\mathcal{P} = \{p_1, \dots, p_n\}$, and transitions, $\mathcal{T} = \{t_1, \dots, t_m\}$, and by the input and output incidence matrices (with respect to places), denoted by $\mathbf{C}^+ = [\mathbf{C}_{ij}^+]$ and $\mathbf{C}^- = [\mathbf{C}_{ij}^-]$, $\mathbf{C}^+, \mathbf{C}^- \in \mathbb{R}_+^{n \times m}$, respectively.

For a node $x \in \mathcal{P} \cup \mathcal{T}$, the preset and postset are denoted $\bullet x$ and $x^\bullet$, respectively. We assume that each transition has at least one input place ($|\bullet t_j| \geq 1$ for all $j = 1, \dots, m$), meaning that there are no source transitions in the net.

Each place in a *continuous PN* can accommodate any non-negative real number of tokens. Thus, the marking vector is $\mathbf{m} = [m_1 \dots m_n]^T \in \mathbb{R}_+^n$, where m_i denotes the marking of the place p_i , for $i = 1, \dots, n$.

In a continuous PN a transition t_j ($j = 1, \dots, m$) is *enabled* at the marking $\mathbf{m} = [m_1 \dots m_n]^T \in \mathbb{R}_+^n$ if all its input places are marked, that is $m_i > 0$ for all $p_i \in \bullet t_j$. The *enabling degree* of transition t_j at the marking $\mathbf{m}$ is defined as

$$e_j(\mathbf{m}) = \min_{p_i \in \bullet t_j} \left\{ m_i / C_{ij}^- \right\}. \quad (1)$$

In the following we consider *timed continuous Petri nets* (TCPNs) defined by a topology $\mathcal{N}$ and a positive vector of constant firing rates associated with the transitions in the net,

Manuscript received September 2, 2011. The first author acknowledges the support of the European Union through the European Social Fund and of the Romanian Government under Grant EURODOC ID 59410.

The authors are with the Department of Automatic Control and Applied Informatics from the Technical University “Gh. Asachi” of Iași, Romania (phone/fax: +40 232 230 751; e-mail: mhanako@ac.tuiasi.ro).

$\lambda \in \mathbb{R}^m$, $\lambda = [\lambda_1 \dots \lambda_m]^T > 0$, and denoted by $\langle \mathcal{N}, \lambda \rangle$.

Under the *infinite server semantics* [6], the token-flow through transition t_j at an arbitrary moment $\tau \in \mathbb{R}_+$, when the marking of the net is $\mathbf{m}(\tau) = [m_1(\tau) \dots m_n(\tau)]^T \in \mathbb{R}_+^n$, equals the product of its firing rate and enabling degree,

$$f_j(\tau) = \lambda_j e_j(\mathbf{m}(\tau)), \quad j = 1, \dots, m. \quad (2)$$

The *state-equation* corresponding to a TCPN $\langle \mathcal{N}, \lambda \rangle$ models the time-dependence of its marking:

$$\dot{\mathbf{m}}(\tau) = \mathbf{C} \mathbf{f}(\tau), \quad \tau \geq 0, \quad (3)$$

where $\mathbf{f}(\tau) = [f_1(\tau) \ f_2(\tau) \ \dots \ f_m(\tau)]^T$ is the flow vector. In (3), matrix $\mathbf{C} = [C_{ij}] = \mathbf{C}^+ - \mathbf{C}^- \in \mathbb{R}^{n \times m}$ is the *incidence matrix* that characterizes completely the topology $\mathcal{N}$ provided that the net is *pure* (it does not contain self-loops, i.e. $\bullet x \cap x \bullet = \emptyset$, $\forall x \in \mathcal{P} \cup \mathcal{T}$).

A topology $\mathcal{N}$ is said to be *join-free* (JF) in the case when every transition has exactly one input place ($|\bullet t_j| = 1$ for $j = 1, \dots, m$). The evolution of the marking of a JF TCPN $\langle \mathcal{N}, \lambda \rangle$ is modeled by a unique linear differential equation:

$$\dot{\mathbf{m}}(\tau) = \mathbf{A} \mathbf{m}(\tau), \quad \tau \geq 0, \quad (4)$$

with

$$\mathbf{A} = \mathbf{C} \mathbf{\Lambda} \mathbf{\Pi}, \quad (5)$$

where $\mathbf{\Lambda}$ is the diagonal matrix built with the transition firing rates, also called *firing rates matrix*,

$$\mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_m\} \in \mathbb{R}_+^{m \times m}, \quad (6)$$

and the elements of matrix $\mathbf{\Pi} = [\pi_{ji}] \in \mathbb{R}_+^{m \times n}$ are given by

$$\pi_{ji} = \begin{cases} 1 / C_{ij}^-, & \text{if } \bullet t_j = \{p_i\}, \quad i = 1, \dots, n, \quad j = 1, \dots, m. \\ 0, & \text{otherwise,} \end{cases} \quad (7)$$

Note that for a JF TCPN the matrix $\mathbf{\Pi}$ defined by (7) has exactly one positive element in each row, because $|\bullet t_j| = 1$ for all $j = 1, \dots, m$, and it satisfies $\mathbf{\Pi} \mathbf{C}^- = \mathbf{I}_m$. The system matrix $\mathbf{A}$ given by (5) is *essentially non-negative* (also called Metzler matrix), i.e. all its off-diagonal elements of are non-negative.

If a TCPN is not join-free, then its mathematical model must take into account all possible combinations of places that command the enabling degree of transitions. The mathematical model of a general TCPN is represented by a set of linear differential systems with state-dependent switching [16].

Paper [16] gives a consistent and rigorous mathematical proof based on set-invariance for the non-negativeness of the marking of general TCPNs.

Lemma 1. [16]

For any choice of the firing rates $\lambda \in \mathbb{R}^m$, $\lambda > 0$, associated with the transitions of a TCPN (with or without joins), the continuous marking of the net remains non-negative, i.e. it satisfies:

$$\mathbf{m}(\tau) \geq 0, \quad \forall \tau \geq 0, \quad (8)$$

for any non-negative initial marking $\mathbf{m}(0) = \mathbf{m}_0 \geq 0$. ■

III. NEW BOUNDEDNESS AND CONSERVATIVITY CONCEPTS

Let $\mathbf{d} = [d_1 \dots d_n]^T \in \mathbb{R}^n$, $\mathbf{d} > 0$, be a positive vector and construct the diagonal matrix $\mathbf{D} = \text{diag}\{\mathbf{d}\}$. In $\mathbb{R}^n$ consider the Hölder p -norm $\|\cdot\|_p$ with $1 \leq p \leq \infty$ and define

$$\mathcal{X}_p^{\mathbf{d}} = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{D}^{-1} \mathbf{x}\|_p \leq 1\}. \quad (9)$$

Definition 1.

The set $\mathcal{X}_p^{\mathbf{d}}$ defined by (9) is called (*positively*) *invariant* with respect to the marking of a TCPN if any marking initiated inside the set $\mathcal{X}_p^{\mathbf{d}}$ will never leave $\mathcal{X}_p^{\mathbf{d}}$, i.e.:

$$\forall \mathbf{m}_0 \in \mathbb{R}^n : \|\mathbf{D}^{-1} \mathbf{m}_0\|_p \leq 1 \Rightarrow \|\mathbf{m}(\tau; \mathbf{m}_0)\|_p \leq 1, \quad \forall \tau \geq 0, \quad (10)$$

where $\mathbf{m}(\tau; \mathbf{m}_0)$ represents the marking of the TCPN at time τ corresponding to the initial marking $\mathbf{m}(0) = \mathbf{m}_0$. ■

Remark 1.

Due to the linearity of system (4), if for a given $\mathbf{d} > 0$ the set $\mathcal{X}_p^{\mathbf{d}}$ (9) is invariant with respect to the marking of the JF TCPN (4), then for any positive scalar $c > 0$, the set $\mathcal{X}_p^{c\mathbf{d}}$ is also invariant with respect to the marking of (4). Therefore, an invariant set $\mathcal{X}_p^{\mathbf{d}}$ is unique up to a positive scaling factor. ■

Remark 2.

In the analysis of a JF TCPN $\langle \mathcal{N}, \lambda \rangle$ modeled by (4) we are interested only in the solutions that correspond to non-negative initial conditions. As commented in the end of Section II, system (4) is positive, or, in other words, the non-negative orthant $\mathbb{R}_+^n$ is invariant with respect to the state-space trajectories of (4). Consequently, if for a given $\mathbf{d} > 0$ the set $\mathcal{X}_p^{\mathbf{d}}$ (9) is invariant with respect to the marking of JF TCPN (4), then the set $\mathcal{X}_{p,+}^{\mathbf{d}} = \mathcal{X}_p^{\mathbf{d}} \cap \mathbb{R}_+^n$ is also invariant with respect to the trajectories of (4). In the following we will study the invariance of the sets $\mathcal{X}_p^{\mathbf{d}}$ given by (9), having in mind that for TCPN analysis we are in fact interested in the invariance of the sets $\mathcal{X}_{p,+}^{\mathbf{d}}$. ■

Two particular cases that allow connections with the structural properties of the discrete Petri nets and a numerical approach based on linear programming problems correspond to the usual norms $p = \infty$ and $p = 1$, detailed further on.

For $p = \infty$, the sets $\mathcal{X}_p^d$ (9) are hyper-rectangles, symmetrical with respect to the origin; the elements of vector $\mathbf{d}$ represent its semi-axes. Taking Remark 2 into account, condition (10) written for non-negative initial markings is equivalent to

$$\forall \mathbf{m}_0 \in \mathbb{R}^n : 0 \leq \mathbf{m}_0 \leq \mathbf{d} \Rightarrow 0 \leq \mathbf{m}(\tau; \mathbf{m}_0) \leq \mathbf{d}, \forall \tau \geq 0, \quad (11)$$

where inequalities are understood componentwise. This particular case was discussed in detail in paper [16].

Vector $\mathbf{d}$ may be regarded as a vector of capacities assigned to the places of the TCPN. Condition (11) may be interpreted in the sense that the capacities of the places are not exceeded at any moment during the evolution of the marking. This condition also ensures that the token-flow through each transition does not go above a certain value.

Definition 2.

Given a positive vector $\mathbf{d} \in \mathbb{R}^n$, $\mathbf{d} > 0$, we say that a TCPN is *$\mathbf{d}$ -upper bounded* if condition (11) is satisfied. ■

For $p = 1$, the sets $\mathcal{X}_p^d$ (9) are hyper-diamonds, symmetrical with respect to the origin; the elements of vector $\mathbf{d}$ represent its semi-axes. Taking Remark 2 into account, condition (10) written for non-negative initial markings is equivalent to

$$\begin{aligned} \forall \mathbf{m}_0 = [m_1^0 \dots m_n^0]^T \in \mathbb{R}^n, \mathbf{m}_0 \geq 0 : \\ \delta_1 m_1^0 + \dots + \delta_n m_n^0 \leq 1 \Rightarrow \delta_1 m_1(\tau) + \dots + \delta_n m_n(\tau) \leq 1, \forall \tau \geq 0, \end{aligned} \quad (12)$$

where $\delta_i = 1/d_i$, for $i = 1, \dots, n$, and $m_1(\tau), \dots, m_n(\tau)$ represent the elements of $\mathbf{m}(\tau; \mathbf{m}_0)$. The weighted sum of the markings appearing in condition (12) resembles to the conservativity condition for a discrete Petri net. This is the reason why we introduce the following property.

Definition 3.

Given a positive vector $\boldsymbol{\delta} = [\delta_1 \dots \delta_n]^T \in \mathbb{R}^n$, $\boldsymbol{\delta} > 0$, we say that a TCPN is *weakly $\boldsymbol{\delta}$ -conservative* if condition (12) is satisfied. ■

Remark 3.

Based on the same line of reasoning as in Remark 1, we notice that if a TCPN is *$\mathbf{d}$ -upper bounded* for some positive vector $\mathbf{d} \in \mathbb{R}^n$, $\mathbf{d} > 0$, then the TCPN is also *$c\mathbf{d}$ -upper bounded* for arbitrary $c > 0$. A similar statement is valid for the property of *weak $\boldsymbol{\delta}$ -conservativeness*. ■

IV. ANALYSIS OF JOIN-FREE TIMED CONTINUOUS PETRI NETS

We consider the case of a join-free TCPN $\langle \mathcal{N}, \mathcal{A} \rangle$ modeled by (4)&(5) and prove the following result.

Theorem 1.

Let $1 \leq p \leq \infty$, $\mathbf{d} \in \mathbb{R}^n$, $\mathbf{d} > 0$, and $\mathbf{D} = \text{diag}\{\mathbf{d}\}$. The following statements are equivalent:

(i) Matrix $\mathbf{A}$ defined by (5) satisfies the inequality

$$\mu_p(\mathbf{D}^{-1}\mathbf{A}\mathbf{D}) \leq 0, \quad (13)$$

where $\mu_p(\cdot)$ is the matrix measure [17] based on the matrix norm $\|\cdot\|_p$, $\mu_p(\mathbf{M}) = \lim_{h \downarrow 0} \frac{1}{h} (\|\mathbf{I} + h\mathbf{M}\|_p - 1)$.

(ii) The function

$$\mathcal{V} : \mathbb{R}^n \rightarrow \mathbb{R}_+, \mathcal{V}(\mathbf{x}) = \|\mathbf{D}^{-1}\mathbf{x}\|_p, \quad (14)$$

is a weak Lyapunov function for the JF TCPN (4), i.e. the *Dini right derivative* of $\mathcal{V}$ along each nontrivial trajectory of system (4) satisfies

$$D^+ \mathcal{V}(\mathbf{m}(\tau)) = \lim_{\eta \downarrow 0} \frac{1}{\eta} [\mathcal{V}(\mathbf{m}(\tau + \eta)) - \mathcal{V}(\mathbf{m}(\tau))] \leq 0, \forall \tau \geq 0. \quad (15)$$

(iii) The set $\mathcal{X}_p^d$ defined by (9) is invariant with respect to the marking of the JF TCPN (4).

Proof: (i) $\Rightarrow$ (ii). We have

$$\begin{aligned} D^+ \mathcal{V}(\mathbf{m}(\tau)) &= \lim_{\eta \downarrow 0} \frac{1}{\eta} [\|\mathbf{D}^{-1}\mathbf{m}(\tau + \eta)\|_p - \|\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p] = \\ &= \lim_{\eta \downarrow 0} \frac{1}{\eta} [\|\mathbf{D}^{-1}e^{\mathbf{A}\eta}\mathbf{D}\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p - \|\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p] \leq \\ &= \lim_{\eta \downarrow 0} \frac{1}{\eta} [\|\mathbf{D}^{-1}e^{\mathbf{A}\eta}\mathbf{D}\|_p \|\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p - \|\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p] = \\ &= \lim_{\eta \downarrow 0} \frac{1}{\eta} [\|e^{\mathbf{D}^{-1}\mathbf{A}\mathbf{D}\eta}\|_p \|\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p - \|\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p] = \\ &= \mu_p(\mathbf{D}^{-1}\mathbf{A}\mathbf{D}) \|\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p \leq 0. \end{aligned}$$

(ii) $\Rightarrow$ (iii). We give a proof by contradiction. Assume that the set $\mathcal{X}_p^d$ (9) is not invariant with respect to the marking of the JF TCPN (4). Let $\mathbf{m}$ be a solution to (4) that violates the invariance condition, i.e. there exist $\tau^*, \tau^{**} \in \mathbb{R}_+$, $\tau^* < \tau^{**}$, such that $\|\mathbf{D}^{-1}\mathbf{m}(\tau^*)\|_p \leq 1$ and $\|\mathbf{D}^{-1}\mathbf{m}(\tau^{**})\|_p > 1$.

This leads to $\mathcal{V}(\mathbf{m}(\tau^{**})) > \mathcal{V}(\mathbf{m}(\tau^*))$, contradicting (15).

(iii) $\Rightarrow$ (ii). Let $\mathbf{m}(\tau)$ be a solution to (4) corresponding to the initial condition $\mathbf{m}(0) = \mathbf{m}_0$. Set $\varepsilon = \|\mathbf{D}^{-1}\mathbf{m}_0\|_p$. If $\varepsilon = 0$, then $\mathbf{m}_0 = 0$, and hence $\mathbf{m}(\tau) = 0$, leading to $\mathcal{V}(\mathbf{m}(\tau)) = \mathcal{V}(\mathbf{m}(0)) = 0$ for all $\tau \geq 0$. If $\varepsilon > 0$, then $\tilde{\mathbf{m}}(\tau) = \varepsilon^{-1}\mathbf{m}(\tau)$ is also a solution to (4) and satisfies $\|\mathbf{D}^{-1}\tilde{\mathbf{m}}(0)\|_p = 1$. Since the set $\mathcal{X}_p^d$ is invariant with respect to system (4), we have $\|\mathbf{D}^{-1}\tilde{\mathbf{m}}(\tau)\|_p \leq 1$, or, equivalently $\|\varepsilon^{-1}\mathbf{m}(\tau)\|_p \leq 1$. Thus, for all $\tau \geq 0$, we get $\mathcal{V}(\mathbf{m}(\tau)) = \|\mathbf{D}^{-1}\tilde{\mathbf{m}}(\tau)\|_p \leq \varepsilon = \|\mathbf{D}^{-1}\tilde{\mathbf{m}}(0)\|_p = \mathcal{V}(\mathbf{m}(0))$, i.e. $\mathcal{V}(\mathbf{m}(\tau))$ is non-increasing along each trajectory of (4).

(ii) $\Rightarrow$ (i). Let $\tau \geq 0$. Based on the definition of the matrix norm, there exists a vector $\mathbf{m}_0 \in \mathbb{R}^n$, $\|\mathbf{D}^{-1}\mathbf{m}_0\|_p = 1$, such that $\|\mathbf{D}^{-1}e^{A\tau}\mathbf{D}\|_p = \|\mathbf{D}^{-1}e^{A\tau}\mathbf{D}\mathbf{D}^{-1}\mathbf{m}_0\|_p$. The solution to (4) initialized in $\mathbf{m}(0) = \mathbf{m}_0$, given by $\mathbf{m}(\tau) = e^{A\tau}\mathbf{m}_0$ satisfies $\|\mathbf{D}^{-1}\mathbf{m}(\tau)\|_p \leq \|\mathbf{D}^{-1}\mathbf{m}_0\|_p = 1$, due to the monotonicity of function (14) along any trajectory of system (4). Thus $\|\mathbf{D}^{-1}e^{A\tau}\mathbf{D}\|_p = \|\mathbf{m}(\tau)\|_p \leq 1$ and $\lim_{\eta \downarrow 0} \frac{1}{\eta} [\|\mathbf{D}^{-1}e^{A\eta}\mathbf{D}\|_p - 1] = \lim_{\eta \downarrow 0} \frac{1}{\eta} [\|e^{\mathbf{D}^{-1}A\mathbf{D}\eta}\|_p - 1] = \mu_p(\mathbf{D}^{-1}A\mathbf{D}) \leq 0$. ■

Remark 4.

Since the system matrix A defined by (5) for a JF TCPN is Metzler, it has a real eigenvalue, denoted by $\theta_{\max}(A)$, that dominates its spectrum in the sense that $\text{Re}\{\theta_i(A)\} \leq \theta_{\max}(A)$, $i = 1, \dots, n$, where $\theta_i(A)$ denote the eigenvalues of matrix A – see Lemma A1 in [16]. Thus, $\theta_{\max}(A)$ represents the *spectral abscissa* of matrix A , $\theta_{\max}(A) = \max\{\text{Re}(\theta_i(A)), i = 1, \dots, n\}$. Given a positive diagonal matrix D , matrices A and $\mathbf{D}^{-1}A\mathbf{D}$ have the same eigenvalues. On the other hand, $\text{Re}\{\theta_i(A)\} \leq \mu_p(A)$, for $i = 1, \dots, n$ [17]. We can conclude that if matrix A satisfies inequality (13) then $\theta_{\max}(A) \leq 0$. ■

For the usual values $p \in \{1, \infty\}$, the necessary and sufficient condition (16) from Theorem 1 can be tested by means of simple linear algebraic inequalities. Reformulating Theorem 1 for these particular norms we get the next result.

Corollary 1.

Let $\mathbf{d}, \boldsymbol{\delta} \in \mathbb{R}^n$, $\mathbf{d}, \boldsymbol{\delta} > 0$.

(i) The join-free TCPN (4) is *$\mathbf{d}$ -upper bounded* if and only if $A\mathbf{d} \leq 0$. (16)

(ii) The join-free TCPN (4) is *weakly $\boldsymbol{\delta}$ -conservative* if and only if

$$A^T \boldsymbol{\delta} \leq 0. \quad (17)$$

Proof: : It follows from the expressions of matrix measures for $p \in \{1, \infty\}$, taking into account the fact that matrix A defined by (5) is essentially non-negative. ■

The following result generalizes Theorem 4 in [16] (that corresponds to the case when $p = \infty$) and presents a situation when the non-positivity of the spectral abscissa is sufficient for the existence of invariant sets $\mathcal{X}_p^{\mathbf{d}}$ (9).

Theorem 2.

Let $\langle \mathcal{N}, \boldsymbol{\lambda} \rangle$ be a JF TCPN modeled by (4) for which $\mathcal{N}$ is a strongly connected graph with respect to the place-type nodes. The following statements are equivalent:

(i) Let $1 \leq p \leq \infty$. There exists a positive vector $\mathbf{d} \in \mathbb{R}^n$, $\mathbf{d} > 0$, so that matrix A (5) satisfies inequality (13) with

$$D = \text{diag}\{\mathbf{d}\}.$$

(ii) The spectral abscissa of matrix A (5) is non-positive:

$$\theta_{\max}(A) \leq 0. \quad (18)$$

Proof: (i) $\Rightarrow$ (ii). As commented in Remark 4, the implication is valid for any Metzler matrix.

(ii) $\Rightarrow$ (i). If $\mathcal{N}$ is a strongly connected graph with respect to the place-type nodes, then matrix A is irreducible. The right and left *Perron-Frobenius* (PF) eigenvectors, associated with its spectral abscissa $\theta_{\max}(A)$, are positive,

i.e. $\mathbf{r} = [r_1 \dots r_n]^T > 0$ and $\mathbf{l} = [l_1 \dots l_n]^T > 0$ satisfy

$$A\mathbf{r} = \theta_{\max}(A)\mathbf{r} \text{ and, respectively, } A^T\mathbf{l} = \theta_{\max}(A)\mathbf{l}.$$

These vectors are normalized so that $\sum_{i=1}^n r_i = 1$ and $\sum_{i=1}^n l_i = 1$.

For a given Hölder p -norm, $1 \leq p \leq \infty$, consider $1 \leq q \leq \infty$ such that $1/p + 1/q = 1$, where the particular cases $p = 1$ and $p = \infty$ mean $1/p = 1$, $1/q = 0$, and $1/p = 0$, $1/q = 1$, respectively. By using the left and right PF eigenvectors, construct the positive vector

$$\mathbf{d}_{PF,p}(A) = \begin{bmatrix} (r_1)^{1/q} & \dots & (r_n)^{1/q} \\ (l_1)^{1/p} & \dots & (l_n)^{1/p} \end{bmatrix}^T, \quad (19)$$

and its associated diagonal matrix

$$D_{PF,p}(A) = \text{diag}\{\mathbf{d}_{PF,p}(A)\}. \quad (20)$$

The pair $[\theta_{\max}(A), D_{PF,p}(A)]$ defines the *Perron-Frobenius p -eigenpattern* of matrix A [18]. According to Theorem 2 in [18], the following equality is fulfilled

$$\mu_p(D_{PF,p}^{-1}(A)A D_{PF,p}(A)) = \theta_{\max}(A), \quad 1 \leq p \leq \infty. \quad (21)$$

Consequently, if inequality (18) is satisfied, then the diagonal matrix $D_{PF,p}(A)$ satisfies condition (13). ■

Taking into account the proof of the previous theorem and Corollary 1 we get the following result.

Corollary 2.

Let $\langle \mathcal{N}, \boldsymbol{\lambda} \rangle$ be a JF TCPN modeled by (4) for which $\mathcal{N}$ is a strongly connected graph with respect to the place-type nodes. For matrix A (5) consider the right and left Perron-Frobenius eigenvectors, $\mathbf{r} > 0$ and, respectively, $\mathbf{l} > 0$. If the spectral abscissa of matrix A (5) satisfies inequality (18), then the join-free TCPN (4) is *$\mathbf{r}$ -upper bounded* and *weakly $\mathbf{l}$ -conservative*. ■

Even though in the analysis of dynamic linear systems the Hurwitz stability of the system matrix is a desired property, in the case of timed continuous Petri nets such a property is less sought-after. This is due to the fact that the Hurwitz stability of matrix A (5), i.e. $\theta_{\max}(A) < 0$, implies that the steady state marking of the JF TCPN modeled by (4) is $\mathbf{m}_{ss} = \lim_{\tau \rightarrow \infty} \mathbf{m}(\tau) = 0$; therefore, the steady state flow through

the transitions is $f_{ss} = \lim_{\tau \rightarrow \infty} f(\tau) = 0$. This corresponds to a deadlock in the TCPN (the steady state marking is not live). In the case when $\theta_{\max}(\mathcal{A}) > 0$, at least one of the components of the marking vector becomes unbounded – which also represents an undesirable property for a Petri net.

Definition 4.

Given a positive vector $\mathbf{b} \in \mathbb{R}^n$, $\mathbf{b} > 0$, we say that a TCPN is *$\mathbf{b}$ -lower bounded* if

$$\forall \mathbf{m}_0 \in \mathbb{R}^n : \mathbf{m}_0 \geq \mathbf{b} \Rightarrow \mathbf{m}(\tau; \mathbf{m}_0) \geq \mathbf{b}, \forall \tau \geq 0. \quad \blacksquare \quad (22)$$

Corollary 6 from [16] characterizes this property as presented beneath.

Lemma 2. [16]

Let $\langle \mathcal{N}, \lambda \rangle$ be a JF TCPN modeled by (4)&(5) and $\mathbf{b} \in \mathbb{R}^n$, $\mathbf{b} > 0$. The JF TCPN is *$\mathbf{b}$ -lower bounded* if and only if $\mathbf{b}$ is a solution to the algebraic inequality

$$\mathbf{A}\mathbf{b} \geq 0. \quad \blacksquare \quad (23)$$

Notice that if a JF TCPN is *$\mathbf{b}$ -lower bounded*, then the steady state flow through the transitions, if exists, satisfies $f_{ss} \geq \mathbf{A}\mathbf{\Pi}\mathbf{b} > 0$.

A direct consequence of the existence of a positive vector satisfying (23) is that the spectral abscissa of matrix $\mathbf{A}$ (5) is non-negative, i.e. $\theta_{\max}(\mathbf{A}) \geq 0$. The simultaneous satisfaction of inequalities (18) and (23) implies

$$\theta_{\max}(\mathbf{A}) = 0 \quad (24)$$

as a necessary condition for the existence of a live steady state marking. Therefore, a special attention must be devoted to the case when matrix $\mathbf{A}$ is singular, as this is the only convenient case, from the point of view of Petri nets applications.

If matrix $\mathbf{A}$ is singular and irreducible, then we can apply the method presented in the proof of Theorem 2 to construct sets of type (9) which are invariant with respect to the JF TCPN marking.

The following result, presented here without a proof for brevity reason, depicts two cases when inequalities (16) and (17) that characterize the *$\mathbf{d}$ -boundedness* and the *weak $\mathcal{D}$ -conservativeness* cannot have positive solutions.

Lemma 3.

Let matrix $\mathbf{A}$ be essentially non-negative and *reducible*. If one of the following conditions is met, then there exists no positive solution to inequalities (19) or (20).

(i) The index of matrix $\mathbf{A}$ is greater or equal to 2.

(ii) By suitable row and column permutations, we get

$\mathbf{PAP}^T = \begin{bmatrix} \mathbf{B} & \mathbf{0} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}$ or $\mathbf{PAP}^T = \begin{bmatrix} \mathbf{D} & \mathbf{C} \\ \mathbf{0} & \mathbf{B} \end{bmatrix}$, where $\mathbf{B}$ is an irreducible matrix with $\theta_{\max}(\mathbf{B}) = \theta_{\max}(\mathbf{A})$, and matrix $\mathbf{C}$ is non-negative, $\mathbf{C} \geq 0$, and $\mathbf{C} \neq 0$. $\blacksquare$

V. CONNECTIONS WITH THE STRUCTURAL PROPERTIES

The structural properties of Petri nets were defined in [1] for Petri nets with discrete markings and token-flows, which explains the restriction of the elements of P- and T-semiflows to integers. In the sequel, for continuous Petri nets we drop this limitation and allow P- and T-semiflows to have positive real numbers as elements.

Definition 5.

We say that a continuous Petri net topology $\mathcal{N}$ having $\mathbf{C} = \mathbf{C}^+ - \mathbf{C}^- \in \mathbb{R}^{m \times n}$ as incidence matrix is:

- *structurally bounded* iff there exists a positive vector $\mathbf{y} \in \mathbb{R}^n$, $\mathbf{y} > 0$, so that $\mathbf{C}^T \mathbf{y} \leq 0$;
- *structurally conservative* iff there exists a positive vector $\mathbf{y} \in \mathbb{R}^n$, $\mathbf{y} > 0$, so that $\mathbf{C}^T \mathbf{y} = 0$; $\mathbf{y}$ is called *P-semiflow*;
- *structurally repetitive* iff there exists a positive vector $\mathbf{x} \in \mathbb{R}^m$, $\mathbf{x} > 0$, so that $\mathbf{C}\mathbf{x} \geq 0$;
- *structurally consistent* iff there exists a positive vector $\mathbf{x} \in \mathbb{R}^m$, $\mathbf{x} > 0$, so that $\mathbf{C}\mathbf{x} = 0$; $\mathbf{x}$ is called *T-semiflow*. $\blacksquare$

Remark 5.

Let $\langle \mathcal{N}, \lambda \rangle$ be a JF TCPN modeled by (4). If the topology $\mathcal{N}$ is conservative, the positive P-semiflow $\mathbf{y} > 0$, $\mathbf{y} \in \mathbb{R}^n$, satisfies $\mathbf{A}^T \mathbf{y} = 0$ implying $\theta_{\max}(\mathbf{A}) = 0$ independent of the timing of the net expressed by λ . Moreover, the marking of the TCPN $\langle \mathcal{N}, \lambda \rangle$ satisfies $\mathbf{y}^T \mathbf{m}(\tau) = \mathbf{y}^T \mathbf{m}_0$ for all $\tau \geq 0$, relation that is similar to the conservation of the marking in a discrete Petri net. $\blacksquare$

Corollary 3.

Let $\langle \mathcal{N}, \lambda \rangle$ be a JF TCPN modeled by (4)&(5). If topology $\mathcal{N}$ is structurally bounded and the vector $\mathbf{y} > 0$ satisfies $\mathbf{C}^T \mathbf{y} \leq 0$, then the TCPN is *weakly $\mathbf{y}$ -conservative* for every timing $\lambda > 0$ (independent of the timing).

Proof. Apply Corollary 1 and take into account the fact that matrices $\mathbf{A}$ and $\mathbf{\Pi}$ are nonnegative. $\blacksquare$

The converse statement holds only if matrix $\mathbf{\Pi}^T \mathbf{A}$ has no row identical to 0 (each row of matrix $\mathbf{\Pi}^T \mathbf{A}$ has at least one non-null element) $\Leftrightarrow$ matrix $\mathbf{\Pi}^T$ has no row identical to 0 $\Leftrightarrow |p_i| \geq 1$, i.e. the net does not contain sink places.

Corollary 4.

Let $\langle \mathcal{N}, \lambda \rangle$ be a JF TCPN modeled by (4)&(5) for which $\mathcal{N}$ is a strongly connected graph with respect to the place-type nodes. Topology $\mathcal{N}$ is structurally conservative, if and only if there exist positive vectors $\mathbf{b}, \mathbf{d} \in \mathbb{R}^n$, $\mathbf{b}, \mathbf{d} > 0$, so that the net is *$\mathbf{b}$ -lower bounded* and *$\mathbf{d}$ -upper bounded*.

Proof. Apply Corollary 3 and Remark 5. Note that in this case we have $\theta_{\max}(\mathbf{A}) = 0$. $\blacksquare$

Corollary 5.

Let $\langle \mathcal{N}, \lambda \rangle$ be a JF TCPN modeled by (4)&(5). If there exists $\mathbf{b} \in \mathbb{R}^n$, $\mathbf{b} > 0$, so that the net is $\mathbf{b}$ -lower bounded then the topology $\mathcal{N}$ is structurally repetitive. The converse statement stands only if the net is also choice-free

Proof. If the net is $\mathbf{b}$ -lower bounded, i.e. $\mathbf{b} > 0$ satisfies $\mathbf{A}\mathbf{b} \geq 0$, then the vector $\mathbf{x} = \mathbf{\Lambda}\mathbf{\Pi}\mathbf{b}$ is positive (due to the structure of matrix $\mathbf{\Pi}$ and since matrix $\mathbf{\Lambda}$ is diagonal with positive elements) and satisfies $\mathbf{C}\mathbf{x} \geq 0$.

Assume that the topology $\mathcal{N}$ is structurally repetitive with $\mathbf{x} > 0$ satisfying $\mathbf{C}\mathbf{x} \geq 0$. There exists a positive vector $\mathbf{b} > 0$ so that $\mathbf{x} = \mathbf{\Lambda}\mathbf{\Pi}\mathbf{b} \Leftrightarrow \mathbf{\Pi}\mathbf{b} = \mathbf{\Lambda}^{-1}\mathbf{x} \Leftrightarrow \mathbf{b} = \mathbf{C}^{-}\mathbf{\Lambda}^{-1}\mathbf{x}$ only if $|p_i| \leq 1$, i.e. the net is choice-free (due to the structure of matrix $\mathbf{\Pi}$, since, in this case, $\mathbf{C}^{-}\mathbf{\Pi} = \mathbf{I}_n$) and $\mathbf{A}\mathbf{b} \geq 0$. ■

Corollary 6.

Let $\langle \mathcal{N}, \lambda \rangle$ be a JF TCPN modeled by (4)&(5) for which $\mathcal{N}$ is a strongly connected graph with respect to the place-type nodes. If there exist positive vectors $\mathbf{b}, \mathbf{d} \in \mathbb{R}^n$, $\mathbf{b} > 0$, $\mathbf{d} > 0$, so that the TCPN is $\mathbf{b}$ -lower bounded and $\mathbf{d}$ -upper bounded then topology $\mathcal{N}$ is structurally consistent. The converse statement stands only if the net is also choice-free.

Proof. In this case matrix $\mathbf{A}$ is irreducible and satisfies $\theta_{\max}(\mathbf{A}) = 0$. Let $\mathbf{r} \in \mathbb{R}^n$, $\mathbf{r} > 0$, be its right PF eigenvector, i.e. $\mathbf{A}\mathbf{r} = 0$. The vector $\mathbf{x} = \mathbf{\Lambda}\mathbf{\Pi}\mathbf{b}$ is positive (due to the structure of matrix $\mathbf{\Pi}$ and since matrix $\mathbf{\Lambda}$ is diagonal with positive elements) and satisfies $\mathbf{C}\mathbf{x} = 0$, showing that $\mathcal{N}$ is structurally consistent.

For the converse part, the same way as in the proof of Corollary 5, if the net is choice-free then, for $\mathbf{x} > 0$ a T-semiflow of $\mathcal{N}$, the vector $\mathbf{b} = \mathbf{C}^{-}\mathbf{\Lambda}^{-1}\mathbf{x}$ is positive, $\mathbf{b} > 0$, and satisfies $\mathbf{A}\mathbf{b} = 0$; we can choose $\mathbf{d} = \gamma\mathbf{b}$ with $\gamma \geq 1$. ■

VI. CONCLUSIONS

The paper introduces a new concept of boundedness for timed continuous Petri nets with infinite server semantics based on the existence of invariant sets with respect to the marking of the net, sets that are defined by weighted Hölder p -norms. For the join-free topology we give a necessary and sufficient condition for the existence of such invariant sets. Some connections with the structural properties associated to the net topology are also pointed out.

ACKNOWLEDGMENT

The authors thank Professor Octavian Pastravanu for his constructive comments and suggestions.

REFERENCES

- [1] T. Murata, Petri nets: properties, analysis and applications. In: *Proc. of the IEEE*, vol. 77, no. 4, pp. 541-580, 1989.
- [2] R. David, and H. Alla, Continuous Petri nets. In *EWATPN 1987: 8th European Workshop on Application and Theory of Petri Nets*, Zaragoza, Spain, 1987, pp. 275-294.

- [3] R. David, and H. Alla, Autonomous and timed continuous Petri nets, In: *Advances in Petri Nets, Lecture Notes in Computer Science*, vol. 674, pp. 71-90, 1993.
- [4] L. Recalde, and M. Silva, Petri Nets fluidification revisited: Semantics and steady state, *Journal européen des systèmes automatisés*, vol. 35, no. 4, pp. 435-449, 2001.
- [5] M. Silva, and L. Recalde, Petri nets and integrality relaxations: A view of continuous Petri nets, *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 32, no. 4, pp. 314-327, 2002.
- [6] C. Mahulea, L. Recalde, and M. Silva, Basic server semantics and performance monotonicity of continuous Petri nets, *Discrete Event Dynamic Systems: Theory and Applications*, vol. 19, no. 2, pp. 189-212, 2009.
- [7] C. Mahulea, L. Recalde, and M. Silva, Observability of continuous Petri nets with infinite server semantics, *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 219-232, 2010.
- [8] C.R. Vazquez, A. Ramirez, L. Recalde, and M. Silva. On controllability of timed continuous Petri nets. *Lecture Notes in Computer Science*, vol. 4981, pp. 528-541, 2008.
- [9] C. Mahulea, A. Giua, L. Recalde, C. Seatzu, and M. Silva, Optimal model predictive control of timed continuous Petri nets. *IEEE Trans. Automatic Control*, vol. 53, pp. 1731-1735, 2008.
- [10] J. Julvez, and R.K. Boel, A Continuous Petri net approach for model predictive control of traffic systems, *IEEE Trans. Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 40, no. 4, 2010.
- [11] C.R. Vazquez, and M. Silva, Timing and liveness in continuous Petri nets, *Automatica*, vol. 47, pp. 283-290, 2011.
- [12] A. Amrah, N. Zerhouni, and A. El Moudni, On the control of manufacturing lines modeled by controlled continuous Petri nets, *Int. J. Syst. Science*, vol. 29, no. 2, pp. 127 – 137, 1998.
- [13] D. Gilbert, and M. Heiner, From Petri nets to differential equations - An integrative approach for biochemical network analysis. *Lecture Notes in Computer Science*, vol. 4024, pp. 181-200, Springer, 2006.
- [14] H.N. Pavel, *Differential Equations: Flow Invariance and Applications*, Pitman, Boston, 1984.
- [15] F. Blanchini, and S. Miani. *Set-Theoretic Methods in Control*, Birkhäuser: Boston, Basel, Berlin, 2008.
- [16] M.H. Matcovschi, and O. Pastravanu, Flow-invariant sets with respect to the markings of timed continuous Petri nets, *Proc. 2005 European Control Conference*, pp. 7726 – 7731, 2005.
- [17] C.A. Desoer, and M. Vidyasagar, *Feedback Systems: Input-Output Properties*, Academic Press, New York, 1975.
- [18] M.H. Matcovschi, and O. Pastravanu, Perron-Frobenius theorem and invariant sets in linear systems dynamics, *Proc. 15th Mediter. Conf. Control and Automation MED'07*, pp.1-6, 2007.

APPENDIX. NOTATIONS AND NOMENCLATURE

- For a vector $\mathbf{x} = [x_1 \dots x_n]^T \in \mathbb{R}^n$: $\|\mathbf{x}\|_p$ is the Hölder vector p -norm defined by $\|\mathbf{x}\|_p = [|x_1|^p + \dots + |x_n|^p]^{1/p}$ for $1 \leq p < \infty$, and by $\|\mathbf{x}\|_\infty = \max_{1 \leq i \leq n} |x_i|$ for $p = \infty$.
- For a square matrix $\mathbf{M} = [m_{ij}] \in \mathbb{R}^{n \times n}$: for $1 \leq p \leq \infty$, $\|\mathbf{M}\|_p$ is the matrix norm induced by the vector p -norm through $\|\mathbf{M}\|_p = \sup_{\mathbf{x} \neq 0} \frac{\|\mathbf{M}\mathbf{x}\|_p}{\|\mathbf{x}\|_p} = \max_{\|\mathbf{x}\|_p=1} \|\mathbf{M}\mathbf{x}\|_p$, and $\mu_p(\mathbf{M}) = \lim_{h \downarrow 0} \frac{1}{h} (\|\mathbf{I} + h\mathbf{M}\|_p - 1)$ is the matrix measure [17], based on the matrix norm $\|\bullet\|_p$. For $p \in \{1, \infty\}$ we get:

$$\|\mathbf{M}\|_1 = \max_{1 \leq j \leq n} \left\{ \sum_{i=1}^n |m_{ij}| \right\}, \quad \mu_1(\mathbf{M}) = \max_{1 \leq j \leq n} \left\{ m_{jj} + \sum_{i=1, i \neq j}^n |m_{ij}| \right\},$$

$$\|\mathbf{M}\|_\infty = \max_{1 \leq i \leq n} \left\{ \sum_{j=1}^n |m_{ij}| \right\}, \quad \mu_\infty(\mathbf{M}) = \max_{1 \leq i \leq n} \left\{ m_{ii} + \sum_{j=1, j \neq i}^n |m_{ij}| \right\}.$$

Control law for Timed Marked Graph constrained by Marking Exclusion Constraint Or/And

Maen ATLI, Zied ACHOUR, Alexandre SAVA and Kondo H.ADJALLAH

LGIPM, INRIA Cousteau

Ecole National d'Ingénieur de Metz

Metz, France

maen.atli@inria.fr; zied.achour@univ-metz.fr; sava@enim.fr; adjallah@enim.fr

Abstract—we have already presented a contribution for synthesizing an approach of supervisory control for Discrete Event Systems (DES) modeled by Timed Place Marked Graphs subject to *Marking Exclusion Constraint*. This paper is continuation work in this area for building the control law for Discrete Event Systems (DES), where the time is taken in consideration. It solves a forbidden state problem characterized by one of the both kinds of Marking Exclusion Constraint: MEC-Or and MEC-And. A computationally efficient technique to build the supervisor is proposed, which we take in consideration the uncontrollable and/or unobservable nature of some events. From the initial state (as known marking), we build time-tables that show us the timeline of tokens distribution over the system. These time-tables help us to build a Control Law Table (CLT), by analyzing this table the supervisor may determine the moment of its intervention.

Keywords-component; Discret Event System; control law; Marking Exclusion Constraint; Petri net and Timed Marked Graph.

1. Introduction

Our research is concerned on the control synthesis of manufacturing system. A manufacturing system can be modeled by a discrete event system DES. Apart from planning (where people work with ratios of products fabricated per week or per day), any modeling has to be based on the concepts of event and activities. An event corresponds to a state change. An activity is a black-box encapsulating what is occurring between two events. When using Petri Nets, events are associated with transitions, and activities with places. However Petri Net is not the unique tool handling events and activities. For example, algebras and formal models [14] are under investigation. This paper deals with the control synthesis problem for Discrete Event System modeled by a class of Timed Petri Net called Timed Marked Graph subject to Marking Exclusion Constraint MEC. The main contribution is to synthesize a supervisor that acts in closed loop with the system, where the whole system/supervisor avoid the forbidden states modeled by MEC. In the context of this synthesis, we propose a technique to build a controller that enforces MEC on partially controllable and/or observable marked graph. The time makes our control policies more permissive by allowing the system to reach more states (marking) which were considered as dangerous marking whether we do not take the time in account. In the sequel, we

recall briefly the approaches which are most related to our research work. According to a chronology, we are going firstly to pass on the researches covering the control synthesis for Discrete Event Systems, and the Petri Net models used for this purpose.

1.1. Control Synthesis and linear specifications:

A Discrete Event System *DES* is a dynamic system that involves in accordance with the abrupt occurrence of physical events at possibly unknown time intervals. Given a discrete event model of the plant and a specification of the desired behaviour, the aim of the control synthesis is to build the appropriate controller that acts in closed-loop with the plant in order to guarantee a desired behaviour. The existence and the characterization of the maximally permissive control were formalized in [14] and [15]. Finite-state machines and formal languages are the modelling framework considered in these studies and the control action is achieved by event feedback. The drawback of automata models is their lack of structure, which makes difficult the representation of behaviours like synchronization, parallelism and resource sharing. However, to overcome these disadvantages, control synthesis methods based on Petri nets were proposed by [6] and [5] to exploit the modelling power of PN and the rich mathematical results characterize them. Over the last decades *Petri Net* has gained increased usage and acceptance as basic model of DES [18]. Petri Net provides an intuitive graphical representation for DES behaviour. Modelling facilities together with interesting analytical properties made this tool appropriate to controller synthesis for DES. The subclass of Petri nets which each place in the net has exactly one incoming arc and exactly one outgoing arc, is known Marked Graph.

In [10], the authors present an efficient method for synthesizing feedback control logic to solve forbidden-state problems for DES modelled by safe controlled marked graphs. The distributed representation of the DES state in terms of the controlled marked graph marking allows an efficient specification of the forbidden states in terms of individual place marking. The authors synthesize a state feedback control which guarantees that the forbidden states are avoided while allowing a maximal set of state transitions to occur. This approach is extended in [11]; by taking into account the liveness property for the closed loop system under the maximally permissive feedback control policies. Using the

concept of synchronic distances in Petri nets, they provide sufficient conditions to build a maximally permissive controller. The forbidden state problem has been widely addressed and covers. An interesting class of specifications called generalized mutual exclusion constraints *GMEC* is presented in [1]. These are a natural way of expressing the concurrent use of a finite number of resources, shared among different processes. A control synthesis technique to enforce *GMEC* specifications based on monitor places is also proposed. They present two PN structure capable of enforcing these constraints [2]. The first is a monitor-based solution and the second one is a supervisory-based solution. They deal with the problem of enforcing generalized mutual exclusion constraints on marked graphs with control safe places. In the same way, the paper [19], propose an algorithm to calculate the optimal solution for nets with acyclic uncontrollable subnets. The controller has to solve online at each step linear integer program. While [9] propose a method to build a Petri net controller based on place invariants to enforce linear constraints on the Petri net marking in case of observable and controllable transitions.

In [4], the authors develop the significations used in [2], and they extended the work of [10] and [11], to construct controller logic for forbidden-state problems based on marked graphs. They deal with the forbidden-state problem of general marked graphs with uncontrollable transitions. The models need not to be safe or cyclic as for the references [10] and [11]. Control requirements are expressed as the conjunction of *Generalized Mutual exclusion constraints GMEC*. They [3], present two supervisory control problems of plant Petri net models, forbidden state and forbidden state transition. Based on the theory of regions, algebraic characterizations of control places are provided for the both problems.

Another basic specification was presented in [12] and [13]; this basic specification is called Marking Exclusion Constraint MEC. They define MEC as a basic constraint item applied on place markings. Thus a MEC focuses on the marking of one place. For instant a MEC constraint may request the marking of a place p to be different from a given value k . Markings constraints of several places may be modelled by using MEC as a basic item. In this case, they propose three kinds MEC-Or, MEC-And and M-MEC. They define MEC-Or a constraint by a union of MECs, and MEC-And a constraint by a conjunction of MECs. We also extend MEC to the case where a weighted sum of a given Petri net marking must be different from a given number of tokens k , we call it M-MEC. In the same context, they built a control synthesis to restrict a *Marked Graph* to respect the specification modelled by a MEC or one of kinds of MEC.

1.2. Control Synthesis of Timed Petri Net:

In order to study the performance aspects of Petri Nets models, we have to include the duration of activities into the model. The two main extensions of Petri net with time are Time Petri Nets [17] and Timed Petri Nets [7]. While the transition may fire within a time interval in for Time Petri Net, the transition in Timed Petri Net fires as soon as possible, according a delay associated to places (Timed Transition Petri Net TTPN) or transition (Timed Placer Petri Net TPPN). The

two subclasses TTPN and TPPN are expressively equivalent. The firing times may be deterministic, random variables described by a probability distribution function, fuzzy variable described by a fuzzy interval. The authors in [20] extended the work of [4], by considering unobservable transitions. They proposed approach based on an observer. The same authors in [21] have extended their work to take in account the time information. They discuss the influence of observability and time constraints on the controller synthesis and they show that the control law designed using time information is more permissive. The problem of observability has no influence in this case, since even we can not observe the firing, time information help us to deduce the their firing instant. In our work [22]; we present a control synthesis approach for Discrete Event Systems (DES) modeled by Timed Place Marked Graphs TPMG where the time is associated with places. It solves a forbidden state problems characterized by Marking Exclusion Constraint (MEC). A computationally efficient technique to build the supervisor is proposed, which we take in consideration the uncontrollable and/or unobservable nature of some events. In this paper, we develop our work in [12], [13] and [23], by discussing the same synthesis to include the both kind of MEC: MEC-And and MEC-Or.

1.3. Summary of content:

Following the algebraic model of Petri nets and a brief summary of the basic linear specifications *Marking Exclusion Constraint MEC* are in section 2. The supervisory control review and our primary contribution of this paper appear in section 3. An Example is introduced in section 4. Finally, we summarized our contributions of this paper and the future research regarding these contributions.

2. Background:

2.1. Modeling with Petri Net and Marked Graph:

A Petri net PN is a graphical and mathematical modeling tool particularly adapted to modeling and analysis of discrete event systems. A complete presentation can be found in [18]. Similarly, the set of input (output) places of a transition t is denoted ${}^{\circ}t$ (t°). A Petri net is said to be ordinary if the weight associated to each arc is equal to 1. A Petri net marking is a mapping $M: P \rightarrow N$ that assigns to each place of a Petri Net a non-negative integer number of tokens, where the tokens are represented by black dots [18]. The following notations will be used throughout this paper:

P	Set of m places
T	Set of n transitions
Pre	Pre-incidence function that defines weighted arcs from places to transitions.
$Post$	Post-incidence function that defines weighted arcs from transitions to places.
${}^{\circ}p$ (respectively p°)	Set of input (respectively, output) transitions of place p .
${}^{\circ}t$ (respectively t°)	Set of input (respectively, output) places of transition t .
$M(p)$	Marking of one place p .
M	Petri net marking (M_0 is the initial marking)

$M[t >]$	represent the markings courant for all places.
$M[t > M']$	Transition t is fireable at marking M .
	Firing transition t at marking M leads to marking M' .
$\sigma(t)$	Number of firing's times of transition t .
$R(N, M_0)$	Set of reachable markings from M_0 .
$R^u(N, M_0)$	The set of reachable markings from a marking M_0
(respectively	by firing only uncontrollable transitions
$R^{uo}(N, M_0)$	(respectively unobservable transitions).

A *Marked Graph* is an ordinary PN where all weights associated to arcs are 1 and every place has exactly one input transition and one output transition, i.e., $|^{\circ}p| = |p^{\circ}| = 1$ for all $p \in P$. The marking of place $M(p)$ in marked graph is changed by firing the input and output transition [4], as the following equation:

$$M(p) = M_0(p) - \sigma(^{\circ}p) + \sigma(p^{\circ}) \quad (1)$$

Firing transition $^{\circ}p$ or p° at marking M_0 leads to marking M .

2.2. Marking Exclusion Constraint MEC:

A *Marking Exclusion Constraint* [12] and [13] is defined as a condition that forbids the existence of a given number of tokens in one critical place. Let a Petri net with reachable marking $R(N, M_0)$. A single *Marking Exclusion Constraint* MEC defines a set of legal markings described as following:

$$M_L = \{M \in R(N, M_0) \mid q \in Q_{mc} \mid M(q) \neq k\} \quad (2)$$

Where q is called *critical place* which is subject to a MEC constraint, $Q_{mc} \subseteq P$, is the set of critical places.

We distinguish two principal kinds of this specification for more than one critical place, MEC-Or and MEC-And.

Definition: Let a Petri net with reachable marking $R(N, M_0)$. A *Marking Exclusion Constraint* of type MEC-Or defines a set of legal markings described as following:

$$M_L = \left\{ M \in R(N, M_0) \mid \bigvee_{q_i \in Q_{mc}} M(q_i) \neq k_i \right\}$$

Definition: Let a Petri net with reachable marking $R(N, M_0)$. A *Marking Exclusion Constraint* of type And MEC-And defines a set of legal markings described as following:

$$M_L = \left\{ M \in R(N, M_0) \mid \bigwedge_{q_i \in Q_{mc}} M(q_i) \neq k_i \right\}$$

2.3. Timed Marked Graph TMG:

Dealing with time in a MG is accomplished by assigning time elapses to transitions or places. The time can be assigned to both nodes but the analysis of TMG models is simpler when the time is attached to one kind of node. In this paper, we assign the time to transition; the resulted Marked Graph in this case called Timed Transition Marked Graph TTMG. TTMG is defined as (N, H, θ) , where N is a normal Petri Net and H is a set of delays (durations); $H = \{h_i\}$ for $i = 1, \dots, m$, each $h_i \in \mathbb{N}$ (Natural number). Each h_i represents the time that the arriving tokens to place p_i stay unavailable. θ_i specifies the firing instant of transition t_i . A token becomes available after h_i time units from the instant of its arrival to p_i . Only the available tokens enable the transitions firing. In other words, a token in

a place p_i , may have two possible states: available or unavailable. In any time delay h , the present marking $M = Ma + Mu$, such that Ma comprises available tokens and Mu comprises unavailable tokens. A transition is enabled if $Ma(p_i) \geq Pre(p_i, t)$, where Ma encompasses the available tokens. When a token arrives in a timed place p_i , it must remain there for a time delay h_i , before becoming available.

3. Supervisor synthesis:

3.1. Controllability problem:

A forbidden marking is a marking that violates at least one MEC constraint. The set of forbidden markings is:

$$M_f = \{M \in R(N, M_0) \mid \omega M(q_i) = k_i\} \quad (3)$$

The transition set is partitioned into two disjoint subsets: $T = T_c \cup T_u$ & $T_c \cap T_u = \emptyset$, where T_c is the set of controllable transitions and T_u is the set of uncontrollable transitions. A direct consequence of the presence of uncontrollable transitions is the existence of dangerous markings. A marking is called dangerous if the starting from this marking, the system may evolve uncontrollably to a forbidden marking. The set of dangerous markings is:

$$M_D = \{M \in R(N, M_0) \mid M[U_{zero}, \sigma > M', M' \in M_f]\} \quad (4)$$

The set of admissible markings, denoted M_a , is defined by the markings which are neither dangerous nor forbidden. The set M_a is the maximal set of states under which there exists a control policy which will prevent the system from reaching a forbidden state $M' \in M_f$.

An *influence path* $\pi(q)$ of a critical place is an elementary path joining t to q such that $t \in T_c$ is a controllable transition and all other transitions in the path are uncontrollable. The controllable transition t of the influence path is called an *influence transition*. The control synthesis approach that we propose relies on the concept of binary control of controllable transition given in [11]. A binary control is a mapping $U: T_c \rightarrow \{0, 1\}$, which assigns a binary value count to each controllable transition. A controllable transition t is allowed fire if $U_{(t)} = 1$.

Let $D(q, M)$ denote the maximum number of times ($^{\circ}q$) may fire without firing any influence transition in $Z(q)$, the influence zone of q , i.e. under U_{zero} .

$$D(q, M) = \min td(M, t' \in T_c, ^{\circ}q) \quad (5)$$

Where $td(M, t' \in T_c, ^{\circ}q)$ be the token distance (the sum of tokens) between an influence transition t' and $^{\circ}q$ (the input transition of a critical place q) at marking M . The influence zone $Z(q)$ of a critical place q is the subnet containing all nodes for which there exists a directed path from an influence transition to q . In this paper, we denote the set of influence transitions by $\Gamma(q)$, it is formally defined by:

$$\Gamma(q) = \{t \in (Z(q) \cap T_c)\} \quad (6)$$

Let $Y(q, M)$ [4] be the marking having the maximal number of tokens in q which may be reached uncontrollably from marking M .

Then $Y(q, M)$ is updated according to transitions firing.

$$Y(q, M) = M_0(q) + D(q, M) + \sigma(q) - \sigma(q^\circ) \quad (7)$$

Where $\sigma(q)$ (respectively $\sigma(q^\circ)$) defines the number of times of transition (q) (respectively (q°)) is fired.

3.2. Supervisory synthesis for Timed Marked Graph:

In this section, we present the idea discussed in [21] and [22]. The supervisor must evaluate the worst case noted by $Y(q, M, \theta)$ which represents the number of tokens that place q could have after a global time θ by starting from a known marking M and firing only uncontrollable transitions. The marking of q is influenced by the marking of the path $\pi(q)$ such as $\pi(q) = t_{\pi_1}^c p_1 t_2 p_2 \dots p_n t_{n+1} q$, where t_c represents the only controllable transition (influence transition) in the path.

When considering the time information, tokens can no longer stay indefinitely in places. A token cannot get to a given place before a certain period of time and has to live it before another given period of time. Consequently, the behaviour of the system is restricted regarding the untimed case and some potentially dangerous situations do not need to be considered anymore when building the controller. Thus, using time information enables to build more permissive control laws.

$$\text{They define: } Y(q, M, \theta) = Y^+(q, M, \theta) - Y(q, M, \theta) \quad (8)$$

Where $Y^+(q, M, \theta)$ (resp. $Y(q, M, \theta)$) represents the number of tokens which arrive to the place q in a period θ (resp. represents the number of tokens which leave the place q in a period θ).

$$Y^+(q, M, \theta) = M(q) + M(p_n) \text{ if } \theta \geq h(\pi_n) + \dots + M(p_1) \text{ if } \theta \geq h(\pi_1) + \sigma(t_1, \theta - e - h(\pi_1)) \quad (9)$$

$$Y(q, M, \theta) = M(q) \text{ if } \theta \geq h(q) + M(p_n) \text{ if } \theta \geq h(\pi_n) + h(q) + \dots + M(p_1) \text{ if } \theta \geq h(\pi_1) + h(q) + \sigma(t_1, \theta - e - (h(\pi_1) + h(q))) \quad (10)$$

Where π_i represent the paths $\pi(pi \rightarrow^\circ q)$, $\sigma(t_i, \tau)$ is the firing times of t_i during the period τ , and e is the moment of influence transition firing.

The supervisor will not tamper with the evolution of system behaviour whether this behaviour respect always $Y(q, M, \theta) \neq k$, otherwise the supervisor will tamper only with the controllable transition firing (either influence transition or output transition). The control law for this case is described as following:

$$U_{(t \in T_c)} = \begin{cases} 1, & \text{if } Y(q, M, \theta) \neq k \\ 0, & \text{otherwise} \end{cases} \quad (I)$$

In the sequel of this section, we recall the significations and rules [22], which are indispensable in this paper:

Supervisor work rule 1: The task of supervisor is to observe the work of system and to control the firing of controllable transition (to fire or to not fire). i.e. the supervisor prevents only the firing when it is necessary. From the

marking M , the supervisor builds the CLT for the period $[0, h(\pi) + h(q)]$.

The supervisor prevents the output controllable transition q° to fire:

1. At the moment θ_i , if the logic value of control law is false at θ_i .
2. At the moment θ_j , if at least one of the three following conditions is not validated:
 - There is an available token or more in the critical place at θ_i .
 - The logic value of control law is true at θ_j .
 - $Y(q, M, \theta_i) < k$, where $\theta_i > \theta_j$, $\theta_j \in [0, \theta_i]$ [and θ_j : the moment before θ_i when the value of control law is false.

Supervisor work rule 2: the supervisor allows the influence transition to fire at $\theta = e$, if all values control law in CLT over $[e, e + h(\pi_1) + h(q)]$ are true.

Supervisor work rule 3: At the firing of an influence transition at $\theta = e$, and the both conditions are validates:

1. There is at least a value false in CLT over $[e, e + h(\pi_1) + h(q)]$ at θ_i .
2. $Y(q, M, \theta_i - 1) > Y(q, M, \theta_i)$.

The supervisor has two choices:

- Either, the supervisor prevents the influence transition to fire at $\theta = e$.
- Or, the supervisor permits the influence transition to fire at $\theta = e$, provided that it prevents the controllable output transition to fire at θ_i .

For more information about these three rules, you may review the paper [22].

3.3. Control Law Tables:

The following is an algorithm to compute the control law tables for building the supervisory control for discrete event systems DESs. We use the control law table [22] to compute the control law at each moment. We distinguish two cases, the first one is without firing any controllable transition (at the initial marking), and the second one whether the system is going to fire one controllable transition.

- At the initial marking M_0 :

1. For each critical place: build a *Control Law Table* CLT as shown in table 1 over $[0, h(\pi_1) + h(q)]$. Use the *supervisor work rule 1* to compute the control law value.

TABLE 1.1. CALCULATING $Y^+(q_i, M_0, \theta)$

θ	$M(q_i)$		$Y^+(q_i, M_0, \theta)$
0			
...			
$h(\pi_1) + h(q)$			

TABLE 1.2. CALCULATING $Y(q_i, M_0, \theta)$

θ	$M(q_i) \text{ if } \theta \geq h(q)$		$Y(q_i, M_0, \theta)$
0			
...			
$h(\pi_1) + h(q)$			

TABLE 1.3. CONTROL LAW TABLE CLT

θ	$Y^+(q_i, M_0, \theta)$	$Y(q_i, M_0, \theta)$	$Y(q_i, M_0, \theta)$	U_i
----------	-------------------------	-----------------------	-----------------------	-------

0				
...				
$h_{(\pi 1)}+h_{(q)}$				

2. Build the *Control Law Table General* CLTG as shown in table 2 for MEC-Or specification, and as shown in table 3 for MEC-And specification.

TABLE 2. CONTROL LAW TABLE CLTG FOR MEC-OR SPECIFICATION

θ	U_l		U_{mc}	$U = U_l \vee \dots \vee U_{mc}$
0				
...				
$h_{(\pi 1)}+h_{(q)}$				

TABLE 3: CONTROL LAW TABLE CLTG FOR MEC-AND SPECIFICATION

θ	U_l		U_{mc}	$U = U_l \wedge \dots \wedge U_{mc}$
0				
...				
$h_{(\pi 1)}+h_{(q)}$				

3. All the controllable transitions are fireable whether the control law value (last column in CLTG) is true (one).
4. If the control law value takes the value zero at the moment θ_i :

- The specification is MEC-Or: it is sufficient to suspend one of the controllable output transition ($q_i \in T_c$)
- The specification is MEC-And: All the controllable transitions are suspended at this moment.

• Firing $t \in T_c$ at the moment e :

- For each critical place: build a *Control Law Table* CLT as shown in table 1 over $[e, e+h_{(\pi 1)}+h_{(q)}]$.
- Build the *Control Law Table General* CLTG.
- All the controllable transitions are fireable whether all values over last column in CLTG are true (rule 2).
- If the control law value takes the value zero at the moment θ_i :
 - The specification is MEC-Or: it is sufficient to check the rule 3 for one critical place.
 - The specification is MEC-And: it must check the rule 3 for all critical places.

4. Illustrative Example:

The production line in our example (figure 1) consists of two cellules, the first one is consisted of a tank of liquid with unlimited capacity, and a pump that fill a litre for each once, and the second one is a conveyor to transport bottles with a capacity of one bottle. The timed marked graph model for this example is presented in figure 1, and the table 4 provides us the description of places, transitions and the associated time delays. The initial state of this example is: $M_0=[3 \ 1 \ 0 \ 3 \ 1 \ 0 \ 0]$.

TABLE 4: INTERPRETATION AND DELAYS OF PLACES AND TRANSITIONS

P	Interpretation	Delay
p_1	empty bottle (bottles) is (are) on the conveyor	Untimed
p_2	The empty Bottle is exactly under the valve	2 time units

	of pump (in the space of pumping).	
p_3	Capacity the space of pumping.	Untimed
p_4	Tank	Untimed
p_5	Liquid of one litre is ready to be pumped.	1 time unit
p_6	Capacity of pump.	Untimed
p_7	Pumping the liquid	2 time unit
T		
t_1	Arriving a new bottle to the conveyer.	Untimed
t_2	Arriving a bottle to the space of pumping.	Untimed
t_3	Exiting the bottle from the space of pumping.	Untimed
t_4	Fill a liquid to the tank.	Untimed
t_5	Empty one litre from the tank.	Untimed
t_6	The pump is ready to work	Untimed
t_7	Finish the work of pump.	Untimed

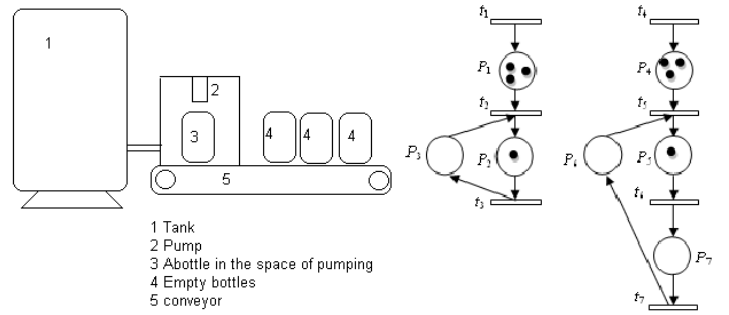


Figure 1. Timed Marked Graph model for example shown in figure 1.

Through the evolution of system, it is necessary to confirm that the pump won't work whether there is no empty bottle in the space of pumping, on the other side; it is possible to have a bottle in this area although the pump has no liquid to fill. In other words, all markings having one token in p_7 and no token in p_2 , are forbidden. But the set of markings where there is a token in p_2 and no token in p_7 are not forbidden. The set of critical places is: $Q_{mc} = \{p_2, p_7\}$, the set of forbidden markings is given: $M_f = \{M \in R(N, M_0) / M(p_2) \neq 1 \vee M(p_7) \neq 0\}$.

Table 5.1. Calculating $Y^*(p_2, M_0, \theta)$

θ	$M(p_2)$	$Y^*(p_2, M_0, \theta)$
0	1	1
1	1	1
2	1	1

TABLE 5.2. CALCULATING $Y(p_2, M_0, \theta)$

θ	$M(p_2)_{if \theta \geq 2}$	$Y(p_2, M_0, \theta)$
0	0	0
1	0	0
2	1	1

TABLE 5.3. CONTROL LAW TABLE CLT FOR p_2

θ	$Y^*(p_2, M_0, \theta)$	$Y(p_2, M_0, \theta)$	$Y(p_2, M_0, \theta)$	U_l
0	1	0	1	1
1	1	0	1	1
2	1	1	0	0

TABLE 6.1. CALCULATING $Y^+(p_7, M_0, \theta)$

θ	$M(p_5)$ if $\theta=1$	$M(p_7)$	$Y^+(p_7, M_0, \theta)$
0	0	0	0
1	1	0	1
2	1	0	1
3	1	0	1

TABLE 6.2. CALCULATING $Y(p_7, M_0, \theta)$

θ	$M(p_5)$ if $\theta=3$	$M(p_7)$ if $\theta=2$	$Y(p_7, M_0, \theta)$
0	0	0	0
1	0	0	0
2	0	0	0
3	1	0	1

TABLE 6.3. CONTROL LAW TABLE CLT FOR p_7

θ	$Y^+(p_7, M_0, \theta)$	$Y(p_7, M_0, \theta)$	$Y(p_7, M_0, \theta)$	U_2
0	0	0	0	1
1	1	0	1	0
2	1	0	1	0
3	1	1	0	1

TABLE 7: CONTROL LAW TABLE CLTG FOR MEC-OR SPECIFICATION

θ	U_1	U_2	$U = U_1 \vee U_2$
0	1	1	1
1	1	0	1
2	0	0	0
3	0	1	1

From the table 7, we may see that our specification won't be respected at $\theta=2$, the pump will let one litre of liquid to go out without a bottle under the valve of this pump, therefore the supervisor must take many choices at this moment. For example it is sufficient that the value of U_1 become true, therefore the supervisor will prevent the transition t_3 to fire at $\theta=2$, and let it to fire at the next step.

5. Conclusion:

This paper is considered as continuation of our work presented in [22]. We present a control synthesis for DES modelled by *Timed Place Marked Graph* TPMG subject to a specification modelled by one of two kind (Or/And) of a *Marking Exclusion Constraint* MEC. We deal with uncontrollable and/or unobservable transition, where there is no deference in this case between the two kinds of transitions because the time information helps us to determine the current marking. In our control policy, we depend on the initial marking as known marking to build time tables that show us the timeline of marking distribution over the system. These time-tables help to build a Control law Table CLT, through analysis the information in this table the supervisor may determine the moment of its intervention. The goal of supervisor in this paper is to observe the system behaviour and to tamper with this behaviour when it is necessary. We present three *Supervisor work rules*, the first one to describe the conditions to control the output transition, and two rules for the influence transitions. Future work is to develop the control synthesis to include Time Marked Graph (the node is attached with time interval).

- [1] A. Giua, F. DiCesare, and M. Silva (1992). Generalized Mutual Exclusion Constraints on Nets with Uncontrollable Transitions. In Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics (Chicago, USA), pp. 974-999.
- [2] A. Giua, F. DiCesare, and M. Silva (1993). Petri Net supervisors for generalized mutual exclusion constraints. In roc. 12th IFAC World Congress (Sidney, Australia), Vol. 1, pp. 267-270.
- [3] A. Ghaffari, N. Rezg, and X. Xie (2002). Algebraic and geometric characterization of Petri net controllers using the theory of regions. Discrete Event Systems. Proceedings. Sixth International Workshop, Pp 219-224.
- [4] A. Ghaffari, N. Rezg, and X. Xie (2003). Feedback Control Logic for Forbidden-State Problems of Marked Graphs: Application to a Real Manufacturing System. IEEE Trans. Automat. Contr., vol. 48, pp. 18-29.
- [5] A. Ishikawa, and K. Hiraishi (1987). Analysis and control of discrete event systems represented by Petri nets. In P. VARAIYA and A. B. KURZHANSKI (eds), Discrete Event Systems: Models and Applications (Berlin: Springer), pp. 115-134.
- [6] B. H. Krogh (1987). Controlled Petri Net and maximally permissive feedback logic. Proceedings of 25th annual Allerton Conference, University of Illinois, Urbana, IL, USA, pp. 317-326.
- [7] C. Ramachandani, (1974). Analysis of asynchronous concurrent systems by timed Petri net, Massachusetts Inst. Tech. Project MAC, Tech. Rep. 120.
- [8] J. O. Moody and P. J. Antsaklis (1999). PN supervisors for DES with uncontrollable and unobservable transitions. Technical Report ISIS-99-004, ISIS Group at the University of Notre Dame.
- [9] K. Yamalidou, J. Moody, M. Lemmon, and P. Antsaklis (1996). Feedback control of PNs based on place invariants; Automatica, vol.32, no. 1, pp. 15-28.
- [10] L. E. Holloway, and B. Krogh (1990). Synthesis of Feedback Control Logic for a Class of Controlled Petri Nets. IEEE transactions on automatic control; vol. 35, no5, pp. 514-523.
- [11] L. E. Holloway, and B. Krogh (1992). On Closed-Loop Liveness of Discrete-Event Systems Under Maximally Permissive Control. IEEE transactions on automatic control; vol. 37, no5, pp. 692 - 697.
- [12] M. Atli, A. Sava, Z. Achour, N. Rezg. (2010). Control synthesis for partially controlled marked graph subject to marking exclusion constraints. IFAC, 12th LSS Symposium: Large Scale Systems: Theory and Applications. Volume 9, Part 1.
- [13] M. Atli, A. Sava, Z. Achour, N. Rezg. (2010). Supervisory Control of partially observable Marked Graph based on Marking Exclusion Constraint. IFAC, MCPL, Villeneuve d'Ascq.
- [14] P. Ramadge, P. and Wonham, M., (1987a). Supervisory Control of a Class of Discrete Event Processes, SIAM J. Control Optim. Volume 25, Issue 1, pp. 206-230.
- [15] P. Ramadge, P. and Wonham, M., (1987b). Modular Feedback Logic for Discrete Event Systems, SIAM J. Control Optim. Volume 25, Issue 5, pp. 1202 - 1218.
- [16] P. Ramadge, P. and Wonham, M., (1989). The control of discrete event systems, Proceedings of the IEEE. Volume 77, Issue 1, pp. 81 - 98.
- [17] P. Merlin and D.J. Faber, (1976). Recoverability of communication protocols, IEEE trans. Commun. COM-24 (9).
- [18] T. Murata (1989). Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE, vol.77, NO.4, pp. 541 - 580.
- [19] Y. Li, and W.M. Wonham (1994). Control of vector discrete event systems II-Controller synthesis". IEEE trans. Automat. Contr., vol. 39, pp.512-531
- [20] Z. Achour, N. Rezg, and X. Xie (2004). Supervisory control of partially observable marked graphs. IEEE transaction on automatic control, vol.49, no.11.
- [21] Z. Achour, A. Sava, and N. Rezg (2008). On the Supervisory Control of Marked Graph. IEEE Singapore, ISBN: 978-1-4244-2629-4. pp.2122-2126.
- [22] M. ATLI, Z. Achour, A. SAVA, K. H. Adjallah and N. Rezg (2011). Supervisory Control of Timed-Place Marked Graph based on Marking Exclusion Constraint. IEEE, CCCA, Hammamet.

A Rule-Based Energy Management Strategy for Parallel Hybrid Vehicles with Supercapacitors

Babici Cezar, Alexandru Onea Member, IEEE

Abstract— In order to improve the lifetime and performance of a hybrid electric vehicle, one of the most challenging tasks is to improve the performance of the electrical energy storage, regarding the electrical power and energy capacity. By introducing a supercapacitor as auxiliary source to increase the system power and to take over the battery stresses, the performance of the combined energy storage unit is improved. This paper introduces a completed model of energy storage system, including a pack of batteries, a pack of supercapacitors and a control strategy. The aim is to discharge the capacitors when the required power from energy storage is above a threshold meaning that the driver needs a boost of power for a period of time. Because of their high density of power supercapacitors have more efficiency in delivering it for short time. This has a positive impact on the fuel consumption and the state of charge at the end of the drive cycle.

I. INTRODUCTION

THE introduction of the supercapacitors in power systems applications became an industrial reality several years ago. Supercapacitors or ultra capacitors were initially used by the US military to start the engines of tanks and submarines. Most applications now are in small appliances, handheld electronics and hybrid electric vehicles. General Motors has developed a pickup truck with a V8 engine that uses the supercapacitor / ultracapacitor to replace the battery. The efficiency of the engine rose by 14%. The supercapacitor supplies energy to the alternator. Toyota has developed a diesel engine using the same technology and it is claimed to use just 2.7 liters of fuel at 100 km.

To achieve great electrical capacity at small scale the dielectric, materials should be with a relative permittivity as high as possible and the distance between the plates as small as possible. The usual construction of capacitors is based on the use of tin foil, zinc, aluminum or other alloys, separated by sheets of insulating material (dielectric).

The principle of operation of a supercapacitor is based on the storage of energy by distribution of the ions coming from the electrolyte in the vicinity of the surface of the two electrodes. Indeed, when one applies a terminal voltage of

supercapacitors, one creates a zone of space charge to the two interfaces electrode-electrolyte. It is what ones call the double electric layer. The storage of energy is thus electrostatic and not faradic as in the case of the batteries, since there is no electrochemical reaction. A supercapacitor has a structure anode-cathode containing active carbon, allowing to have a surface considerably higher compared to the traditional capacitors, and thus to obtain very high values of the capacity (1 to 5000 F). This construction of the supercapacitors, the potential elements of auxiliary storage makes them an ideally complementary energy source to the batteries or the fuel cell. The use of parallel-series structures of several cells of supercapacitors makes it possible to reach a voltage and a high current output [1].

For simulating the hybrid electrical vehicles it is used ADVISOR (Advanced Vehicle Simulator), interactively with Matlab and Simulink [2]. It will be tested a hybrid parallel topology with classical batteries and with the new model of energy storage developed, including the batteries and the supercapacitors. The results regarding fuel consumption and state of charge at the end of the drive cycle will be discussed at the end of this paper.

II. BATTERIES MODEL USED IN ADVISOR

Batteries are a basic component of hybrid electrical vehicles. They are used with the electric motor(s) as a device to optimize the efficiency of the ICE (internal combustion engine), as well as to recover the kinetic energy during braking or coasting of the vehicle. In hybrid electrical vehicles (HEVs), the essential parameters regarding the energy storage system (ESS) are: the available power, the capacitance, their life, and charging and discharging time until the state of charge (SOC) reaches a specific threshold. Terminal voltage of the ESS is different when charging, and when discharging:

$$U_{ch} = E - E_p - I_{required} \cdot R . \quad (1)$$

$$U_{disch} = E + E_p + I_r \cdot R . \quad (2)$$

where U_{ch} , and U_{disch} are the charging and discharging voltages, E is the open circuit voltage, E_p is a contrary electromotive force given by the polarization phenomena, I_r is the delivered current by the batteries during the discharge, and R is the internal resistance [3]. The ESS chosen for the experiments and used in ADVISOR for parallel architecture, is composed from 25 modules of lead

Cezar C. Babici is with the Automatic Control and Applied Informatics Department, "Gheorghe Asachi" Technical University of Iasi Faculty of Automatic Control and Computer Engineering, Str. Prof. dr. doc. D. Mangeron, nr. 27, 700050, Iasi, Romania (+40766434951 e-mail: cbabici@ac.tuiasi.ro.)

Al. Onea is with the Automatic Control and Applied Informatics Department, "Gheorghe Asachi" Technical University of Iasi Faculty of Automatic Control and Computer Engineering, Str. Prof. dr. doc. D. Mangeron, nr. 27, 700050, Iasi, Romania (e-mail: aonea@ac.tuiasi.ro).

batteries with a rated voltage of 308 volts. The model used in simulations (Fig.1) is considering the power requested, and the temperature of the system. A cooling system is inserted, and the cooling efficiency parameter is taken in account in calculating the available power.

Requested power from the driver after the transformations through vehicle bloc, wheel and axle bloc, final drive bloc,

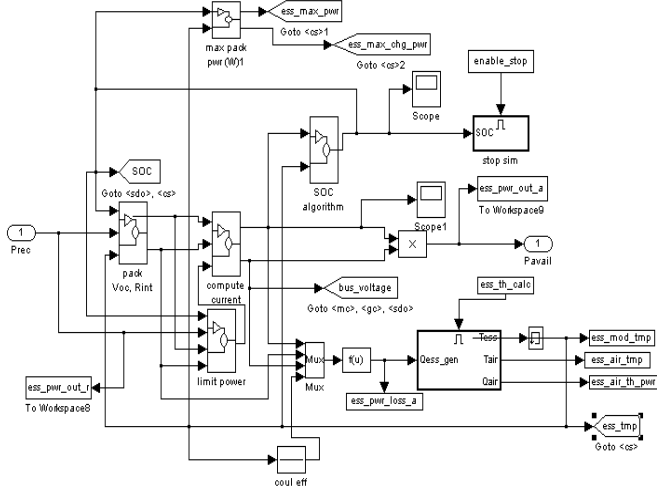


Fig. 1. ESS model used in simulations.

gear box bloc, and torque coupler bloc, is divided into power requested from (ICE), and power requested from the electrical motor, which give the required power from the ESS. Division of this power is made by the control strategy through the torque coupler, so the ICE should operate as much as possible in the most efficient way. If requested power from the driver is higher than the power available of the ICE in the efficient operating point, then the rest of it is covered as much as possible from the ESS through the electrical motor (EM), if the SOC is higher than a specific threshold. If not, then additional torque is required from ICE to charge the batteries.

$$P_{req} = T_{req} \cdot S_{rec} \quad (3)$$

where P_{req} is the requested power, T_{req} is the required torque, and S_{rec} is the required speed .

$$T_{reqM} = \frac{T_{Icreq} - T_{ICEa}}{tc \quad mc \quad to \quad fc \quad ratio} \quad (4)$$

$$S_{Mrec} = \min(S_{ICEa}, S_{Trec}) \cdot tc_mc_to_fc_ratio \quad (5)$$

where T_{reqM} is the required torque from motor, T_{Tcreq} is the required torque from the torque coupler, T_{ICEa} is the available torque from the ICE, and $tc_mc_to_fc_ratio$ is the constant ratio used in the ADVISOR model of the hybrid parallel vehicle, and represents a constant gear ratio (in this case 1.73). From torque and speed required from electrical motor, the electrical power required from ESS is calculated.

If requested power P_{req} is ≥ 0 , then the pack resistance of the batteries R_{int} is:

$$R_{\text{int}} = \text{ess_}r_disch \cdot \text{resist_scale_fun} \quad (6)$$

Otherwise if requested power is <0 then R_{int} is:

$$R_{\text{int}} = \text{ess_r_chg} \cdot \text{resist_scale_fun} \quad (7)$$

where ess_r_disch , and ess_r_chg are the charging and discharging resistance of ESS, and $resist_scale_fun$ are gain factors also defined in ADVISOR model of the batteries, ess_module_num is the ESS module number, a parameter chosen from the interface, in this case equal with 25, and ess_cap_scale , is the ESS capacitance scale equal with 1.

In the Advisor model of the batteries bank the current across the batteries is I_{bty} if $I_{bty} \geq I_{abs}$, and I_{abs} otherwise:

$$I_{bty} = \frac{\left(V_{oc} \cdot \left(V_{oc}^2 - 4 R_{int} \cdot P_{bty} \right)^5 \right)}{2 R_{int}} \quad (8)$$

$$I_{abs} = \frac{V_{oc} - V_{max}}{R_{int}} \quad (9)$$

where V_{oc} is the pack open circuit voltage, and P_{bty} is the limited output power.

Voltage of the pack batteries is given by:

$$U = V_{oc} - I_{bty} \cdot R_{int} \quad (10)$$

and the available power of the ESS is:

$$P_a = U \cdot I_{bty} \quad (11)$$

State of charge of the batteries pack, is an important parameter, permanently monitored and it is calculated as below:

$$SOC = \frac{\max_Ah_capacity - Ah_used}{\max_Ah_capacity} \quad (12)$$

where $\max_Ah_capacity$ is the maximum capacity of the ESS, in hour amps, and Ah_used is the used capacity.

The ICE can be stopped if the vehicle is at a stop, or if vehicle speed is lower than a preset threshold and the electric motor is used to drive the vehicle along. The ICE operation is optimized by adjusting the speed and torque of the engine.

III. SUPERCAPACITOR MODEL

Due to the frequency of start-stop operation mode in

hybrid-electric vehicles, charge and discharge profile of the energy storage system varies greatly. The average power

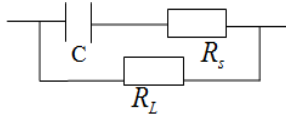


Fig. 2. Capacitor model with power transfer.

required from the energy storage system is much lower than the peak power for a short period of time needed for acceleration or climbing a slope. The ratio of peak power and average power can be up to 10:1. In fact, the energy involved in a transitory mode of acceleration and deceleration is approximately two thirds of the total energy use in an urban drive cycle of the vehicle. In designing a hybrid electric vehicle peak power delivering capacity is more important than energy capacity and in generally grows its size dimensions. Design of the battery pack should provide a compromise between specific energy and their life. Source of energy, mainly batteries and fuel cells, has high specific energy, while power supply has high specific power. Power supply can be recharged in recovery braking mode [4]. A power source that can be used in electric and hybrid-electric vehicles is a capacitor bank. These are characterized by a higher specific power compared to electrochemical batteries and a lower specific energy. Its specific energy is in the range of a few watt-hours per kilogram. But its special power can reach up to 3kW/kg, more than any type of battery.

The advantages of a supercapacitor are many. Some of them are: virtually unlimited life cycle: cycles millions of time -10 to 12 year life, low impedance, charges in seconds, no danger of overcharge, very high rates of charge and discharge, high cycle efficiency (95% or more), light weight (total mass of the vehicle has a drastic influence in fuel consumption).

As super capacitor disadvantages it can be mention that linear discharge voltage prevents use of the full energy spectrum and low energy density. Typically it holds one-fifth to one-tenth the energy of an electrochemical battery Cells have low voltages, and serial connections are needed to obtain higher voltages. Voltage balancing is required if more than three capacitors are connected in series. High self-discharge rate is considerably higher than that of an electrochemical battery. Supercapacitor performance can be expressed by the voltage of the terminal during charging and discharging with different values of the current [5]. A supercapacitor is characterized by three parameters: the capacitance C , series resistance R_s and the dielectric leakage resistance R_L .

Theoretical voltage of the capacitor can be expressed as:

$$u_c = \frac{1}{C} \int i(t) \cdot dt \quad (13)$$

The voltage across the capacitor is:

$$u(t) = u_c(t) - r_s \cdot i(t) \quad (14)$$

Another important parameter is charging or discharging time constant:

$$\tau_0 = r_s \cdot C \quad (15)$$

Maximum energy of the capacitor is given by:

$$E_{0,max} = \frac{1}{2} C \cdot U_0^2 \quad (16)$$

where U_0 is the initial charge voltage.

Capacitor can be modeled as it follows:

$$P_l = i^2 \cdot R_l = \left(\frac{u_c}{r_s + R_l} \right)^2 \cdot R_l \quad (17)$$

$$P_{l,max} = \left(\frac{u_c}{4 \cdot r_s} \right)^2 \quad (18)$$

Any power equation of the capacitor is defined in relation with the terminal load in discharging. For the model in figure 2, R_s is the equivalent series resistance, and R_l is a pure resistive load

$$Q = C \cdot v_c(t) \quad (19)$$

$$I_{sc}(t) = \frac{dq(t)}{dt} = C \cdot \frac{dv_c(t)}{dt} \quad (20)$$

$$\frac{I_{sc}(t)}{C} = \frac{dv_c(t)}{dt} \Rightarrow \frac{I_{sc}(s)}{C} = s \cdot V_c(s) \quad (21)$$

$$I_{sc}(s) = s \cdot C \cdot V_c(s) \quad (22)$$

The model of the capacitor used in the experiment is shown in Fig. 3.

The switches are used to simulate the transistors, which

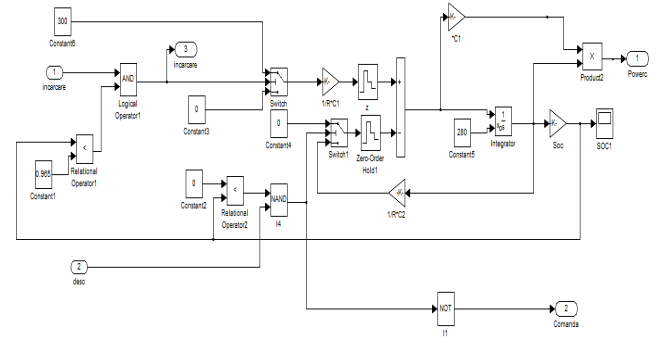


Fig. 3. Capacitor model used in simulations

allow in every moment one of 3 possibilities: charging, discharging, or standby, depending on the power request on the bus. Because of the oscillatory power request we will have modulated charging, or discharging. Otherwise if the requested power was continuous, charging/discharging was continuous, and the currents along the system were too high.

Charging equation is given by:

$$\frac{dV_c(t)}{dt} = -\frac{1}{R \cdot C} V_c(t) + \frac{1}{R \cdot C} u(t) \quad (23)$$

where $u(t)$ is the unit step size (source voltage) [6]. Using Laplace Transform and U as source voltage we obtained:

$$V_c(s) = \frac{1}{s} \left(\frac{1}{R \cdot C} \cdot U(s) - \frac{1}{R \cdot C} V_c(s) \right) \quad (24)$$

The completed model proposed includes the ESS, control strategy and capacitors also as in Fig.4.

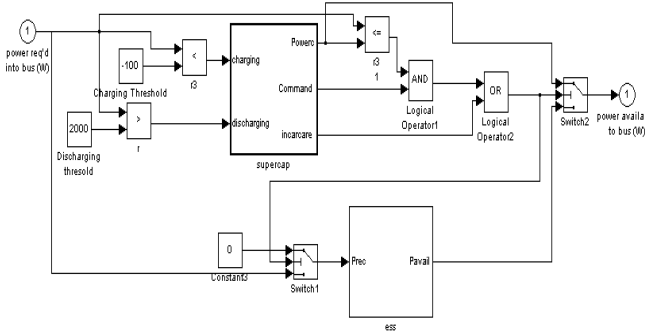


Fig. 4. ESS completed model.

When charging the auxiliary source, the input into supercapacitor model is given by the negative power required from energy storage system. On the contrary when discharging the same source, the input is 0 and the energy is given from the initial condition plus the amount cumulated during charging. So the SOC is increasing, decreasing or remaining the same during a drive cycle, depending if the requested power is above, or below some thresholds imposed in the control strategy.

IV. CONTROL STRATEGY FOR COMPLEX ENERGY STORAGE

Classic strategy used in ADVISOR is based on power required by the driver. This power, theoretical, should be equal with the power available of the system, composed from the power delivered by the ICE (internal combustion engine) and the power given by the ESS through electrical motor. When the requested power from ESS is greater than 0, means that the sources of energy are discharging, and when the opposite it happens, the sources are charging. Of course when one source is charging or discharging, the other one is the short time to do that, and in discharging when their available power is above a threshold. in standby. Capacitors have priority in charging because of Charging and discharging modes will never work simultaneously and the charge or discharge slope will be as the starting point for that mode. For example if the maximum voltage is 300 volts and the charging mode begins at the 200 volts, the charge rate will be smaller than the rate if the starting point is 50 volts.

Additional torque for charging batteries or additional electrical loads is given also by the ICE with extra cost in fuel consumption. We took for analysis in ADVISOR, an

urban cycle called CYC_UDDS with a lot of stops and starts and with few moments exceeding speed above 50km/h. This is a typical cycle with a lot of fuel consumption. We have chosen to test a parallel architecture with 41kw ICE, 25 modules of lead batteries (rated voltage: Vnom=308 volts), and a 75Kw electric motor. As control strategy in ADVISOR the default one used for parallel topology, called PTC_PAR is offering the best results. After a UDDS cycle, meaning 12km of urban traffic in 1369 sec, (average speed 31,5km/h), and initial SOC of 0,7, the obtained required

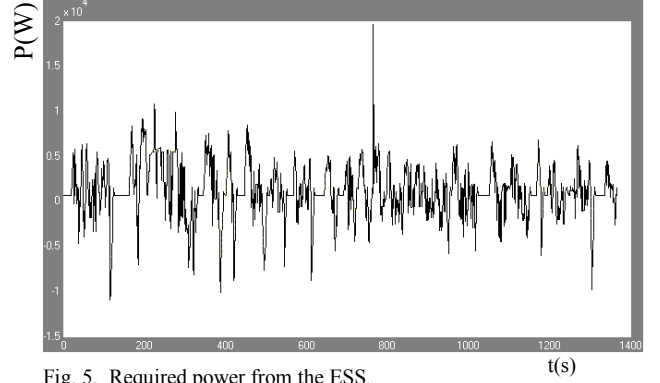


Fig. 5. Required power from the ESS.

power from the ESS was as the one shown in Fig.5.

Looking at this data we can observe that are a lot of moments when it is needed great power for short periods of time. So we thought that super capacitors will do a great job in this kind of moments. Practically, most of the time, power varies between +8000Wats and -9000Wats. With well chosen capacitors, the power above discharge threshold can be assured by the auxiliary source for a short period of time , and power available to bus will be greater than the power given by the batteries at the same power request.

In Fig. 6 it is shown the power available at bus, with the capacitors on, and it can be seen a maximum power much

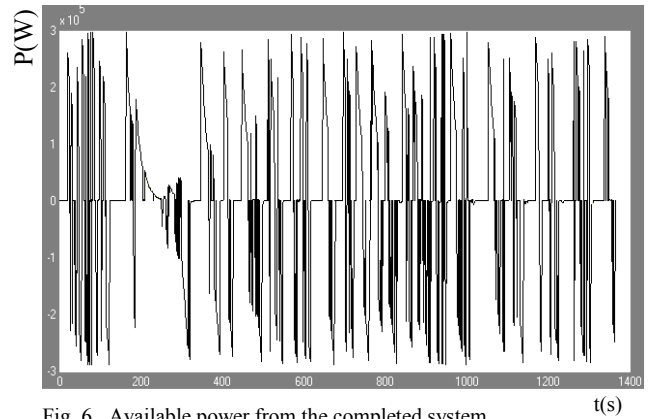


Fig. 6. Available power from the completed system.

more higher than the maximum given by the batteries.

When the capacitors are on, the classic batteries are no longer used and their SOC is no longer decreasing. Of course when charging the capacitors, SOC of the batteries is no more increasing, but this is also for short periods of time. When the required power is less than -100Wats (charging

threshold) and SOC of capacitors is less than 96.5% (charge conditions), then the alternative source is charging, otherwise the batteries are.

To do this, we used a switch commanded by the charge conditions, mentioned above, that gives at input 300 volts when charging and 0 otherwise. Discharging mode is selected when the required power is greater than discharging threshold and less than available power of the capacitors. The initial voltage of the capacitors is 280Volts.

A guide scheme for the control strategy is shown in Fig.7 where P_{req} is the power required from ESS; Chg B means charge batteries; SocS is the state of charge of the capacitors; $P_a = P_aB$ means that power available from ESS, is the power available from batteries; Dis B is discharge batteries; Chg S is charging supercapacitors; $P_a = P_aS$ means that power available is given by the supercapacitors; and $P_{req} > Trd$ means that power required is greater than the threshold for discharging the capacitors; Dis S is discharging supercapacitors.

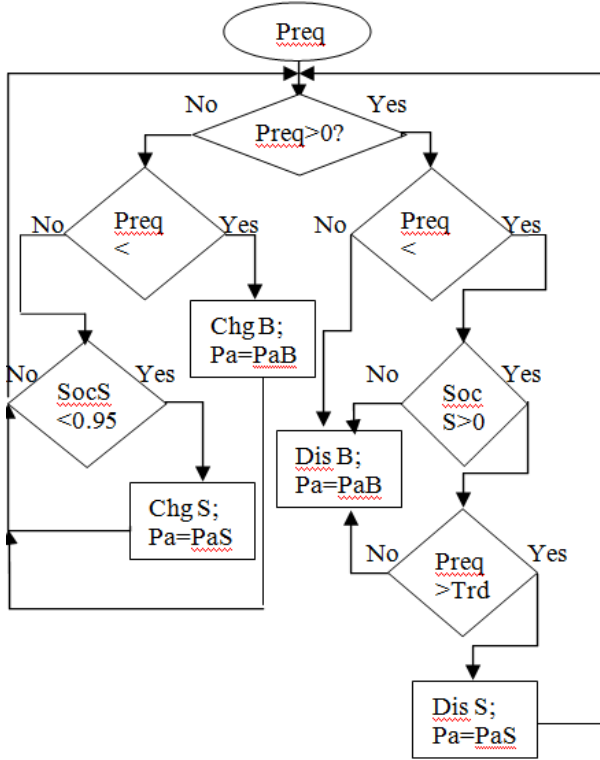


Fig. 7. Control strategy for ESS

When the required power is between charge threshold (T_{chg}) and discharge threshold (T_{dis}), the capacitors are not used thanks to the switch that select as available power from the ESS, the available power of the batteries. It is commanded by the using of the alternative source. When the alternative source is running, its available power is greater than the requested power from ESS. Using a scope we can observe the variation of the power signal is between $-3 \cdot 10^5$, and $+3 \cdot 10^5$.

State of charge of the capacitors, after finishing an urban drive cycle is between 80% and 90%, because of their short

periods of time of functioning, and because of their priority in charging mode.

This phenomenon is also decreasing fuel consumption.

V. RESULTS

For testing the initial model with parallel topology, without the auxiliary source and the completed model, it was used the UDDS drive cycle, from ADVISOR, considered most relevant in matters of fuel consumption, for urban traffic because of its distance (12Km), and because of a lot of stops and starts as in Fig.8.

Using the classic parallel architecture proposed by simulator with parameters as in table 2, it was obtained a

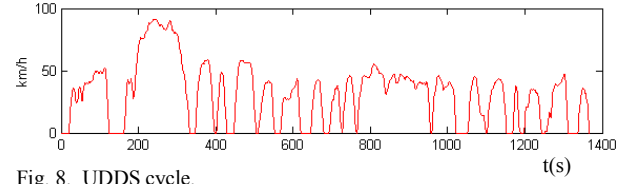


Fig. 8. UDDS cycle.

fuel consumption of 6,9l/100Km, and a final SOC of

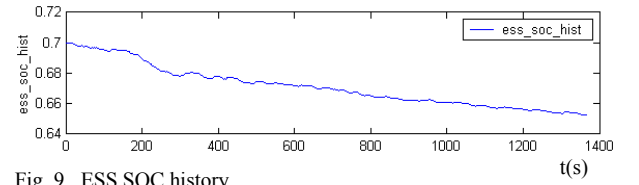


Fig. 9. ESS SOC history.

65.27%.as in Fig.9.

In terms of emissions we can observe $H_c=0,374$, $Co=1,425$, and $No_x=0,303$.

For the second case, when we use the completed model of ESS in the same initial condition, and starting SOC of the capacitors of 93,33%, we obtained a fuel consumption of 6,7l/100km, and a final SOC of 68,49% like in Fig.10.

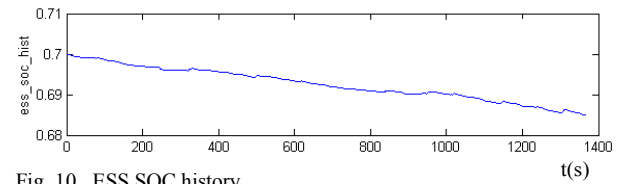


Fig. 10. ESS SOC history

The emissions were $H_c=0,371$, $Co=1,395$, and $No_x=0,293$.

The pack of capacitors used, have an internal resistance of 0,3Ohm, and a Capacitance of 100Farads. Threshold for discharging (T_{dis}) is 2000Wats, and threshold for charging (T_{chg}) is -100Wats. After completing the UDDS drive cycle, the final SOC for them it is 85.84%, history is shown in Fig.11:

In both simulations it was used the strategy called PTC_PAR, (default in ADVISOR for parallel topology) where the additional torque required is calculated as follows:

$$cs_ch \arg e_trq = 0,25 \cdot \min(fc_max_trq). \quad (25)$$

where fc_max_trq is the fuel converter maximum torque, equal with: [61.0000 67.6000 73.7000 78.5000 80.9000 77.3000 76.2000 73.3000 68.7000].

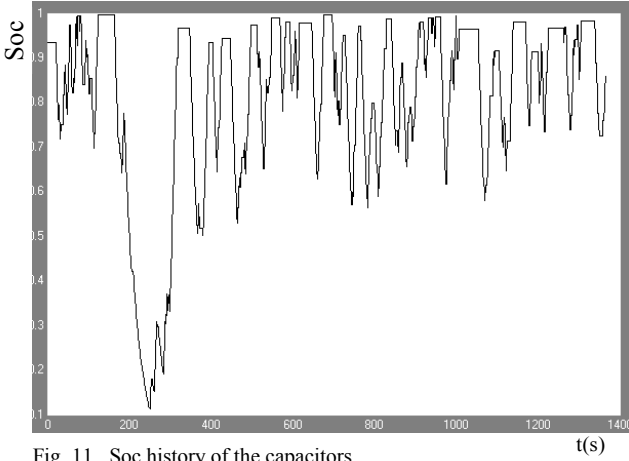


Fig. 11. Soc history of the capacitors.

It was considered another case with the same initial conditions as in case two but with additional torque required calculated in other way:

$$cs_charge_trq = \frac{25}{cs_high_soc - cs_low_soc} \quad (26)$$

It was obtained a fuel consumption of 6,4l/100km and a remaining SOC of 68.08% (a difference of 2.84% in SOC and 0,5l/100Km in fuel consumption). Lower additional torque is reducing fuel consumption.

For the next case it was used for simulation another type of drive cycle, named CYC_CLEVELAND (15 km), which it is considered conclusive for highway traffic. Speed of the vehicle is between 100 and 150Km/h most of the time, like in Fig.12.

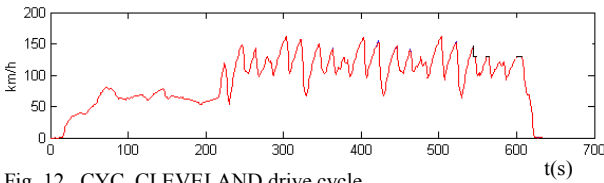


Fig. 12. CYC_CLEVELAND drive cycle.

Using the conventional architecture and the same parameters the obtained results are as follows:

- fuel consumption 7,8l/100Km;
- state of charge : 29,17% with the warning that the vehicle could not finish the drive cycle, in normal conditions, because of the SOC that decreases under the low threshold (cs_low_SOC), before the end of the cycle.

Including the capacitors model in the ESS, it was obtained a significant difference in the fuel consumption, but most important in matters of SOC, as follows:

- fuel consumption: 7.3/100Km (0,5l/100Km);
- state of charge: 49.38% (20% more energy in ESS) and no warnings, meaning that the vehicle finished the drive cycle in good conditions.

In the control strategy it was modified the threshold for discharging the supercapacitors from 2000Wats for urban traffic, to 54000Wats for highway traffic. At the end of the drive cycle the SOC of the capacitors is 0.98%. It was used the notations of the variables from ADVISOR.

Using them in the control strategy to calculate the additional torque required, best results were obtained as it follows:

- fuel consumption 6,9l/100 km (a difference of 0,9l/100Km)
- final SOC 64.78% (a difference of 35%)

Final SOC of the capacitors at the end of the drive cycle is very high, (Fig.13) meaning that the auxiliary source is ready to be used in another drive cycle without charging

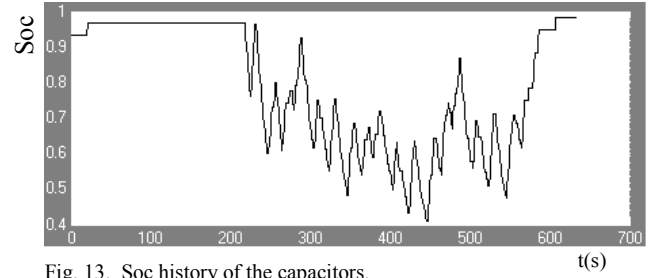


Fig. 13. Soc history of the capacitors.

first.

VI. CONCLUSIONS

The parallel architecture only with Pb batteries has good performances, but in some cases like those presented above, the fuel consumption, and especially the final state of charge could be improved. Supercapacitors have the advantage of high power and disadvantage because that power is given for short periods of time. Yet they could be very useful in replacing the batteries only when needed great amount of power for those short periods of time. Discharging fast means that the capacitors are also charged very fast in comparison with conventional Pb batteries. Because of that, in the control strategy they are charged first, and then the conventional energy storage.

The best configuration is parallel, with the completed model including the capacitors and the control strategy.

REFERENCES

- [1] J.N. Marie-Francoise, H. Gualos, R. Outbib, "42V Power Net with supercapacitor and battery for automotive applications", Elsevier B.V, 2005.
- [2] Aaron Brooker, Kristina Haraldsson, Terry Hendricks, Valerie Johnson, Kenneth Kelly, Bill Kramer, Tony Markel, Michael O'Keefe, Sam Sprik, Keith Wipke, Matthew Zolot, "ADVISOR Documentation", NREL, 2002.
- [3] Gheorghe Livint, Vasile Horga, and Marcel Ratoiu "Vehicule Electrice Hibride", (Ed. Venus, 2006).
- [4] Mehrdad Ehsany, Yimin Gao, Sebastien E. Gay, Ali Emadi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles*. Boca Raton, CRC Press, 2004..
- [5] Allen E. Fuhs, *Hybrid Vehicles and the future of personal transportation*. Boca Raton, CRC Press, 2009.
- [6] Iqbal Husain, *Electric and Hybrid Vehicles - Design Fundamentals*, Boca Raton, CRC Press, 2003.

Experimental Evaluation of Service Negotiation Protocols in Agent Systems

Costin Bădică, Mihnea Scafeș

University of Craiova, Software Engineering Department

Bvd.Decebal 107, Craiova, 200440, Romania

{badica_costin, scafes_mihnea}@software.ucv.ro

Abstract—In this paper we provide experimental results concerning the impact of the negotiation protocol onto the quality of the negotiation outcome as well as onto the communication complexity of interactions incurred during one-to-many automated negotiations. We evaluate experimentally three negotiation protocols (Direct Task Assignment, Contract Net and Iterated Contract Net) with respect to two performance measures: negotiation outcome (i.e. utility) and communication complexity (i.e. number of exchanged messages) for different profiles of the contractor agents. We find that the Direct Task Assignment protocol delivers the worst average outcome, but at the same time it consumes the lowest number of messages. The Contract Net and Iterated Contract Net deliver much higher utility on average, while the Iterated Contract Net obtains the highest outcome for some configurations at the cost of the highest number of exchanged messages.

I. INTRODUCTION

The increasing complexity of real-world problems demands special support for distributed collaborative problem solving. Multi-agent systems (MAS) are a special class of distributed systems that combine interaction and coordination with distribution of computation to improve performance of problem solving processes. MAS were successfully applied for solving problems that require distributed reasoning, decentralization and collaboration [18]. An example is a collaboration system for helping human experts and population to deal with disasters (see the FP7 DIADEM project¹ that targets crisis management in the context of chemical incidents in industrial and urban areas).

A workflow is a structured set of coordinated tasks and information exchanges defined to carry out a well-defined business process in (possibly multiple networked) organizations [17]. Collaborative problem solving in heterogeneous, unpredictable, and dynamic environments can be achieved by providing an increased flexibility in workflow formation, that goes far beyond the classic approach of static and centralized workflow definitions. Examples based on empirical analysis of real experiences can be given in the area of organizational adaptation to disasters [8]. Such infrastructure support can be provided by service-oriented MAS, like for example the Dynamic Process Integration Framework (DPIF) [12].

Costin Bădică was supported by DIADEM project. Diadem project is funded by European Union under Information and Communication Technologies (ICT) theme of the 7th Framework Programme for R&D, ref. no: 224318. Mihnea Scafeș was supported by IOSUD-AMPOSDRU contract 109/25.09.2008.

¹DIADEM – Distributed information acquisition and decision making for environmental management: <http://www.ist-diadem.eu/>.

In such frameworks, communication links between local processes in agents are facilitated using service discovery: whenever an agent supplying some service (we will call this agent the *manager*) in a workflow requires data provided by some other service (we will call this agent the *contractor*), a communication link must be established between them. A similar problem occurs in optimal resource allocation for workflow scheduling [9], [6], as well as in dynamic role binding in MAS organizations [10].

The self-configuration provided by service discovery and matching can be improved by enhancing it with a finer level of control based on one-to-many service negotiation ([5], [11], [13], [19]). In our work we proposed a novel negotiation model that follows the conceptual framework of one-to-many service negotiation [2].

In this paper we provide experimental results concerning the impact of the negotiation protocol onto the quality of the negotiation outcome as well as onto the communication complexity of interactions incurred during negotiations. This paper is a continuation of [20] by considering different experimental settings, and also an extended version of a previously published paper [25]. We evaluate experimentally three negotiation protocols (Direct Task Assignment, Contract Net and Iterated Contract Net) with respect to two performance measures: negotiation outcome (i.e. utility) and communication complexity (i.e. number of messages transferred), by assigning different busy profiles to the contractors. We find that the Direct Task Assignment delivers the worst average outcome, but at the same time it uses the lowest number of messages. The Contract Net and Iterated Contract Net deliver much higher utility on average, but the Iterated Contract Net obtains the highest outcome for some configurations and also uses the highest number of messages.

The paper is organized as follows. We start in Section II by introducing the negotiation protocols that are currently supported by our negotiation framework. In Section III we provide an experimental analysis of the effect of negotiation protocol selection onto the negotiation outcome as well as onto the communication complexity of negotiation interactions. In Section IV we cover related works, while in Section V we present our conclusions and point to future works.

II. NEGOTIATION PROTOCOLS

Negotiation is a process that describes the interaction between one or more participants that must agree on a subject by exchanging proposals about this subject [7]. Negotiation

about a service that one or more participants agree to provide to other participants is called *service negotiation*. We have developed a conceptual framework for service negotiation that addresses protocols, subjects and decision components. Our framework supports generic one-to-many negotiations and it defines two roles: manager and contractor [16], [11]. The manager is the agent that requests a service and thus initiates the negotiation. The contractor is the agent that is able to provide the service requested by the manager. For a more complete description of the conceptual framework, please see [2]. A brief description of the design and implementation of this framework is given in [15].

Currently we have configured our framework with three negotiation protocols that we have found useful in disaster and environment management problems – the area where our framework is currently being applied [20]. These protocols are:

- 1) *Direct Task Assignment*. Using this protocol there is actually no negotiation happening between the manager and the contractors. The manager simply picks up one or more contractors and assigns them the task². Although simple, *Direct Task Assignment* – *DTA* negotiation protocol is actually very useful to model subordination relationships, that are common for task assignments to team members that operate in a disaster environment. For example, in a disaster management information system, an *Incident Commander* can assign specialized tasks to subordinated team members that were dispatched and are operating in the area affected by the disaster.
- 2) *Contract Net*. Proposed by [16], *Contract Net* – *CNET*³ is probably one of the most influential negotiation protocols utilized in distributed and collaborative problem solving, with many extensions currently available (see for example [19]). *CNET* is essentially a one phase negotiation protocol: the manager announces a task to the contractors in the announcement stage, each contractor proposes or refuses to submit a bid in the bidding stage, and finally the manager decides to award the task to at least one of the contractors in the awarding phase. It can be easily noticed that the *DTA* protocol is in fact a simplified *CNET* without the announcement and the bidding steps.
- 3) *Iterated Contract Net*. *CNET* can be extended to multiple negotiation phases thus obtaining *Iterated Contract Net* – *ICNET*⁴. This protocol is very useful in situations when a single phase is not sufficient for the manager to select an appropriate contractor for awarding the task. This may happen for example if the requirements set for the task announced by the manager are too

restrictive and thus neither of the contractors is able to meet them in a satisfactory way. In this case the task requirements must be relaxed by the manager before reaching a suitable agreement. Another situation is when contractors are operating in a constrained environment that does not allow them to bid. For example, considering a disaster management scenario, one of the stakeholders (for example a chemical expert) might be caught in an important meeting that forbids him or her to answer to incoming phone calls. However, repeated calls, probably originating from repeated task announcements in a negotiation caused by the occurrence of a very important event, can enable him or her to pick up the phone and answer, i.e. to bid.

A negotiation type (corresponding to a negotiation protocol supported by the framework) was defined by configuring a set of generic negotiation steps: (i) negotiation subject identification and negotiation announcement (initiation of negotiation), (ii) bidding, i.e. making proposals and counter-proposals, (iii) deciding whether an agreement or a conflict was reached, and (iv) termination.

Negotiation subject comprises the service description and a subset of the service parameters that are important decision factors during negotiation (i.e. their current values are taken into consideration when selecting the appropriate service providers). During negotiation, these parameters are considered *negotiation issues*. For more information about the negotiation issues, see [2].

Service negotiations in disaster management are *cooperative*. Cooperation stems from the fact that the overall goal of participants is the improvement of the response for situation assessment. Negotiations for a certain service provision are carried out only with agents that can provide the required service (i.e. that possess the domain knowledge or physical capabilities that are needed to provide the service). Provider agents will usually accept to offer their services if they are currently able to do so (i.e. if they possess all the necessary resources). During a negotiation: (i) the manager is the leading decision factor seeking to improve the assignment of the negotiated task to the contractor(s); (ii) the contractor(s) make their best proposals for the manager, taking into account their current duties and availability, thus preserving their autonomy according to the principles of professional bureaucracy [1].

Negotiation participants playing either the manager or contractor roles use utility functions to quantify their preferences over proposals. In our framework the manager uses a weighted additive utility function over the negotiation issues to evaluate proposals and to select the service provider.

III. EXPERIMENTS

In this section we discuss the results of our initial experiments with service negotiations utilizing the different negotiation protocols currently supported by our framework: *DTA*, *CNET*, and *ICNET*. We first propose a model for the experimental analysis of one-to-many negotiations and then we

²*DTA* has similarities with *FIPA Request* interaction protocol, see <http://www.fipa.org/specs/fipa00026/>, excepting that the contractor does not have the option to refuse the task.

³*CNET* is standardized by Foundation for Intelligent Physical Agents, see <http://www.fipa.org/specs/fipa00029/>.

⁴*ICNET* is standardized by Foundation for Intelligent Physical Agents, see <http://www.fipa.org/specs/fipa00030/>.

provide experimental results that cover negotiation outcome and communication complexity of negotiation interactions.

A. Modeling Assumptions

Let us assume that a manager agent M is negotiating for contracting a service provided by a contractor agent that is member of a set of n available contractors $C_1, \dots, C_n$. Simplifying things we assume that each contractor C_i can offer a utility value u_i to the manager such that $u_i \in [u_i^{\min}, u_i^{\max}] \subseteq [0, 1]$. The interval $[u_i^{\min}, u_i^{\max}]$ is part of the profile of the contractor. For example, if the utility intervals of contractors C_i and C_j have an empty intersection and $u_i^{\min} < u_j^{\min}$ it means that C_j will always appear as more profitable than C_i for the manager. However, if the intersection of the intervals is non-empty it means that sometimes C_i can also be more profitable than C_j for the manager. Note that we assume that contractors are trustworthy. This basically means that in our current model each contractor is 100% reliable, i.e. if it promises to provide a certain utility during a negotiation, that utility will be actually delivered to the manager during the service provisioning.

Moreover, each negotiation will always involve requirements that are set by the manager and must be met by the contractor in order to be allowed to bid for a contract. These requirements depend on the negotiation issues and on the constraints on the negotiation issues that are set by the manager in the request for service. For example, considering the *Concentration Measurement* service that is provided by a *Measurement Team* agent that is able to measure gas concentrations in the field, a *Chemical Adviser* agent may require the *Measurement Team* agents to bid if and only if they are equipped with certain specialized measurement devices like for example *drager tubes*⁵. We assume that for each contractor C_i there is a probability of $c_i \in [0, 1]$ that she will be able to satisfy the requirements set by the manager's request. The probability c_i is also part of contractor's C_i profile.

Contractors are usually critical resources that do not always have the possibility to bid for a service contract. Let us take for example the *Chemical Adviser* that can be caught in an important meeting while health complaints start being reported. In such a situation the *Incident Commander* must assign the task *Map of Critical Zones* to an expert in the chemistry of gases, i.e. to the *Chemical Adviser* agent in this particular case. Then the *Incident Commander* must adopt an iterative negotiation protocol to allow the *Chemical Adviser* to find time to bid – note that chemical experts are very busy persons. Let us assume that in our experimental setting each contractor C_i has a “busy” profile $b_i = \{b_{i_1}, b_{i_2}, \dots, b_{i_m}\}$, where m is the maximum number of iterations and b_{i_j} , $1 \leq j \leq m$ is the probability of the contractor C_i of being busy during negotiation iteration j . We assume that always $b_{i_m} = 0$, so that C_i will definitely be able to propose a bid during the last negotiation iteration. The busy profiles are not taken into

account when using the *DTA*. When the manager agent is using *CNET*, only the probability of being busy during the first iteration is taken into account. Busy profiles are mostly useful for experimenting using the *ICNET* protocol, which is based on multiple iterations.

Differently from the previous experiments reported in [20], the manager does not have a profile that indicates its probability for selecting one of the available negotiation protocols. Rather, we assume that the manager is using one of the available negotiation protocols and it applies a utility function to measure the outcome of the negotiation. This assumption allows us to effectively compare the negotiation protocols, rather than to experimentally evaluate the effect of dynamically changing the negotiation protocol for a given negotiation.

Depending on the utilized protocol, the negotiation will incur a certain communication cost estimated as the number of messages exchanged between the manager and the contractor during the negotiation. Moreover the quality of the negotiation outcome will be estimated as the utility perceived by the manager for the contracted service. Here we adhere to the same assumptions as utilized in [20].

If the manager is utilizing the *DTA* negotiation protocol then she will randomly assign the task to one of the contractors. However, a contractor that does not meet the requirements will not be able to provide the service, so she will have to report failure. In this case the manager will randomly select another contractor and so on, until a suitable contractor is found (we assume this is always the case). Note however that this trial-and-error process performed by the manager affects the outcome of the negotiation by decrementing her perceived utility. More precisely, if the successful contractor C_i that could perform the task was selected in the k -th trial then the utility perceived by the manager will be $u_i \times (1 - (k-1)/n)$ rather than u_i . Moreover, the communication cost associated to this negotiation interaction consists of $2 * k$ message exchanges.

If the manager is utilizing the *CNET* negotiation protocol then she will select the contractor C_i that provides her the highest utility u_i from those contractors that met the requirements of the call for proposals and were not busy (i.e. they were able to bid). The communication cost consumed for a busy contractor consists of 2 message exchanges, while for a not busy contractor (it doesn't matter if she could meet or not the requirements, according to the *CNET* negotiation protocol [16] she either proposed or refused to bid) the communication cost consists of 3 message exchanges.

If the manager is utilizing the *ICNET* protocol we assume that she will perform as many negotiation iterations as needed to select one contractor. We simplify the negotiation scenario by assuming that contractors do not change their bids between iterations. This assumption is not as restrictive as it might look, because some contractors are busy and can bid only in a late iteration. Moreover, we also assume that a busy contractor that meets the requirements of the call for proposals will always find time to bid during a certain negotiation iteration. The communication cost for a busy

⁵They are tubes that contain chemicals that react to a specific compound to form a stain or color when passed through the gas.

contractor consists in 2 message exchanges, while for a non-busy contractor it consists of 3 message exchanges, for each negotiation iteration.

B. Experimental Results and Discussions

We created a simulation experiment assuming the model introduced in Section III-A. In the simulation we considered one manager and 100 contractors.

Each contractor C_i is characterized by her profile defined as a triple $([u_i^{\min}, u_i^{\max}], c_i, b_i)$. There are at most 3 negotiation iterations and we consider 3 busy profiles that we can assign to contractors:

- 1) available during the first and the second iteration with equal probability. In this case, $b_i = [0.5, 0, 0]$. This profile says that a contractor will be able to respond during the first or the second iteration, but no later than the second iteration. These contractors respond during the early stages of the negotiation.
- 2) mostly busy during the first iteration, but available during the second iteration for sure. In this case, $b_i = [0.9, 0, 0]$. This profile says that a contractor will be able to respond during the first iteration in very few situations, while it will surely respond during the second iteration. These contractors respond during the middle stage of the negotiation.
- 3) not available during the first iteration, but available during the second and third with equal probability. In this case, $b_i = [1, 0.5, 0]$. Contractors having this profile do not respond during the first iteration, but they will respond during the second or the third iteration, no later than the third iteration. These contractors respond during the late stages of the negotiation.

In the experiment, we assign the three busy profiles to contractors by following a certain configuration and for each configuration we run a bundle of 2000 negotiations. A certain configuration is a tuple (triple in our case, as we considered 3 busy profiles) of fractions representing the proportion of contractors with a given profile from the set of all the available contractors. We vary the percentages of contractors that have been assigned the busy profiles with a step of 5%. For example, we start from (0, 0, 100), meaning that all the contractors have been assigned the third busy profile and then we continue with (0, 5, 95), (0, 10, 90), $\dots$ (100, 0, 0). In the final configuration all the contractors were assigned the first busy profile.

The values of the utilities u_i are randomly selected for each negotiation instance assuming uniform distributions. For this experiment, we consider $[u_i^{\min}, u_i^{\max}] = [0, 1]$. The status of the contractor (as satisfying or not satisfying the requirements of the manager) is randomly selected for each negotiation instance, according to the probability c_i . We fixed the probability c_i to 0.5 for all negotiations, for all configurations and for all contractors.

We were interested to experimentally evaluate how each of the proposed negotiation protocols performs in terms of outcome and message traffic for each configuration of

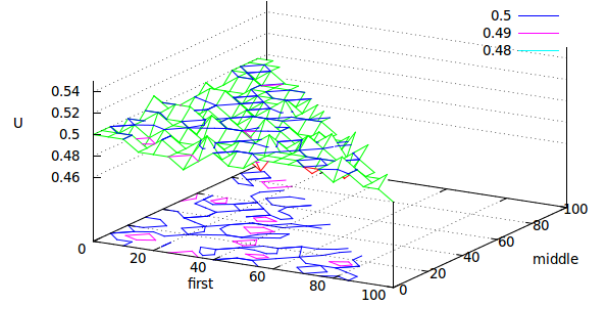


Fig. 1. Utility for DTA

contractors. So we ran the negotiations in a simulation experiment for each negotiation protocol.

Figure 1 shows the utility perceived by the manager when using the *DTA* protocol. The scale labeled “first” shows the percentage of the contractors that can propose mostly during the first iterations of the negotiation, i.e. they have been assigned the first busy profile. The scale labeled “middle” shows the percentage of contractors that have been assigned the second busy profile. The percentage of the contractors that have been assigned the third profile is not shown in that figure, but it can be obtained by taking into account the fact that the sum of all the percentages is 100.

The maximum utility for *DTA* is a little over 0.5, making this the most inefficient protocol in terms of utility. Note that the status that a contractor is “busy” cannot be taken into account by the manager when she is playing the *DTA* negotiation protocol, i.e. if she selects a certain contractor then the task will be assigned to her in any case. Nevertheless, if the assigned contractor cannot finalize the task successfully then she will report failure (possibly with some delay that is not explicitly considered in our model) and consequently the manager will retry the operation of service contracting by assigning the task to another contractor. Note that the utility is almost the same for all configurations, the average being around 0.5. However, sometimes the utility drops below 0.5 because of its decrease after each retry of the task assignment.

Figure 2 shows the message statistics for the same protocol. In terms of messages, *DTA* performs very well, being the protocol incurring the lowest number of message exchanges, thus outperforming the other protocols by far.

Figure 3 shows the utility for *CNET*, which was expected to decrease as the number of contractors that propose during the first iteration decreases. This can be explained by the fact that the manager receives less proposals and the probability to receive good proposals (i.e. of higher utility) is lower. An interesting fact is that the utility decreases abruptly as the number of contractors that propose during the first iteration decreases.

The message count increases linearly with the number of contractors that propose during the first iteration (Figure 4).

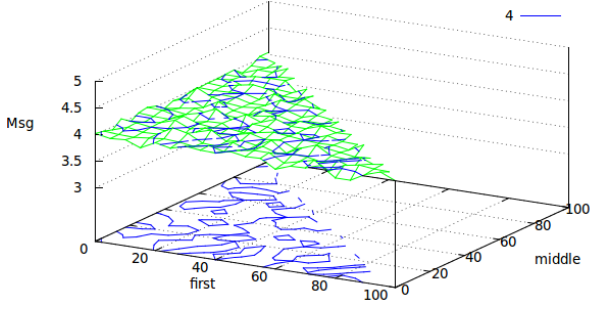


Fig. 2. Messages for DTA

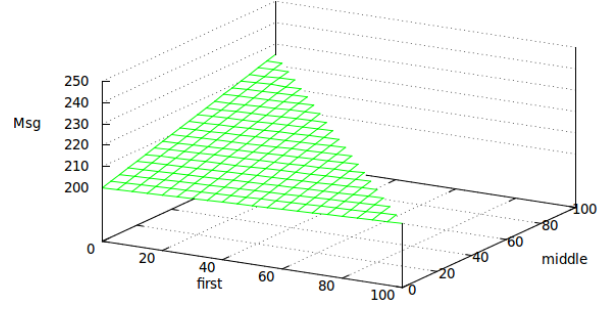


Fig. 4. Messages for CNET

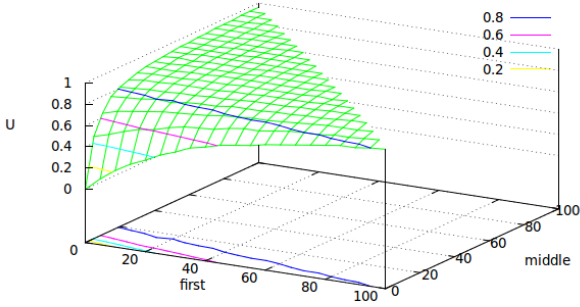


Fig. 3. Utility for CNET

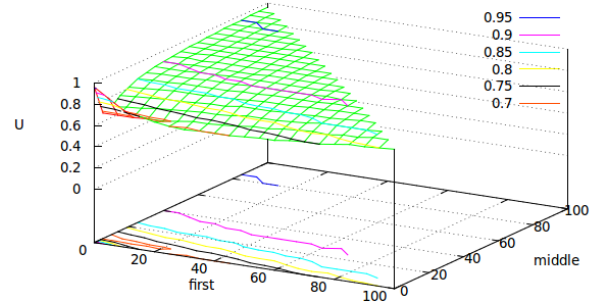


Fig. 5. Utility for ICNET

For *ICNET* (Figure 5), the utility decreases almost as for *CNET* until a point where almost all contractors propose during iterations 2 and 3. Then it grows back to the initial maximum very fast.

When most of the contractors propose during iterations 2 and 3, the message count grows abruptly to over 400 (see Figure 6). For the rest of the configurations, the message count varies almost the same as for *CNET*.

IV. RELATED WORK

Recently, a service negotiation framework has been proposed in paper [11]. That framework was applied to Web service negotiations between insurance companies and car repair companies. The negotiation is implemented as an interaction between Web services and the participants' preferences are represented using an ontology. The framework utilizes *ICNET* protocol for negotiation, with the message exchanges implemented as Web service method invocations.

They evaluate only the *ICNET* protocol, with respect to various negotiation strategies. They consider utilities for both the manager and the contractor. We evaluate three negotiation protocols in different scenarios, when the contractors can be busy or not satisfy the conditions of the manager.

There are few papers that deal with evaluation of service negotiation mechanisms, most of them discussing the *CNET*

protocol. In [24] the authors evaluate through experiments different types of contracts introduced in [23]. Our study is different because our problem deals with service negotiation between collaborative agents that do not accept payments. Agents are able to provide only certain sets of services (depending on their capabilities) and tasks cannot be exchanged between agents.

Various extensions of *CNET* have been proposed. A survey and evaluation of these according to a set of criteria have been presented in [22], but without a specific problem in mind. We focus on performance of negotiations in the context of service negotiation in agent-based disaster management systems. Moreover, our analysis considers specific features of disaster management problems like not being able to respond immediately to a call for proposal for a service.

A scalability analysis of *CNET* has been done in [21]. The authors draw the conclusion that *CNET* cannot be used without deadlines and that negotiations have to be repeated under heavy load.

V. CONCLUSIONS AND FUTURE WORK

In this paper we presented a simple experimental setting and experimental results aimed at analyzing the impact of the service negotiation protocol on the quality of the negotiation outcome as well as on the communication complexity of

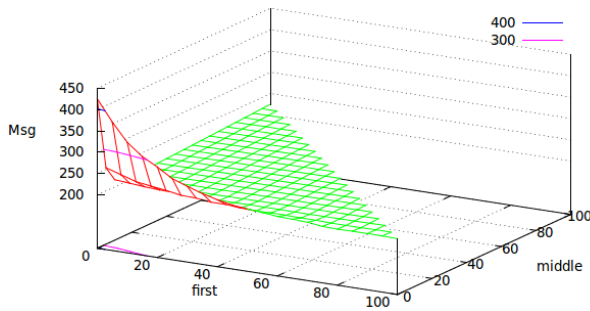


Fig. 6. Messages for ICNET

negotiation interactions depending on several profiles of contractor agents; the results can help the manager to tune his strategy for better selection of the negotiation protocol. The experiments are follow-ups of the experiments initially reported in [20], but using different negotiation settings. We evaluated experimentally the Direct Assignment Protocol, the Contract Net protocol and the Iterated Contract Net protocol with respect to the communication complexity and the negotiation outcome. We found that the *DTA* protocol delivers the smallest communication overhead to the system, but the worst outcome. However, if sometimes a smaller utility could be acceptable (in our experiments the average outcome was 0.5) then this protocol can still be useful, taking also into account the lower communication complexity of the system. Both *CNET* and *ICNET* deliver higher utility, but among these two the *ICNET* leads to a higher average utility for all the agent configurations (you can compare figures 3 and 5 and notice that while the utility for *CNET* decreases and reaches 0, the utility for *ICNET* starts to decrease but at some point it grows back to its maximum). The communication overhead brought by *ICNET* grows to a higher value and with a higher steep than the *CNET*, but only for a couple of configurations.

As future work we plan to expand the experiments in at least two directions: (i) to consider more complex negotiation instances that take into account more negotiation iterations, dynamic contractor bids and dynamic manager strategies; (ii) to consider more complex workflows involving at least two interdependent negotiations such that the contracted service might also involve contracting of other required services.

REFERENCES

- [1] Argente, E., Julian, V., and Botti, V.: Multi-Agent System Development Based on Organizations. *Electronic Notes in Theoretical Computer Science* 150, 55–71 Elsevier (2006)
- [2] Bădică, C., Scafeș, M.: Conceptual Framework for Design of Service Negotiation in Disaster Management Applications. In: Bai, Q., Fukuta, N. (eds.), *Advances in Practical Multi-Agent Systems*, *Studies in Computational Intelligence* 325, 359–375 Springer Verlag (2010)
- [3] Bădică, C., Ilie, S., Kamermans, M., Pavlin, G., and Scafeș, M.: Using Negotiation for Dynamic Composition of Services in Multi-Organizational Environmental Management. *International Symposium on Environmental Software Systems ISESS'2011*, 177–188 (2011)
- [4] Cervenka, R., Trencansky, I.: *The Agent Modeling Language – AML*, *Whitestein Series in Software Agent Technologies and Autonomic Computing* (2007)
- [5] Hemaissia-Jeannin, M., El Fallah Seghrouchni, A., Labreuche, C.: A New Multilateral Multi-issue Negotiation Protocol and its Application to a Crisis Management Problem. *Multiagent and Grid Systems* 4(1), 103–123 IOS Press (2008)
- [6] Hsieh, F.-S.: Analysis of contract net in multi-agent systems. *Automatica* 42(5), 733–740 Elsevier (2006)
- [7] Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra, C., and Wooldridge, M.: *Automated Negotiation: Prospects, Methods and Challenges*. *Group Decision and Negotiation* 10(2), 199–215 Springer (2001)
- [8] Kreps G.A., Bosworth, S.L.: *Organizational Adaptation to Disaster*. In: Rodriguez, H., Quarantelli, E.L., and Dynes, R.R. (eds.), *Handbook of Disaster Research*, 297–315 Springer (2007)
- [9] Lee, K.M.: Adaptive Resource Scheduling for Workflows Considering Competence and Preference. In: Negoita, M.Gh., Howlett, R.J., and Jain, L.C. (eds.) *Knowledge-Based Intelligent Information and Engineering Systems*. *Lecture Notes in Computer Science* 3214, 723–730 Springer (2004)
- [10] Lee, J., Lee, S.-J., and Chen, H.-M.: Dynamic role binding with Agent-centric Contract Net Protocol in agent organizations. *IEEE International Conference on Systems, Man and Cybernetics, SMC 2008*, 536–643 (2008)
- [11] Paurolally, S., Tamma, V., and Wooldridge, M.: A Framework for Web service negotiation. *ACM Transactions on Autonomous and Adaptive Systems* 2(4), ACM Press (2007)
- [12] Pavlin, G., Kamermans, M., and Scafeș, M.: Dynamic Process Integration Framework: Toward Efficient Information Processing in Complex Distributed Systems. *Informatica* 34, 477–490 (2010)
- [13] Sandholm, T.: An implementation of the contract net protocol based on marginal cost calculations. *Proceedings of the 12th International Workshop on Distributed Artificial Intelligence*, 295–308 (1993)
- [14] Scafeș, M., and Bădică, C.: Framework for Performance Evaluation of Service Negotiations in Agent Systems. *Proc. of the 8th Int. Workshop on Modelling, Simulation, Verification and Validation of Enterprise Information Systems MSVVEIS 2010*, 19–29, INSTICC Press (2010)
- [15] Scafeș, M., Bădică, C., Pavlin, G., and Kamermans, M.: Design and Implementation of a Service Negotiation Framework for Collaborative Disaster Management Applications. *Proceedings of the 2nd International Conference on Intelligent Networking and Collaborative Systems INCOS'2010*, 519–524 (2010)
- [16] Smith, R.G.: The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. *IEEE Transactions on Computers* 29(12), 1104–1113 IEEE Computer Society (1980)
- [17] van der Aalst, W., van Hee, K.: *Workflow Management: Models, Methods, and Systems*. MIT Press (2002)
- [18] Wooldridge, M.: *An Introduction to MultiAgent Systems* (2nd ed.), John Wiley & Sons (2009)
- [19] Yang, J., Li, W.-L., and Hong, C.-Y.: An Improvement to CNCP in Large-Scale Multi-Agent System. *Proceedings of the 3rd International Conference on Innovative Computing Information and Control ICIC'08*, 147 (2008)
- [20] Bădică, C., Scafeș, M., and Ilie, S.: Dynamic Negotiations in Multi-Agent Systems. In: Ermolayev, V. et al. (eds.) *Proc. 7-th Int. Conf. ICTERI 2011, Kherson, Ukraine, May 4-7, 2011*, *CEUR-WS.org/Vol-716*, ISSN 1613-0073, 8–22, online ceur-WS.org/Vol-716/ICTERI-2011-CEUR-WS-Invited-1-p-8-22.pdf
- [21] Juhasz, Z., Paul, P.: Scalability analysis of the contract net protocol. In: *CCGRID '02: Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid*, Washington, DC, USA, IEEE Computer Society (2002) 346
- [22] Bozdog, E.: A survey of extensions to the contract net protocol. Technical report, CiteSeerX - Scientific Literature Digital Library and Search Engine [<http://citeseerx.ist.psu.edu/oai2/>] (United States) (2008)
- [23] Sandholm, T.: Contract types for satisficing task allocation: I theoretical results. In: *AAAI Spring Symposium: Satisficing Models*. (1998)
- [24] Andersson, M.R., Sandholm, T.: Contract types for satisficing task allocation: II experimental results. In: *AAAI Spring Symposium: Satisficing Models*. (1998)
- [25] Bădică, C., and Scafeș, M.: Experiments with Protocols for Service Negotiation. *Proceedings of the Workshop on Applications of Software Agents*, 25–31, Novi Sad, Serbia, July 3–5, 2011 (2011)

Floating-point to fixed-point code conversion with variable trade-off between computational complexity and accuracy loss

Alexandru Bârleanu, Vadim Băitoiu and Andrei Stan, *Member, IEEE*

Abstract— This paper describes a method of converting floating-point expressions into equivalent fixed-point code in DSP software. Replacing floating-point expressions with specialized integer operations can greatly improve the performance of embedded applications. This is a new method that is developed for Direct-Form I filters with constant coefficients and input variables whose low/high bounds are known. Two conflicting objectives are considered simultaneously: computational complexity and accuracy loss. The algorithm presented here can construct multiple fixed-point solutions for the same floating-point code: from “high-complexity-high-accuracy” to “low-complexity-low-accuracy”. A so-called cost function conducts the data flow transformation decisions. By changing the cost function coefficients, different fixed-point forms can be obtained. The data flow transformation takes very little time: less than 100 milliseconds for a 32-tap FIR filter. The generated fixed-point code is tested on 8-bit (AVR ATmega), 16-bit (MSP430), and 32-bit (ARM Cortex-M3) microcontrollers. It provides, in all cases, execution speeds better than if using floating-point code.

I. INTRODUCTION

Floating-point code is somehow inappropriate for embedded applications. The computing capabilities of microcontrollers are reduced in general and, in most cases, no hardware support for floating-point operations is provided. To overcome this problem, the mathematical function contained in the floating-point code must be expressed with fixed-point code. Doing this manually – that is, rewriting by hand a floating-point function into a sequence of integer operations can be a difficult task.

II. RELATED WORK

There has been a significant effort to develop frameworks to automate the conversion of floating-point code to integer code [1]–[4]. Two distinct approaches can be identified: statistical (simulation-based) and analytical. The difference between them is in the way the dynamic intervals of variables are computed. A statistical method performs a series of simulations and may require an important amount of time. An analytical method is obligatorily based on a concrete data model (for example, propagation rules) and can give precise information in very short time.

In [1] is described one of the first floating-point to fixed-point converters: AUTOSCALER for C. It is able to optimize the number of shift operations by equalizing the

word-length of specific variables or constants. In [2] is presented a method that performs CDFG optimizations under accuracy constraints. It makes extensive use of the equations representing the system. In [4], is described a genetic algorithm employed to find the optimal trade-off between signal quality and implementation complexity.

This paper is based on a previous work detailed in [5]. Paper [5] addresses the same task as this paper, but is primarily focused on generating ANSI C compliant code.

III. METHOD OVERVIEW

The method presented in this paper is designed to transform dot products with constant coefficients (floating-point literals) and integer variables with known intervals:

$$\sum_{i=0}^N a_i \cdot x_i \quad (1)$$

Failing to state the correct intervals of the integer variables can lead to erroneous results. The manipulation of intervals [6] is central to the optimization procedure.

The following types of nodes are used to represent the data flow:

- **Stand-alone nodes:** nodes whose values do not depend on other nodes. Stand-alone nodes are used to represent constants, parameters: a_i , x_i , etc.
- **Operators:** *add*, *multiply*, *shift* and *change sign*. An operator has one or more operands (child nodes). These can be stand-alone nodes or other operators.

A node has an associated interval and fractional word length (FWL). The interval represents the extreme values of the node run-time integer (the memory or register variable). *Scaling a node* is a frequent operation encountered in the optimization process. **Note:** the FWL of a node is an integer value. A scaling operation does not change the fixed-point (or real) value of a node. The node interval is always altered together with the node FWL.

A node can be “realized” in code as a 16-bit or 32-bit integer. The values that can “pass through” a node must have as many as possible significant bits, to carry precise information, but, on the other hand, must be limited to a specific interval. The FWL is not necessarily the same for every node.

The data flow structure is modified in steps. A step can be viewed as an inference operation:

1. **Effect.** The integer interval of a node must be decreased or increased.

2. **List of possible causes.** A list of candidate data flow transformations is constructed.

3. **Best cause selection.** The optimal data flow transformation is selected with the help of a cost function whose coefficients represents in essence the importance given to the computational effort and to the accuracy loss.

The method described in this paper is implemented in Java (mostly because of the Java support for object-oriented programming and advanced IDEs available).

IV. DATA FLOW TRANSFORMATION

A. Problem Difficulty

The initial form of the data flow represents a true image of the floating-point dot product expression. There is one *add* operator with N child nodes – a *multiply* operator for each $a_i \cdot x_i$ term, as in (1).

Node a_i has a very long fractional part – 24 bits if the floating-point literals of the dot product expression are parsed as single precision values. If node x_i has, for example, the run-time interval equal to $[0; 1023]$, then the *multiply* node overflows. Four data types are permitted for a node: signed/unsigned 16-bit and 32-bit integers. To make the *multiply* node not to overflow it is necessary to make its integer interval smaller (to decrease the fractional part). There are two possibilities: shift to right a_i at design-time or shift to right x_i at run-time. Each solution has its own impact on the data flow computational complexity and accuracy. In this case it is simple to decide which one solution to select – discarding some least significant bits from the very precise constant at design-time means no run-time overhead and causes less accuracy loss than in comparison with the right shift of x_i . But this is a rare case. In general case, a solution is either “low-complexity-low-accuracy” or “high-complexity-high-accuracy” (not “low-complexity-high-accuracy”). This makes it difficult to compare candidate solutions. It is necessary to determine quantitatively the complexity and accuracy of a particular solution. There is no other way, because the number of the alternative possibilities grows very quickly with the size of the data flow area below the node whose integer interval must be modified.

B. Computational Complexity

A data flow node has an associated computational complexity. This is an estimator of the computational effort required to obtain at run-time the node value. At design-time the exact computational complexity is difficult to evaluate. The Java application simply counts the operators

contained in the data flow area below the target node. This is a sufficiently good approximation.

C. Node Error Interval (Drift)

Every data flow node stands for a fixed-point value which can vary within a specific interval. This interval refers to the node value *at run-time*, which is in essence an integer very close to the ideal infinite-precision value. Thus, every node has a specific error. The interval of the infinite-precision value that can “pass through” a node, and the corresponding interval of the run-time integer, can be calculated at design-time, which means that, for each node, the interval of the error can be obtained without actually running the code.

The error interval of an operator node can be calculated using the integer interval and the error interval of every child node [6]. For example, the error interval of an *add* node can be calculated by adding the error interval of every child node.

The low/high values of an error interval are considered to be absolute values; not relative values or, in other words, “units in the last place” (*ulps*) [7].

D. Multi-Objective Search

The simplest way to decrease or increase the integer interval of a node is to basically perform a shift operation. This can be done at design-time if the node represents a constant value or at run-time if the node represents an operator and (very important) the integer interval is valid (does not overflow). But these are not frequent cases. The most usual situation is when an operator overflows and its integer interval must be decreased. The problem is that the integer interval of an operator cannot be changed directly – the integer intervals of the child nodes must be altered. To force the integer interval of an *add* node it is necessary to force the integer interval of *every* child node (logical AND). To force the integer interval of a *multiply* node it is necessary to force *one or more* child nodes (logical OR).

In general case, there are multiple ways the increase or decrease the integer interval of a node (logical OR). One possible way is called a *solution*. A solution involves a number of data flow *changes* (logical AND). A change can be viewed in the simplest way as a node switch – the child node of an operator is replaced with another child node. A change is *invertible* – a change can be *applied* and can be *undone*. This is a very important feature. Because a change is always a part of a solution, it makes sense to say that a solution is applied or undone (meaning that all the changes included are applied or undone).

Multiple solutions can be viewed as being *concurrent* if all of them are built with the same purpose – for example, to make the integer interval of a specific node smaller. But

each solution consists of a number of particular (different) changes. Thus each solution has its own computational complexity and influence on accuracy. The Java application compares concurrent solutions by these two metrics. To evaluate the complexity and error interval of a solution, it is applied (some child nodes are disconnected and others are connected).

This algorithm step (switching between different solutions) is essentially a search. The method described in this paper resembles other methods if regarding the way the data flow is implemented (types of nodes) and the usage of operator properties (value propagation). From this point of view, the method described here can be considered analytical. But it still does a search! It can be regarded as being search-based, but, anyway, it is very different from other search-based methods. The method described in this paper connects/disconnects various data flow “fragments”, while other methods scan large, large multi-dimensional spaces that represent fractional word-lengths.

To compare several concurrent solutions it is necessary to combine, for each solution, the complexity and error interval into a single indicator. In order to do this, a linear function is used:

$$cost = k_1 * complexity + k_2 * error \quad (2)$$

Varying the cost function coefficients is like, for example, favoring solutions which introduce considerable computational overhead but give high accuracy results in place of „low-complexity-low-accuracy” solutions.

Although the cost function has two parameters, the variation space is one-dimensional. The cost function can be represented geometrically as a line which passes through the origin point in a two-dimensional space (Fig. 1).

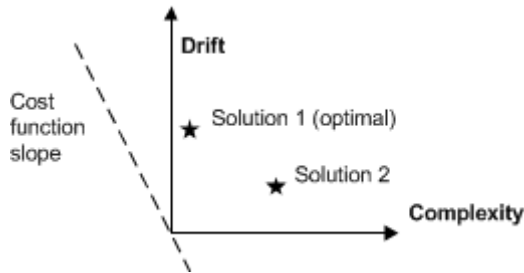


Fig. 1: Solution space

In Fig. 1 the cost of one solution is directly proportional to the shortest distance to the cost function line. The complexity coefficient (k_1) and the error coefficient (k_2) determine together the slope of cost function line. Two coefficients are used, because, otherwise it would be impossible to represent the vertical line (+INF slope). For simplicity, the sum of the cost coefficients is kept constant:

$$k_1 + k_2 = 1 \quad (3)$$

The cost of one solution has no meaning if considered separately. It does make sense only in comparison with the costs of other solutions.

E. Transformation Example

Fig. 2 shows two extreme data flow structures obtained for a dot product with 12 terms. (Such images can be created with Graphviz software.)

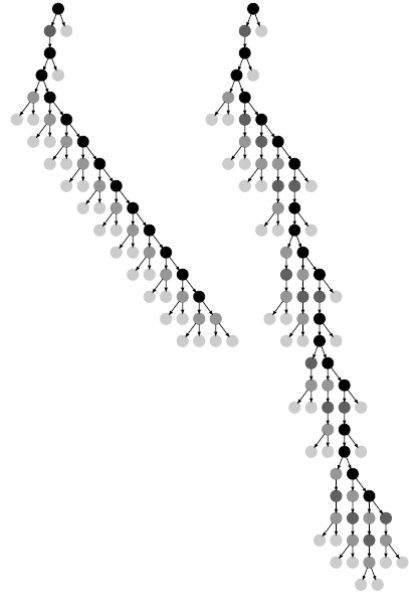


Fig. 2: High level view of two extreme data flow structures obtained for a dot product expression with 12 terms – “low-complexity-low-accuracy” (left) and “high-complexity-high-accuracy” (right).

In Fig. 2, the data flow on the left hand side has a specific pattern. The fractional word length is the same for the most part of the nodes. The number of operators is minimal (complexity = 26). In contrast, the data flow on the right hand side is very developed and doesn’t have a specific pattern (at global level). Some nodes have very long fractional parts (which is not visible). The number of operators is maximal (complexity = 53).

V. DESIGN-TIME TECHNIQUES

A. Node Cache Information

The optimization process makes extensive use of some node attributes like integer interval and drift. For operator nodes this information depends on the child nodes (operands) and must be computed. The time required for this can become significant for large data flows – the „high” nodes generate a lot of subsequent calls to nodes located below to get the necessary information. This „traffic” can be somehow diminished. The data flow structure is not itself very dynamic – a change that is applied in the optimization process has a limited impact area. In many cases the integer interval and drift information can be reused. For this

purpose, each operator node is designed with its own cache. In this way, obtaining the integer interval or the drift information can be very straightforward; unless the cache of the target node is invalidated. The invalidation of the cache is crucial. Doing this for fewer nodes than required may lead to erroneous results, and, doing this for more nodes than it is required can lower the cache hit rate.

The cache invalidation is triggered in the following manner – whenever a node N is connected with an operator F , a „message of change” is propagated along the chain of operators from F to the root operator to invalidate the corresponding cache data.

The design time is reduced considerably with node cache information. This is easy to observe as the filter length is increased. Without caching, transforming the data flow of a dot product with 16 terms can take more than 10 seconds. Turning the cache mechanism on, the data flow is optimized in tens of milliseconds.

Fig. 3 shows the execution time of the optimization procedure for dot products with different lengths (node cache information is used).

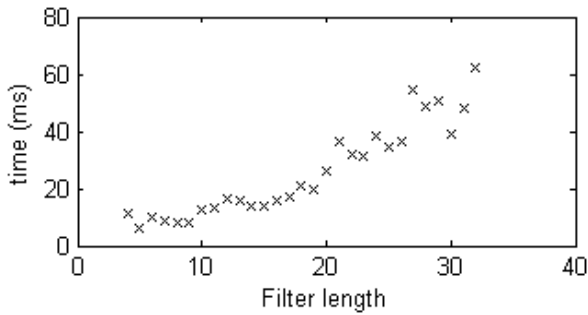


Fig. 3: Average data flow transformation time

B. Automatic Search of Data Flows

Varying the coefficients of the cost function leads to different data flow structures. The coefficients can be set by hand; but this is not very practical. The reason is that the coefficients themselves do not carry very much information (except for the extreme cases). In a concrete situation it might be more desirable to generate all the possible data flows, create code for all of them, and later select the most convenient function.

From a high level point of view, the search method, whichever is, should traverse the one-dimensional search space from 0 to 1, generate various data flows, and pick-up the unique ones. The ideal search method should generate *as few as possible equivalent* data flows. Two data flows are considered equivalent if, while traversing both structures in-depth in parallel, every node that is encountered has the same type (*add*, *multiply*), has the same integer interval (low/high values, fractional length) and has the same number of child nodes as its „mirror” node.

Performing a sequential search can be very time-consuming. Varying a coefficient from 0 to 1 using a constant step and generating all the corresponding data flows can be very inefficient. A large number of data flow structures are equivalent and the generation of a single one requires an important amount of time – for example, a dot product with 10 terms requires 10-15 milliseconds. On the other hand, the increment step has to be small enough to capture all the possible data flow structures.

Fortunately, it is possible to perform a more selective search. It is just necessary to use the following context information: if the end-points of a segment inside the search space „generate” the same data flow structure then it does not make sense to „sweep” this particular segment. No new data flow structures can be discovered in this area. But if the end-points of a segment „generate” different data flows then the segment should be halved and the same procedure should be applied further for the resulted segments. This method is very efficient, because the number of the discarded data flow structures is minimal.

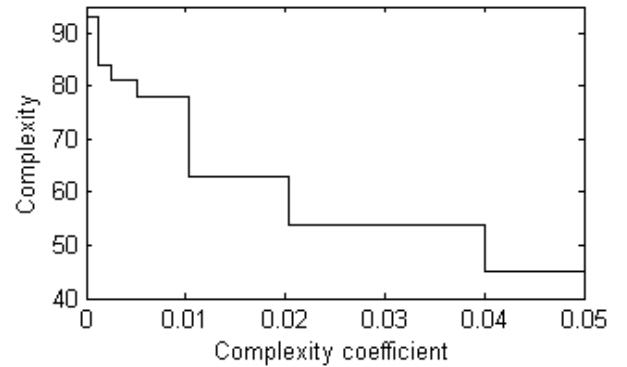


Fig. 4: Complexity of data flows that are found for a dot product with 20 terms (partial view). The complexity coefficient is swept from 0 to 1, while the drift coefficient is set to a complementary value (2). A horizontal segment represents one or more data flows with the same complexity.

Given an arbitrary filter, the number of non-equivalent data flows that can be found by varying the cost coefficients is proportional to the filter length. As a rule, if N is the filter length, then the second search method yields $0.5N \dots 1.5N$ non-equivalent data flows.

VI. CODE GENERATION

Generating fixed-point C code for a particular data flow is, in essence, a straightforward process. However, there are two important aspects: the declaration of the intermediary variables and the explicit data type casts [8].

The C code can be generated in two very different forms: as a long sequence of short assignments (one operator in every right hand side) and a lot of intermediary variables or as a single, very long, arithmetic expression and a lot of

parentheses. Although honestly both forms of code look meaningless, the first variant can be used for debugging purposes, because all the intermediary variables are declared and can be watched step by step. The second variant is preferable in case no compiler optimizations are applied.

Generating a very long line with arithmetic operators poses some problems. It is so because the compiler must deduce the data type for some subexpressions. (In case the intermediary variables are declared, their data type is clearly stated.) Examples:

- **Short multiplication.** The compiler might consider that the result of a multiplication between two 16-bit integers is a 16-bit integer. This is in general not desirable, because most *multiply* nodes produce 32-bit values; so, when the code is generated, short integers that must be multiplied are explicitly cast to long integers.

- **Signed/Unsigned arithmetic.** There are cases when a signed integer is added with an unsigned integer and the result is known to be nonnegative, but the compiler assumes that it is signed. If such an integer must be shifted to right, the compiler might do an arithmetic (not logical) shift, which is wrong, because the most significant bit would be interpreted differently. To avoid this, additional casts are inserted when generating the code.

VII. RESULTS

A. Accuracy

When a fixed-point C function is generated, the error interval of its result is already known. This is the worst-case indicator computed at design-time – the drift of the data flow root node.

Note: The error is considered as the difference between the floating-point value obtained with the original floating-point expression (the reference value) and the integer value obtained with the generated fixed-point code.

A more relevant accuracy indicator is the signal-to-quantization-noise-ratio (*SQNR*) computed with the mean of the absolute reference values (S) and the mean of the absolute error values (N):

$$SQNR = 10 * \log_{10} \left(\frac{S}{N} \right) \quad (3)$$

The *SQNR* values are computed on a high-speed computer (not on microcontrollers). It is important to run the fixed-point code with as many as possible different input parameters.

Fig. 5 illustrates the accuracy and the complexity of the solutions that are found (automatically) for a dot product with 24 terms. The accuracy is represented as the difference between the highest possible (ideal) *SQNR* and the *SQNR* of the generated code. The highest possible *SQNR* is

defined as the *SQNR* of a function that would return the integer nearest to the ideal floating-point value.

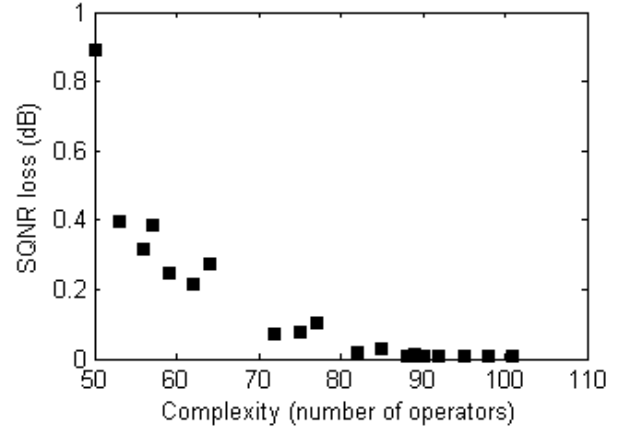


Fig. 5: Solutions found for a dot product with 24 terms, random coefficients within the interval $[-1, 1]$ and variables within the interval $[0, 4095]$

The *SQNR* degrades as the dot product number of terms grows and, especially, as the complexity cost coefficient is increased.

B. Speed

The execution time of the generated fixed-point code depends on many factors:

- **The filter.** The number of data flow nodes is directly proportional to the number of filter taps. This holds true before and after the data flow is optimized.

- **The cost function.** Varying the cost coefficients leads to specific data flow transformation decisions (as discussed in section *Multi-Objective Search*).

- **The code generation.** If the fixed-point code is generated as one very long expression (everything inline), then most of the intermediary values are allocated in registers and, in effect, the number of load/store operations is decreased. This is especially important when no compiler optimizations are applied.

- **The compiler.** Turning the compiler optimizations on can greatly accelerate the fixed-point code. This is worth considering especially when the intermediary variables are declared.

- **The microprocessor.** The microprocessor capabilities are not regarded in detail, because the main purpose is to generate platform-independent code, not assembler. The only thing assumed is that there are no floating-point units, which is characteristic for embedded microprocessors. The microprocessors used for testing are shown in Table I. Some instruction sets include integer division (something that can be used instead of bitwise shift), but this is not a general feature and is not considered.

TABLE I
MICROPROCESSORS USED FOR TESTING

Microprocessor	Register width	Compiler
ATmega16	8-bit	IAR
MSP430F149	16-bit	IAR
STM32F	32-bit	gcc
LPC1768	32-bit	IAR

The speed factor between the fixed-point code and the floating-point code can vary within a wide range. One very important cause is the cost function used throughout the data flow optimization. For “low-complexity-low-accuracy” solutions the speed can be increased by 15 times or more. For “high-complexity-high-accuracy” solutions the speed can be increased by at least 3 times. (These results are obtained with floating-point dot products with 4-32 terms generated randomly.)

C. Memory Usage (Flash and SRAM)

The fixed-point code takes in general slightly more Flash space (code memory) than the floating-point code.

The SRAM (data memory) usage is determined mainly by the stack requirements. The fixed-point code, if generated as a single arithmetic expression (no intermediary variables), occupies almost no stack space. The floating-point code needs a specific amount of stack, because it is calling low-level functions.

VIII. CONCLUSIONS

A method for transforming floating-point expressions to integer C code for embedded processors is described. Direct-Form I non-adaptive filters with predefined input bounds are targeted. The algorithm presented uses a parametrizable cost function and is able to produce multiple solutions for the same given floating-point expression.

The method can be applied for FIR filters, as well as for IIR filters if the intervals of the output variables can be specified. (There is work in progress for recursive filters.)

The generated code is tested on 8-bit, 16-bit, and 32-bit microprocessors, using different compilers.

There can be two major “realizations” of the presented algorithm: as a stand-alone application for code conversion (as it is currently implemented) or as a separate type of compiler IR optimization (requires integration in a compiler system).

ANNEX

A floating-point expression is converted to fixed-point code, for illustrative purposes:

```
0.023159746f*x[0]+0.007362494f*x[1]+0.109808266f*x[2]-
0.8996903f*x[3]-0.52352905f*x[4]+0.34677517f*x[5]+
0.50765723f*x[6]+0.9989124f*x[7]+0.5545187f*x[8]-
0.73752284f*x[9]
```

This expression can be viewed as a FIR filter. The interval of the input variables $x[0-9]$ is set to $[0; 4095]$. The coefficients are generated randomly in the interval $[-1; 1]$.

The conversion to fixed-point takes 106 milliseconds and yields 11 non-equivalent data flows. ANSI-C integer code is generated. Here is the compact form of one solution (code without intermediary variables):

```
((unsigned long)(1159921664L + (((((unsigned long)7720 * (unsigned long)x[1]) + ((unsigned long)24284 * (unsigned long)x[0])) + (115142L * (unsigned long)x[2])) + (532317L * (unsigned long)x[6])) >> 2) + ((unsigned long)(1047435L * (unsigned long)x[7]) >> 2) + ((unsigned long)(581455L * (unsigned long)x[8]) >> 2) + (90905L * (unsigned long)x[5])) >> 1) - ((unsigned long)(193337L * (unsigned long)x[9]) >> 1) + ((-117924L) * (signed long)x[3]) + ((-68620L) * (signed long)x[4])) >> 17) - 8849
```

The accuracy of this fixed-point code is estimated by running $1.9e+6$ random test cases.

The SQNR of the fixed-point code is 38.694524dB. This value is 0.000098dB less than the ideal SQNR. The error distribution is as follows: in 99.80% of the cases the result of the fixed-point code is the same as the integer nearest to the floating-point expression, in 0.07% of the cases the error is 1 and in 0.13% of the cases the error is -1.

IAR Embedded Workbench for ARM is used to measure the performance of the integer code. LPC1768, which is an ARM Cortex-M3 microprocessor, is selected as the target architecture. It occurs that, without compiler optimizations, in Simulator, the floating-point code takes 737-754 cycles and the integer code takes 50 cycles. Thus, the execution time is decreased by approximately 15 times.

REFERENCES

- [1] K. I. Kum, J. Kang, W. Sung, “AUTOSCALER For C: An Optimizing Floating-Point to Integer C Program Converter For Fixed-Point Digital Signal Processors”, *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 47, issue 9, pp. 840-848, Sep. 2000.
- [2] D. Menard, D. Chillet, F. Charot, O. Sentieys, “Automatic Floating-point to Fixed-point Conversion for DSP Code Generation”, in *Proc. of the 2002 International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*, Oct. 2002.
- [3] C. Shi, R. W. Brodersen, “An Automated Floating-point to Fixed-point Conversion Methodology”, in *Proc. of IEEE International Conf. on Acoustics, Speech, and Signal Processing*, vol. II, pp. 529-32, 2003.
- [4] K. Han, “Automating transformations from floating-point to fixed-point,” Ph.D. dissertation, Faculty of the Graduate School of the University of Texas at Austin, 1996.
- [5] A. Bârleanu, V. Băitoiu, A. Stan, „Digital filter optimization for C language,” in *Advances in Electrical and Computer Engineering*, to be published.
- [6] R. B. Kearfott, “Interval Computations: Introduction, Uses, and Resources,” *Euromath Bulletin*, vol. 2, no. 1, pp. 95-112, 1996.
- [7] D. Goldberg, “What Every Computer Scientist Should Know About Floating-Point Arithmetic,” *ACM Computing Surveys (CSUR)*, vol. 23, issue 1, 1991.
- [8] Programming languages — C, International Standard, ISO/IEC 9899:TC2.
- [9] R. J. Mitchell and P.R. Minchinton, —A Note on Dividing Integers by Two, *The Computer Journal*, 32, No. 4, Aug 1989, 380.

Decentralized set-valued state estimation and prediction for hybrid systems: A symbolic approach

N. Bajcinca and Y. Kouhi

Abstract—A symbolic approach to decentralized set-valued state estimation and prediction for systems that admit a hybrid state machine representations is proposed. The decentralized computational scheme represents a conjunction of a finite number of distributed state machines, which are specified by an appropriate decomposition of the external signal space. It aims at a distribution of computational tasks into smaller ones, allocated to individual distributed state machines, leading to a potentially significant reduction in the overall space/time computational complexity. We show that, in general, such a scheme outerapproximates the state set estimates and predictions of the original monolithic state machine. By utilizing structural properties of the transition relation of the latter, in a next step, we propose constructive decomposition algorithms for a recovery of the exact state set outcomes.

Index Terms—Decentralized estimation, distributed estimation, hybrid systems, discrete event systems

I. INTRODUCTION

Conservative set-valued based computation has versatile applications in the analysis and synthesis of complex systems. In particular, such an approach can be efficiently employed for the prediction of the system's behaviour comprehending physical and measurement uncertainties. Its usability is justified as the state set estimates are guaranteed to contain the true system state. However, the estimation computational cost increases, in general, exponentially with the system's state space dimension. Therefore, the decentralized approach, leading to a potential reduction in computational complexity, has received a considerable attention, particularly in discrete-event systems (see e.g. [1] and its references).

This article follows a similar approach in dealing with the state estimation problem using hybrid state machine representations with a finite external signal space. Such symbolic models are obtained by a discrete approximation of the behaviour of continuous, discrete-event and hybrid systems, see e.g. [2] and the references therein. For the purpose of decentralized computation, the signal space is decomposed into a finite number of subspaces, equipped with specific aggregation functions. It turns out that, in general, the decentralized scheme provides conservative overapproximate outcomes as compared to the monolithic ones. For a recovery of the exact state set estimates and predictions, constructive decomposition algorithms by utilizing the structural properties of the transition relation of the monolithic state machine are devised. Therefore the simple concept of

“non-deterministic chains” – first introduced in [3] – has been employed. Non-deterministic chains represent a special class of transition relations featuring inherent injectivity properties in the corresponding transition functions. We show that for every transition relation that assumes a partitioning into a set of non-deterministic chains, conjunctive decentralized schemes producing exact state set estimation exist always. Moreover, by means of proper state space aggregations, we conceive an algorithm which extends this simple idea to the general class with non-injective state set transition functions.

The remaining of the paper is organized as follows. In Section II the reader is made familiar with the used notation and basic preliminary concepts. Section III represents the core of the work. We define here the conjunctive decentralized computation scheme, and introduce algorithms for the decomposition of the external signal space. Several examples are used to illustrate the main ideas and procedures.

II. PRELIMINARIES

A. Notation & conventions

Capital letters denote signal spaces, e.g. X and W represent the state space and the external signal space, respectively. The corresponding elements of a space are denoted by greek lowercase letters, e.g. $W = \{\omega_1, \dots, \omega_m\}$, $\xi \in X$, etc. We consider the discrete time domain, hence signals, which are denoted by lowercase letters, are sequences of symbols from the appropriate signal space, e.g. $w : \mathbb{N}_0 \rightarrow W$ represents an external signal. The restriction of a signal, or a set of signals to an interval $[\tau, t]$, with $0 \leq \tau \leq t \in \mathbb{N}_0$, is denoted by $\cdot|_{[\tau, t]}$, e.g. $w|_{[\tau, t]} = w(\tau)w(\tau+1) \dots w(t)$. The space of the finite sequences (strings) $w|_{[\tau, t]}$ will be denoted as $W^{[\tau, t]} = W^{t-\tau+1}$. The string $w|_{[\tau, t]}$ will be considered as an element of $W^{[\tau, t]}$, i.e. they will be represented by an $(t-\tau+1)$ -tuple ordered by the time parameter.

Let $\sim$ be an equivalence relation defined on a set W . The equivalence class of an element $\alpha \in W$ is the subset of all elements in W which are equivalent to α , that is: $[\alpha] = \{\omega \in W; \omega \sim \alpha\}$. The quotient set of W by $\sim$ is defined as $W/\sim = \{[\alpha]; \alpha \in W\}$. The canonical projection map $\pi : W \rightarrow W/\sim$ maps elements of W to equivalence classes. Finally, the equivalence classes will be labeled by symbols with the help of an injective function $L : W/\sim \rightarrow V = \{\theta_1, \dots, \theta_q\}$, which we refer to as a labeling function.

Let f be an arbitrary function f defined on some domain X . If Ξ is a subset of X , we will use the convention $f(\Xi) := \cup_{x \in \Xi} f(x)$. For singletons we often avoid the brackets, that is instead of $\{\omega\}$ we rather use ω .

N. Bajcinca and Y. Kouhi are with *Max Planck Institute for Dynamics of Complex Technical Systems*, Sandtorstr. 1, 39106 Magdeburg, Germany, and with *Technische Universität Berlin, Control Systems Group*, Einsteinufer 17, EN 11, 10587 Berlin, Germany. Corresponding email: bajcinca@mpi-magdeburg.mpg.de.

B. Systems & realizations

A dynamical system Σ is defined as a triple $(T, W, \mathcal{B})$, with *time axis* $T \subseteq \mathbb{R}$, the *external signal space* W , and the *behaviour* $\mathcal{B} \subseteq W^T$, where $W^T = \{w : T \rightarrow W\}$, see [4]. In words, $\mathcal{B}$ represents a family of sequences $w : \mathbb{N}_0 \rightarrow W$ which are compatible with the dynamics of the system Σ . A *state machine* is defined as a tuple $P = (X, W, \Delta, X_0)$ where X denotes the state space, W the external signal space, $\Delta \subseteq X \times W \times X$ the transition relation, and $X_0 \subseteq X$ the initial state set. If $X = \mathbb{R}^n \times D$, where $n \in \mathbb{N}$ and $D \subset \mathbb{N}$ is a finite set, then P is referred to as a *hybrid state machine*; for $n = 0$, P is a *finite state machine*. For the sake of simplicity, here P is assumed generally to be non-blocking and $X_0 = X$.

Hybrid state machines cover a wide range of system classes including the time-driven (continuous), event-driven and hybrid systems. The synthesis of the transition relation Δ consists in symbolic encoding of the system behaviour $\mathcal{B}$ in terms of the transitions between the states in X . For illustration purposes, consider a time-driven continuous system defined by $\dot{\xi} = a(\xi)$ where $\xi \in X \subseteq \mathbb{R}^n$ and $a : X \rightarrow X$. Introduce the external signal space $W = L(X/Q)$, where Q represents a finite equivalence relation in X , and L a labeling function. Unique solutions $\phi(t, \xi)$ can be associated with each initial value $\xi \in X$ if $a : X \rightarrow X$ is Lipschitz on X . Then, $(\xi, \omega, \xi') \in \Delta$ if $\xi' = \phi(T_s, \xi)$ with T_s representing the sampling time, and $L(\pi_Q(\xi')) \neq L(\pi_Q(\xi)) = \omega \in W$. If $\phi(t, \xi) \in \pi_Q(\xi)$ for all $t \in \mathbb{R}$, then we adopt $(\xi, \omega, \xi) \in \Delta$. Other encoding scenarios for the transition relation Δ can be utilized alternatively.

For systems exhibiting an input/output structure, the external signal space W can be decomposed as $W = U \times Y$, with U and Y being the sets of input and output symbols. Then $P = (X, U \times Y, \Delta, X_0)$ is said to be an *I/S/- machine* if for each reachable state $\xi \in X$ and each $\mu \in U$, there exists a $\nu \in Y$ and a $\xi' \in X$, such that $(\xi, (\mu, \nu), \xi') \in \Delta$. If ν and ξ' are unique for all $\xi \in X$ and $\mu \in U$, P is said to be an *I/S/O machine*. Note that I/S/O machines are by definition deterministic, that is, if $(\xi, (\mu, \nu), \xi') \in \Delta$ and $(\xi, (\mu, \nu), \xi'') \in \Delta$, then $\xi' = \xi''$.

A state machine $P = (X, W, \Delta, X_0)$ induces a state space system $\Sigma_S = (\mathbb{N}_0, W \times X, \mathcal{B}_S)$, where $\mathcal{B}_S$ is referred to as the *full behaviour*, and is defined as

$$\mathcal{B}_S := \{(w, x); (x(t), w(t), x(t+1)) \in \Delta, t \in \mathbb{N}_0, x_0 \in X_0\}. \quad (1)$$

The *external behaviour* $\mathcal{B}_{\text{ex}}$ of Σ_S is then defined to be the projection of $\mathcal{B}_S$ onto $W^{\mathbb{N}_0}$, that is $\mathcal{B}_{\text{ex}} := \mathcal{P}_W \mathcal{B}_S = \{w; \exists x \in X^{\mathbb{N}_0}, (w, x) \in \mathcal{B}_S\}$. A *state space path* associated with a string $w|_{[\tau, t]}$, denoted as $(\xi_\tau, w|_{[\tau, t]}, \xi_{t+1})$, is said to be *present* in the state machine P , if $(w, x) \in \mathcal{B}_S$ exists such that $\xi_\tau = x(\tau)$ and $\xi_{t+1} = x(t+1)$. Finally, a state machine $P = (X, W, \Delta, X_0)$ with induced external behaviour $\mathcal{B}_{\text{ex}}$ is called a realization of a dynamical system $\Sigma = (\mathbb{N}_0, W, \mathcal{B})$ if $\mathcal{B}_{\text{ex}} = \mathcal{B}$. This will be denoted by $P \cong \Sigma$.

C. State sets

Let $\mathcal{B}_S$ and $\mathcal{B}_{\text{ex}}$ be the induced full and external behaviour of the state machine $P = (X, W, \Delta, X)$, respectively, and

consider a string $w|_{[\tau, t]} \in W^{[\tau, t]}$. The sets defined by

$$\chi(w|_{[\tau, t]}) := \{\xi; \exists (w', x) \in \mathcal{B}_S, x(t) = \xi, w'|_{[\tau, t]} = w|_{[\tau, t]}\}, \quad (2a)$$

$$\rho(w|_{[\tau, t]}) := \{\xi; \exists (w', x) \in \mathcal{B}_S, x(t+1) = \xi, w'|_{[\tau, t]} = w|_{[\tau, t]}\}, \quad (2b)$$

will be referred to as the *estimated* and *predicted* state sets compatible with the external finite sequence $w|_{[\tau, t]}$, respectively. Both, χ and ρ , represent families of set-valued functions $W^{[\tau, t]} \rightarrow 2^X$. Note that $w|_{[\tau, t]} \in \mathcal{B}_{\text{ex}}|_{[\tau, t]} \Leftrightarrow \chi(w|_{[\tau, t]}) \neq \emptyset$ and $\rho(w|_{[\tau, t]}) \neq \emptyset$.

In general, more information from the past leads to more accurate state estimations and predictions. This fact is reflected by the following general inclusion relationships

$$\chi(w|_{[0, t]}) \subseteq \chi(w|_{[1, t]}) \subseteq \dots \subseteq \chi(w|_{[t, t]}), \quad (3a)$$

$$\rho(w|_{[0, t]}) \subseteq \rho(w|_{[1, t]}) \subseteq \dots \subseteq \rho(w|_{[t, t]}). \quad (3b)$$

Introduce, further, the *parametrized state transition* function $\hat{\rho}_\omega : X \rightarrow 2^X$, where $\omega \in W$, with

$$\hat{\rho}_\omega(\xi) := \{\xi'; (\xi, \omega, \xi') \in \Delta\}. \quad (4a)$$

For $\Omega \subseteq W$, in accordance with the adopted convention

$$\hat{\rho}_\Omega(\Xi) := \bigcup_{\omega \in \Omega, \xi \in \Xi} \hat{\rho}_\omega(\xi). \quad (4b)$$

Then, the predicted states in X , resulting from the occurrence of the symbol $\omega \in W$, are computed by

$$\rho(\omega) = \hat{\rho}_\omega(\chi(\omega)), \quad (4c)$$

where according to (2a)

$$\chi(\omega) = \{\xi; \exists \xi', (\xi, \omega, \xi') \in \Delta\}, \quad (4d)$$

with $\omega = w|_{[t, t]}$. For sequences,

$$\rho(w|_{[\tau, t]}) = \hat{\rho}_{w(t)}(\chi(w|_{[\tau, t]})). \quad (4e)$$

Notice, that in light of this equation, (3b) follows directly from (3a) as the set in the argument of $\hat{\rho}_{w(t)}$ is shrinking. Moreover, by definition (2a)

$$\chi(w|_{[\tau, t]}) = \rho(w|_{[\tau, t-1]}) \cap \chi(w(t)), \quad (4f)$$

which along with (4e) reveals a recursive structure in computing $\chi(w|_{[\tau, t]})$ and $\rho(w|_{[\tau, t]})$. Observe that now in view of (3b), equation (3a) follows immediately from (4f).

III. DECENTRALIZED COMPUTATION

A. Signal space decomposition

Introduce a finite set of equivalence relations A_k on the external signal space W , with π_k representing the corresponding canonical projection map

$$\pi_k : W \rightarrow W/A_k, \quad k \in \{1, \dots, p\}. \quad (5)$$

Introduce functions $L_k : W/A_k \rightarrow V_k$ which assign to each equivalence class $[\omega]_{A_k}$ a labeling symbol θ_k . The composition $\mathcal{A}_k := L_k \circ \pi_k$,

$$\mathcal{A}_k : W \rightarrow V_k, \quad k \in \{1, \dots, p\}, \quad (6)$$

ushers V_k as an “aggregate” space of the external signal space W . Formally, we denote this by $V_k = L_k(W/A_k)$.

Throughout the article we consider classes of equivalence relations A_k , $k \in \{1, \dots, p\}$, which fulfill the following *consistency condition*: $\cap_{k=1}^p [\omega]_{A_k} = \omega, \forall \omega \in W$. With regard to (6), this is equivalently stated by

$$\cap_{k=1}^p \mathcal{A}_k^{-1}(\mathcal{A}_k(\omega)) = \omega \quad (\forall \omega \in W). \quad (7)$$

Then, each symbol $\omega \in W$ is uniquely associated with an ordered p -tuple of symbols $(\theta_1, \dots, \theta_p) \in V_1 \times \dots \times V_p$. This will be referred to as the *decomposition* of the signal space W and formally designated as

$$W \rightsquigarrow V_1 \times \dots \times V_p. \quad (8)$$

Note that in general the opposite does not hold: not every p -tuple in $V_1 \times \dots \times V_p$ has an associate in W . Throughout the article we will adhere to the consistency condition (7).

Now consider the space of infinite sequences $W^{\mathbb{N}_0}$, and introduce the equivalence relation A_k thereon as: $(w', w'') \in A_k$ if $(w'(t), w''(t)) \in A_k$, for all $t \in \mathbb{N}_0$. In this sense, each sequence $w \in W^{\mathbb{N}_0}$ is assigned an equivalence class $[w]_{A_k} := \prod_t [w(t)]_{A_k} = [w(0)]_{A_k} \times [w(1)]_{A_k} \times \dots$ whereby in accordance with the adopted conventions in Section II-A, the sequences in $W^{\mathbb{N}_0}$ are represented by ∞ -tuples. A natural labeling policy for such equivalence classes in $W^{\mathbb{N}_0}/A_k$ by making use of those for W/A_k leads to labels in form of sequences in $V_k^{\mathbb{N}_0}$: $L_k([w]_{A_k}) := L_k([w(0)]_{A_k}) L_k([w(1)]_{A_k}) \dots = \theta_k^0 \theta_k^1 \dots$, where $\theta_k^t = L_k([w(t)]_{A_k})$, $\forall t \in \mathbb{N}_0$. This infers the labeling map $L_k := W^{\mathbb{N}_0}/A_k \rightarrow V_k^{\mathbb{N}_0}$, which leads to the extension of (6) to $\mathcal{A}_k : W^{\mathbb{N}_0} \rightarrow V_k^{\mathbb{N}_0}$, $k \in \{1, \dots, p\}$ by $\mathcal{A}_k = L_k \circ A_k$. It is now obvious that if consistency condition (7) for symbols holds, so it holds for sequences, as well. Specifically,

$$\cap_{k=1}^p \mathcal{A}_k^{-1}(\mathcal{A}_k(w|_{[\tau, t]})) = w|_{[\tau, t]} \quad (w \in W^{\mathbb{N}_0}). \quad (9)$$

Then a string $w|_{[\tau, t]}$ is always associated with a unique tuple of strings $(v_1|_{[\tau, t]}, \dots, v_p|_{[\tau, t]}) \in V_1^{\tau, t} \times \dots \times V_p^{\tau, t}$. It can be also shown that for any $w \in W^{\mathbb{N}_0}$ and $\mathcal{A}_k$

$$\mathcal{A}_k^{-1}(\mathcal{A}_k(w|_{[\tau, t]})) = \mathcal{A}_k^{-1}(\mathcal{A}_k(w))|_{[\tau, t]}. \quad (10)$$

Note also that the aggregation maps $\mathcal{A}_k$ associate with each $w|_{[\tau, t]} \in W^{\tau, t}$ a subset of $W^{\tau, t}$ defined by

$$\mathcal{D}_A(w|_{[\tau, t]}) := \cup_{k=1}^p \mathcal{A}_k^{-1}(\mathcal{A}_k(w))|_{[\tau, t]}. \quad (11)$$

The behaviour of the map ρ within such a domain for a fixed $w|_{[\tau, t]}$ will play a key role in our forthcoming derivations.

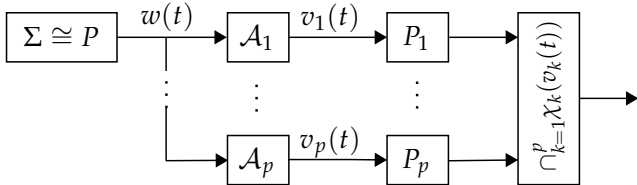


Fig. 1. Decentralized set-valued state estimation scheme.

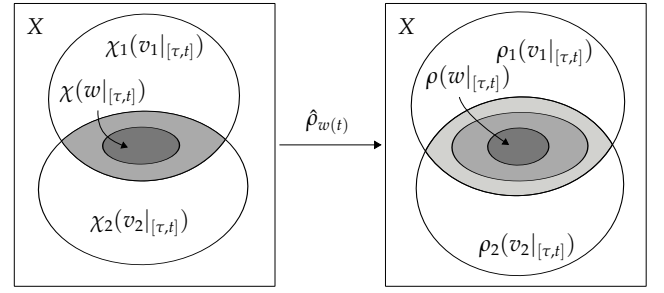


Fig. 2. Illustrating the decentralized set-valued state estimation and prediction in a setting with two state machines P_1 and P_2 exerting the strings $v_1|_{[\tau, t]}$ and $v_2|_{[\tau, t]}$, respectively, which are consistent to $w|_{[\tau, t]}$; refer to (14a) and (14b).

B. Distributed state machines

The equivalence relations A_k , as defined in the previous section, bring us to the concept of *distributed state machines*.

Definition 1: Consider a monolithic state machine $P = (X, W, \Delta, X)$. The distributed state machines P_k , with $k \in \{1, \dots, p\}$, induced by the equivalence relations A_k on the external signal space W are defined as $P_k = (X, V_k, \Delta_k, X)$, with $V_k = L_k(W/A_k)$, and $\Delta_k \subseteq X \times V_k \times X$ given by

$$\Delta_k = \{(\xi, \theta_k, \xi'); \exists \omega \in \mathcal{A}_k^{-1}(\theta_k), (\xi, \omega, \xi') \in \Delta\}. \quad (12)$$

Hereof it is clear that the full and external behaviour of a distributed state machine P_k are determined by those of the monolithic state machine P : $\mathcal{B}_{s,k} = \{(\mathcal{A}_k(w), x); (w, x) \in \mathcal{B}_s\}$ and $\mathcal{B}_{ex,k} = \{\mathcal{A}_k(w); w \in \mathcal{B}_{ex}\} =: \mathcal{A}_k(\mathcal{B}_{ex})$, respectively. Clearly, $\cap_{k=1}^p \mathcal{A}_k^{-1}(\mathcal{A}_k(\mathcal{B}_{ex})) \supseteq \mathcal{B}_{ex}$.

The corresponding decentralized estimation and prediction functions $\chi_k : V_k^{\tau, t} \rightarrow 2^X$ and $\rho_k : V_k^{\tau, t} \rightarrow 2^X$ of the distributed state machines P_k are defined using analogous expressions to (2a-2b). In fact, ρ_k and χ_k can be computed as

$$\chi_k = \chi \circ \mathcal{A}_k^{-1}, \quad \rho_k = \rho \circ \mathcal{A}_k^{-1}. \quad (13)$$

If ρ is applied on both sides of (9), then using the fact that for any function ϕ : $\phi(M_1 \cap M_2) \subseteq \phi(M_1) \cap \phi(M_2)$ we get

$$\begin{aligned} \rho(w|_{[\tau, t]}) &= \rho(\cap_{k=1}^p \mathcal{A}_k^{-1}(\mathcal{A}_k(w|_{[\tau, t]}))) \\ &\subseteq \cap_{k=1}^p \rho(\mathcal{A}_k^{-1}(\mathcal{A}_k(w|_{[\tau, t]}))) \\ &= \cap_{k=1}^p \rho_k(v_k|_{[\tau, t]}), \end{aligned} \quad (14a)$$

where (13) is used in the 3rd line, and $v_k|_{[\tau, t]} = \mathcal{A}_k(w|_{[\tau, t]})$ represent the consistent external strings to $w|_{[\tau, t]}$, exerted on the distributed state machines P_k . Similarly,

$$\cap_{k=1}^p \chi_k(v_k|_{[\tau, t]}) \supseteq \chi(w|_{[\tau, t]}). \quad (14b)$$

Equations (14a) and (14b) suggest a decentralized computation scheme for the set-valued state estimate $\chi(w|_{[\tau, t]})$ and prediction $\rho(w|_{[\tau, t]})$ involving p state machines P_k as depicted in Fig. 1. Referring to (14a) and (14b), such a parallel computation provides in general overapproximate results of the monolithic machine P ; see Fig. 2 for a mapping picture of the setup with $p = 2$. Yet for certain classes of transition relations Δ of the monolithic state machine P , specific decomposition policies (8) may lead to a decentralized scheme producing the outcomes that exactly match with those of the monolithic state machine P . To this end, our

effort will particularly consist in a suitable manipulation of the aggregation maps $\mathcal{A}_k$, or, equivalently, the signal space decomposition (8). Therefore, in this article, we employ the concept of “non-deterministic chains”.

C. Non-deterministic chains

Definition 2: Consider a state machine $P=(X, W, \Delta, X)$. A transition subrelation $\delta \subseteq \Delta$ is said to be a *non-deterministic chain* over a subspace $\Omega \subseteq W$ if

$$\delta = \{(\xi, \omega, \xi') \in \Delta; \forall \omega \in \Omega\}, \quad (15)$$

and for all $\omega', \omega'' \in \Omega$:

- (i) $(\xi, \omega', \xi') \in \delta, (\xi, \omega'', \xi') \in \delta \Rightarrow \omega' = \omega''$,
- (ii) $(\xi', \omega', \xi) \in \delta, (\xi'', \omega'', \xi) \in \delta \Rightarrow \xi' = \xi'', \omega' = \omega''$.

A non-deterministic chain δ over the subspace Ω can naturally be assigned the transition functions $\chi^c : \Omega \rightarrow 2^X$, $\rho^c : \Omega \rightarrow 2^X$, and $\hat{\rho}^c : \chi(\Omega) \rightarrow 2^X$ defined by

$$\chi^c := \chi|_{\Omega}, \quad \rho^c := \rho|_{\Omega}, \quad \hat{\rho}^c := \hat{\rho}_{\Omega}, \quad (16)$$

where χ, ρ and $\hat{\rho}_{\Omega}$ refer to the state machine P , as discussed in Section II-C. Then, (i) can be equivalently restated as $\chi^c(\omega') \cap \chi^c(\omega'') \neq \emptyset \Rightarrow \omega' = \omega''$, while (ii) is equivalent to $\hat{\rho}^c(\xi') \cap \hat{\rho}^c(\xi'') \neq \emptyset \Rightarrow \xi' = \xi''$. This leads us to the following statement.

Proposition 1: A transition relation δ over Ω is a non-deterministic chain if and only if $\chi^c : \Omega \rightarrow 2^X$ and $\hat{\rho}^c : \chi(\Omega) \rightarrow 2^X$ are absolutely injective set-valued maps.

Note that by definition (16) the functions χ^c and $\hat{\rho}^c$ take symbols as arguments. As a consequence of absolute injectivity of $\hat{\rho}^c$, for $w(t) \in \Omega$, from (4e) and (4f) it follows

$$\rho(w|_{[\tau, t]}) = \hat{\rho}^c [\chi^c(w(t)) \cap \rho(w|_{[\tau, t-1]})]. \quad (17)$$

This equation will shortly prove useful in designing the external space signal decomposition.

Definition 3: A state machine $P=(X, W, \Delta, X)$ is said to be *chain-decomposable* if W and Δ can be partitioned as

$$W = \cup_{j=1}^r \Omega_j \quad \text{and} \quad \Delta = \cup_{j=1}^r \delta_j, \quad (18)$$

such that for each $j \in \{1, \dots, r\}$, δ_j represents a non-deterministic chain over Ω_j .

Example 1: Consider an I/S/- machine $P = (X, U \times Y, \Delta, X)$ with *singleton output maps*. Let $U = \{\mu_j; j = 1, \dots, r\}$, and introduce the partitioning: $W = U \times Y = \cup_{j=1}^r \Omega_j$, where $\Omega_j := \mu_j \times Y$. This induces a partitioning of the transition relation: $\Delta = \cup_{j=1}^r \delta_j$. By definition, functions $f: X \times U \rightarrow 2^X$ and $h: X \times U \rightarrow Y$ exist, such that

$$(\xi, (\mu_j, \nu_j), \xi') \in \delta_j \Leftrightarrow \xi' \in f(\xi, \mu_j), \quad \nu_j = h(\xi, \mu_j). \quad (19)$$

Then, each state $\xi \in X$ can be associated with a unique symbol pair $(\mu_j, \nu_j) \in \Omega_j$. Hence, (i) in Definition 2 is fulfilled. Define $\hat{\rho}_j^c : X \rightarrow 2^X$ as $\hat{\rho}_j^c(\xi) := f(\mu_j, \xi)$, and let it be *absolutely injective*. For instance, this is always true if the underlying vector field in case of a continuous time-driven system, refers to a Lipschitz continuous function on X . Hence, δ_j is a non-deterministic chain for all $j \in \{1, \dots, r\}$. In particular, this conclusion holds if P refers to an I/S/O

machine and if $f(\mu_j, \xi)$ is injective for all $\mu_j \in U$; this case has been considered in [5]. ■

Let P be chain-decomposable and introduce *arbitrary* consistent signal space decompositions (8) for all subspaces Ω_j , $j \in \{1, \dots, r\}$ in (18):

$$\Omega_j \rightsquigarrow V_{j,1} \times \dots \times V_{j,p}, \quad (20)$$

which leads to a decomposition of the external signal space $W \rightsquigarrow V_1 \times \dots \times V_p$, as well, for instance by selecting

$$V_k = \cup_{j=1}^r V_{j,k}. \quad (21)$$

Next, we show that a decentralized scheme (see Fig. 1) built upon the distributed state machines P_k , $k \in \{1, \dots, p\}$, which are induced by the underlying equivalence relations $\mathcal{A}_k$, will produce the exact outcomes of the monolithic state machine P . Referring to (16) and Proposition 1, it is important to keep in mind that the maps

$$\chi_j^c := \chi|_{\Omega_j} \quad \text{and} \quad \hat{\rho}_j^c := \hat{\rho}_{\Omega_j} \quad (22)$$

are, per construction, absolutely injective in their domains of definition. Now, fix an external signal $w|_{[\tau, t]}$, and consider any two different strings $w'|_{[\tau, t]}$ and $w''|_{[\tau, t]}$ from the subdomain $\mathcal{D}_{\mathcal{A}}(w|_{[\tau, t]})$ from the corresponding restriction domain $\mathcal{D}_{\mathcal{A}}(w|_{[\tau, t]})$ from (11). Note also that for any $\ell \in [\tau, t]$, all elements in $\mathcal{A}_k^{-1}(\mathcal{A}_k(w(\ell)))$ belong to the same subspace Ω_j for all $k \in \{1, \dots, p\}$ and some $j \in \{1, \dots, r\}$. Then, it follows from (17) and the injectivity of $\hat{\rho}_j^c$ that for any two different strings $w'|_{[\tau, t]}$ and $w''|_{[\tau, t]}$ from the subdomain $\mathcal{D}_{\mathcal{A}}(w|_{[\tau, t]})$:

$$\begin{aligned} \rho(w'|_{[\tau, t]}) \cap \rho(w''|_{[\tau, t]}) &= \\ &= \hat{\rho}_j^c [\chi_j^c(w'(t)) \cap \chi_j^c(w''(t)) \cap \rho(w'|_{[\tau, t-1]}) \cap \rho(w''|_{[\tau, t-1]})], \end{aligned}$$

for some $j = j(t) \in \{1, \dots, r\}$. Due to the injectivity of χ_j^c , the argument on the right-hand side is an empty set, unless $w'(t) = w''(t) = w(t)$. In the latter case, the computation shrinks to the interval $[\tau, t-1]$:

$$\begin{aligned} \rho(w'|_{[\tau, t]}) \cap \rho(w''|_{[\tau, t]}) &= \\ &= \hat{\rho}_j^c [\chi_j^c(w(t)) \cap [\rho(w'|_{[\tau, t-1]}) \cap \rho(w''|_{[\tau, t-1]})]]. \end{aligned}$$

Yet a largest $\ell \in \{\tau, \dots, t-1\}$ must exist, such that $w'(\ell) \neq w''(\ell)$. Then, after repeating this procedure $t - \ell$ times, the expression $\chi_j^c(w'(\ell)) \cap \chi_j^c(w''(\ell))$ appears in the argument of the function $\hat{\rho}_j^c$ for some $j = j(\ell) \in \{1, \dots, r\}$, which is per construction empty, implying that the whole left hand side expression in the above equation must be empty, as well. In other words, $\rho : W^{[\tau, t]} \rightarrow 2^X$ results to be absolutely injective within the subdomain $\mathcal{D}_{\mathcal{A}}(w|_{[\tau, t]})$ inferred by the string $w|_{[\tau, t]}$. As a direct consequence:

$$\begin{aligned} \cap_{k=1}^p \rho_k(v_k|_{[\tau, t]}) &= \cap_{k=1}^p \rho(\mathcal{A}_k^{-1}(\mathcal{A}_k(w|_{[\tau, t]}))) \\ &= \rho(\cap_{k=1}^p \mathcal{A}_k^{-1}(\mathcal{A}_k(w|_{[\tau, t]}))) \\ &= \rho(w|_{[\tau, t]}). \end{aligned} \quad (23)$$

Theorem 1: Consider a chain-decomposable state machine $P=(X, W, \Delta, X)$ and a decentralized computation setting involving distributed state machines P_k (see Fig. 1) induced

by the equivalence relations A_k , $k \in \{1, \dots, p\}$, resulting from the external signal space decomposition given by (20) and (21). Then, for any $w|_{[\tau, t]} \in W^{[\tau, t]}$:

$$\cap_{k=1}^p \chi_k(v_k|_{[\tau, t]}) = \chi(w|_{[\tau, t]}), \quad (24a)$$

$$\cap_{k=1}^p \rho_k(v_k|_{[\tau, t]}) = \rho(w|_{[\tau, t]}). \quad (24b)$$

Proof: The case $\tau = t$ in (24a) follows immediately from the absolute injectivity of a non-deterministic chain. In other cases, (24a) results after combining (24b), the recursive formula (4f) for χ and χ_k , as well as the absolute injectivity of the maps χ_j^c for all $j \in \{1, \dots, r\}$:

$$\begin{aligned} \cap_{k=1}^p \chi_k(v_k|_{[\tau, t]}) &= \cap_{k=1}^p \rho_k(v_k|_{[\tau, t-1]}) \cap \chi_k(v_k(t)) \\ &= \rho(w|_{[\tau, t-1]}) \cap \chi_j^c(\cap_{k=1}^p \mathcal{A}_k^{-1}(v_k(t))) \\ &= \chi(w|_{[\tau, t]}). \end{aligned} \quad (25)$$

Note that the derivations above put no further requirement but the consistency condition for the decomposition (20) of a signal subspace Ω_j corresponding to a nondeterministic chain δ_j , $j \in \{1, \dots, r\}$. This is a unique feature of nondeterministic chains. Yet, the opposite is not necessarily true: a transition relation δ , which guarantees (25) under any decomposition (8), need not be a nondeterministic chain. ■

Example 2: Consider the finite state machine in Fig. 3. Observe that it is chain decomposable. One way for partitioning its external signal space in accordance with (3) is $W = \Omega_1 \cup \Omega_2$ with $\Omega_1 = \{a_1, b_1, c_1, d_1\}$ and $\Omega_2 = \{a_2, b_2, c_2, d_2\}$.

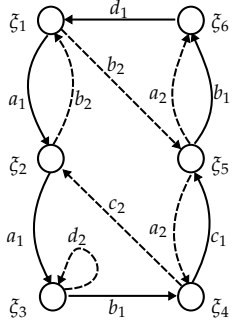


Fig. 3. Finite state machine.

As indicated by the solid and dashed lines, the corresponding transition relations δ_1 and δ_2 as defined by (15) are both non-deterministic chains. Note that δ_2 is blocking and disconnected, which is a legal matter. According to the previous elaborations, any consistent decomposition (20) on subspaces Ω_1 and Ω_2 is legal. For instance, a particular is obtained from the selections

$$\begin{aligned} V_{1,1} &= \{\theta_1^1, \theta_1^2\} \text{ with } \theta_1^1 = L_1(\{a_1, b_1\}), \theta_1^2 = L_1(\{c_1, d_1\}), \\ V_{1,2} &= \{\theta_2^1, \theta_2^2\} \text{ with } \theta_2^1 = L_2(\{a_1, c_1\}), \theta_2^2 = L_2(\{b_1, d_1\}), \\ V_{2,1} &= \{\theta_3^1, \theta_3^2\} \text{ with } \theta_3^1 = L_1(\{a_2, b_2\}), \theta_3^2 = L_1(\{c_2, d_2\}), \\ V_{2,2} &= \{\theta_4^1, \theta_4^2\} \text{ with } \theta_4^1 = L_2(\{a_2, c_2\}), \theta_4^2 = L_2(\{b_2, d_2\}), \end{aligned}$$

where L_1 and L_2 designate labeling functions. The inferred decomposition then reads $W \rightsquigarrow V_1 \times V_2$, where $V_1 = V_{1,1} \cup V_{2,1}$ and $V_2 = V_{1,2} \cup V_{2,2}$, leading to the distributed state machines $P_1 = (X, V_1, \Delta_1, X)$ and $P_2 = (X, V_2, \Delta_2, X)$. Now, consider a string, e.g. $w|_{[0,1]} = a_1 b_2$. The corresponding estimate of the monolithic machine is $\chi(a_1 b_2) = \xi_2$. The distributed machines measure accordingly the strings $v_1|_{[0,1]} = \theta_1^1 \theta_1^3$ and $v_2|_{[0,1]} = \theta_2^1 \theta_2^4$, providing the estimates $\chi_1(\theta_1^1 \theta_1^3) = \xi_2$ and $\chi_2(\theta_2^1 \theta_2^4) = \{\xi_2, \xi_3\}$, respectively. The decentralized estimate is thus given by $\chi(\theta_1^1 \theta_1^3) \cap \chi(\theta_2^1 \theta_2^4) = \xi_2$, which is exactly the same outcome obtained by the monolithic state machine $P = (X, W, \Delta, X)$. According to Theorem 1, this must hold for all strings accepted by the machine P . ■

A state machine $P = (X, W, \Delta, X)$ involving a transition relation Δ such that $(\xi', \omega, \xi) \in \Delta$ and $(\xi'', \omega, \xi) \in \Delta$, with $\xi' \neq \xi''$ is not chain-decomposable due to the violation of the injectivity condition (ii) in Definition 2. On the other hand, condition (i) is always fulfilled by the trivial signal space partitioning (3) given by $\Omega_j = \omega_j$, $j \in \{1, \dots, m\}$.

Proposition 2: A state machine $P = (X, W, \Delta, X)$ is chain decomposable if and only if $\hat{\rho}_W: W \rightarrow 2^X$ is absolutely injective.

D. Generalized decomposition rules

Consider the state machine $P = (X, W, \Delta, X)$, and an equivalence relation Q on the state space X , with $\pi_Q: X \rightarrow X/Q$ representing its canonical projection. Again, introduce a labeling for the equivalence classes $L_Z: X/Q \rightarrow Z$, and the map $\mathcal{Q} := L_Z \circ \pi_Q$:

$$\mathcal{Q}: X \rightarrow Z, \quad (26)$$

which represents the formal definition of an “aggregate” state space Z , denoted by $Z = L_Z(X/Q)$.

Definition 4: The quotient state machine $T = P/Q$ of $P = (X, W, \Delta, X)$ by the state space equivalence relation Q is defined by $T = (Z, W, \Lambda, Z)$, with $Z = L_Z(X/Q)$ and $\Lambda \subseteq Z \times W \times Z$ given by

$$\Lambda = \{(\zeta, \omega, \zeta'); \exists \xi \in Q^{-1}(\zeta), \xi' \in Q^{-1}(\zeta'), (\xi, \omega, \xi') \in \Delta\} \quad (27)$$

Then, set-valued state functions $\chi': W \rightarrow 2^Z$, $\rho': W \rightarrow 2^Z$ and $\hat{\rho}'_\Omega: Z \rightarrow 2^Z$, $\Omega \subseteq W$, associated with T exist such that

$$\chi' := \pi_Q \circ \chi, \quad \rho' := \mathcal{Q} \circ \rho, \quad \hat{\rho}'_\Omega := \mathcal{Q} \circ \hat{\rho}_\Omega, \quad (28)$$

where $\chi, \rho, \hat{\rho}$ refer to the state machine P , as defined in Section II-C. The functions χ' and ρ' are further naturally extended to mappings $W^{[\tau, t]} \rightarrow 2^Z$ in accordance with Section II-C. It is hereof important to state that by definition

$$\chi(\omega) \subseteq Q^{-1} \circ \chi'(\omega) \quad (29)$$

and

$$\rho(w|_{[\tau, t]}) \subseteq Q^{-1} \circ \rho'(w|_{[\tau, t]}) \quad (30)$$

hold for any $\omega \in W$ and any $w|_{[\tau, t]} \in W^{[\tau, t]}$, respectively.

Proposition 2 indicates that Theorem 1 does not apply for a state machine $P = (X, W, \Delta, X)$ if and only if the transition relation Δ implies non-injective state transition functions, i.e. if $\hat{\rho}_\omega(\xi') \cap \hat{\rho}_\omega(\xi'') \neq \emptyset$ for some $\omega \in W$ and $\xi' \neq \xi'' \in X$. Then, no non-deterministic chain δ in P exists over some subspace Ω if $\omega \in \Omega$. To overcome this difficulty, in this section, we upgrade the algorithm for signal space decomposition by originating an equivalence relation Q on the state space X inducing a chain-decomposable quotient state machine $T = P/Q$. As we show shortly, this effectively leads to a further restriction of the region $\mathcal{D}_A(w|_{[\tau, t]})$ from (11), such that $\chi|_{\mathcal{D}_A(w|_{[\tau, t]})}$ and $\rho|_{\mathcal{D}_A(w|_{[\tau, t]})}$ are absolutely injective functions for any $w|_{[\tau, t]} \in W^{[\tau, t]}$.

Lemma 1: Consider a state machine $P = (X, W, \Delta, X)$ and let Q be an equivalence relation defined by: $(\xi', \xi'') \in Q$ if $\xi \in X$ and $w|_{[\tau, t]} \in W^{[\tau, t]}$ exist, such that state

space paths $(\xi', w|_{[\tau, t]}, \xi)$ and $(\xi'', w|_{[\tau, t]}, \xi)$ are present in P . Then, the quotient state machine P/Q is chain-decomposable.

Proof: We provide the proof by contradiction. Let $T = P/Q = (Z, W, \Lambda, Z)$, and suppose that condition (ii) in Definition 2 is violated. Then, two different transitions $(\zeta', \omega, \zeta) \in \Lambda$ and $(\zeta'', \omega, \zeta) \in \Lambda$ must exist, with $\zeta' \neq \zeta''$. As a consequence, referring to Definition 4, $(\xi', \omega, \xi_1) \in \Delta$ and $(\xi'', \omega, \xi_2) \in \Delta$ exist, such that $\xi' \in \pi_Q^{-1}(\zeta')$, $\xi'' \in \pi_Q^{-1}(\zeta'')$ and $\xi_1, \xi_2 \in \pi_Q^{-1}(\zeta)$. Due to $\xi' \neq \xi''$, $\xi_1 \neq \xi_2$ ensues, otherwise $(\xi', \xi'') \in Q$ would entail $\zeta' = \zeta''$. As a consequence, with $(\xi_1, \xi_2) \in Q$, $w|_{[\tau, t]} \in W^{[\tau, t]}$ and $\xi \in X$ must exist, such that the paths $(\xi_1, w|_{[\tau, t]}, \xi)$ and $(\xi_2, w|_{[\tau, t]}, \xi)$ are present in P . But, $(\xi', \langle \omega, w|_{[\tau, t]} \rangle, \xi)$ and $(\xi'', \langle \omega, w|_{[\tau, t]} \rangle, \xi)$ are then present, as well, implying $(\xi', \xi'') \in Q$, which is in contradiction with the initial assumption of the proof. ■

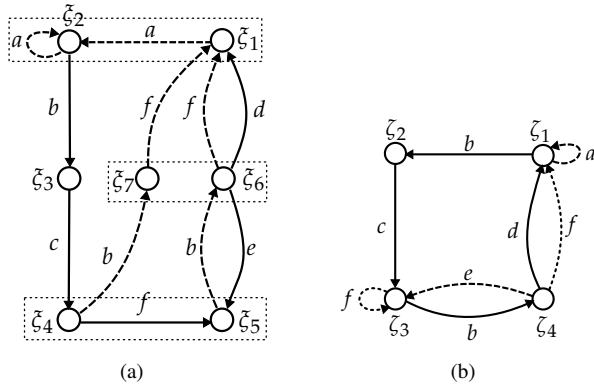


Fig. 4. Illustrating the generalized decomposition algorithm.

Example 3: For illustration purposes, consider $P = (X, W, \Delta, X)$ with $X = \{\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6, \xi_7\}$ and $W = \{a, b, c, d, e, f\}$, as depicted in Fig. 4(a). Fig. 4(b) shows a quotient state machine $T = (Z, W, \Lambda, Z)$ induced by the state space aggregation Q leading to $Z = \{\zeta_1, \zeta_2, \zeta_3, \zeta_4\}$ in accordance with $\zeta_1 = L_Z(\{\xi_1, \xi_2\})$, $\zeta_2 = L_Z(\xi_3)$, $\zeta_3 = L_Z(\{\xi_4, \xi_5\})$ and $\zeta_4 = L_Z(\{\xi_6, \xi_7\})$. Note that P is not chain decomposable due to the non-injectivity of functions $\hat{\rho}_a$ and $\hat{\rho}_f$, whereas T is. As indicated in Fig. 4(b), the external signal space W can be accordingly partitioned as $W = \Omega_1 \cup \Omega_2 \cup \Omega_3$, with $\Omega_1 = \{b, c, d\}$, $\Omega_2 = \{e, a\}$ and $\Omega_3 = \{f\}$. ■

Now introduce a state space decomposition for the quotient state machine $T = (Z, W, \Lambda, Z)$ as suggested by the algorithm given by (20) and (21). Consider first any two symbols $\omega' \neq \omega'' \in W$. Then, using the fact that $\mathcal{Q}^{-1}$ and χ' are both absolutely injective functions, from (29) it follows

$$\chi(\omega') \cap \chi(\omega'') \subseteq \mathcal{Q}^{-1} \circ (\chi'(\omega') \cap \chi'(\omega''))$$

that is

$$\chi(\omega') \cap \chi(\omega'') = \emptyset. \quad (31)$$

Similarly, consider a $w|_{[\tau, t]} \in W^{[\tau, t]}$ and pick up any two different $w'|_{[\tau, t]} = w''|_{[\tau, t]}$ from the corresponding signal space restriction $\mathcal{D}_A(w|_{[\tau, t]})$. Then, by using the fact that $\rho|_{\mathcal{D}_A(w|_{[\tau, t]})}$ is absolutely injective, from (30) it follows

$$\rho(w'|_{[\tau, t]}) \cap \rho(w''|_{[\tau, t]}) = \emptyset. \quad (32)$$

Hereof we conclude that as a consequence of our aggregation policy, absolutely injective mappings $\rho|_{\mathcal{D}_A(w|_{[\tau, t]})}$ and $\chi|_{\mathcal{D}_A(w|_{[\tau, t]})}$ are identified, which brings us to another main statement.

Theorem 2: Consider a state machine $P = (X, W, \Delta, X)$. A signal space decomposition (8) resulting from applying the algorithm (20) and (21) on the chain-decomposable quotient state machine $T = P/Q$, where the equivalence relation Q is defined according to Lemma 1, leads invariably to exact computations (24a) and (24b) in a decentralized setup.

IV. CONCLUSIONS

A general decentralized framework for set-valued state estimation and prediction for hybrid state machines has been discussed in this article. The outcome of the decentralized scheme is computed as the intersection of the outcomes provided by the individual distributed state machines. The latter are constructed by means of abstract “aggregation” maps that lead to a decomposition of the external signal space. In practice, such maps may refer to a set of sensors with different zooming and resolution capabilities. We show that, in the general case, decentralized schemes provide overapproximate estimate and prediction outcomes. However, using the concept of “non-deterministic chains”, we are able to recover the exact computation outcomes with the help of constructive decomposition algorithms that take into account the structural properties of transition relation of the original state machine. The algorithms are systematically extended for transition relations that account for non-injectivity in the state transition function. In particular, we show that exact decentralized computation holds for all systems that assume an I/S/- realization with a singleton output map. Due to the “smaller” external signal spaces and transition relations associated with the individual distributed state machines, significant reduction in the overall space/time computational complexity may be expected. In this sense, a thorough complexity analysis on a case study can be found in [3]. Moreover, major advantages in terms of robustness and reliability are gained due to the redundancy of computation units. Our decentralized framework can be applied in different control contexts, including data fusion, failure detection & diagnosis, etc. Optimal construction of signal space decomposition leading to a minimization of the space/time complexity represents a natural extension to the work in this article.

REFERENCES

- [1] S. Xu and R. Kumar, “Distributed state estimation in discrete event systems,” in *Proceedings of American Control Conference*, 2009.
- [2] J. Lunze and F. L. Lagarrigue, Eds., *Handbook of hybrid systems control: theory, tools, applications*. Cambridge University Press, 2010.
- [3] N. Bajcinca, Y. Kouhi, and V. Nenchev, “Decentralized set-valued state estimation based on non-deterministic chains,” *sub. in American Control Conference, ACC2012, Montreal*.
- [4] J. Willems, “Paradigms and puzzles in the theory of dynamical systems,” *IEEE Transactions on Automatic Control*, vol. 36, no. 2, pp. 259–294, 1991.
- [5] J. Raisch, T. Moor, N. Bajcinca, S. Geist, and V. Nenchev, “Distributed state estimation for hybrid and discrete event systems using l-complete approximations,” *WODES*, vol. 43, pp. 139–144, 2010.

Predictive control of an electro-hydraulic actuated wet clutch

A. E. Balau and C. Lazar

Abstract—During the last few years automotive actuators have become mechatronic systems in which mechanical components coexist with electronics and computing devices and because pressure control valves are used as actuators in many control applications for automotive systems, a proper dynamic model is necessary. In this paper, the concept of modeling an electro-hydraulic actuated clutch system in the automatic transmission is presented and two different control strategies are applied on the develop models. The simulation result are presented and discussed.

I. INTRODUCTION

In recent years, the use of control systems for automated clutch and transmission actuation has been constantly increasing, the trend towards higher levels of comfort and driving dynamics while at the same time minimizing fuel consumption representing a major challenge.

The basic function of any type of automotive transmission is to transfer the engine torque to the vehicle with the desired ratio smoothly and efficiently, and the most common control devices inside the transmission are clutches and actuators. Such clutches can be hydraulic actuated, motor driven or actuated using other means.

Hydraulic control valves are devices that use mechanical motion to control a source of fluid power and are used as actuators in many control applications for automotive systems. They vary in arrangement and complexity, depending upon their function. The many types of valves available are best classified according to their function. Three broad functional types can be distinguished: directional control valves, pressure control valves and flow control valves. Pressure control valves act to regulate pressure in a circuit and may be subdivided into pressure relief valves and pressure reducing valves. Pressure relief valves, which are normally closed, open up to establish a maximum pressure and bypass excess flow to maintain the set pressure. Pressure reducing valves, which are normally open, close to maintain a minimum pressure by restricting flow in the line. Because control valves are the mechanical (or electrical) to fluid interface in hydraulic systems, their performance is under scrutiny, especially when system difficulties occurs. Therefore knowledge of the performance characteristics of valve is essential.

During the last years, the automated actuated clutch systems and different valve types used as actuators have been actively researched and different models and control strategies have been developed: physics-based nonlinear model for an exhausting valve [1], nonlinear state-space model description

of the actuator that is derived based on physical principles and parameter identification [2], [3], [4], [5] nonlinear physical model for programmable valves [6], mathematical model obtained using identification methods for a valve actuation system of an electro-hydraulic engine [7], a model for an electro-hydraulic valve used as actuator for a wet clutch [8], dynamic modeling and control of electro-hydraulic wet clutches [9], PID control for a wet plate clutch actuated by a pressure reducing valve [10], predictive and piecewise LQ control of a dry clutch engagement [11], switched control of an electro-pneumatic clutch actuator [12], Model Predictive Control of a two stage actuation system using piezoelectric actuators for controllable industrial and automotive brakes and clutches [13].

The automatic control of the clutch engagement plays a crucial role in AMT vehicles, and in this paper we deal with the problem of modeling and controlling an electro-hydraulic actuated wet clutch. Two mathematical models of the actuator-clutch system are presented: an input-output model and a state-space model. Then, two different controllers are designed in order to control the clutch piston displacement: a generalized predictive controller (GPC) and a PID controller.

The rest of the paper is organized as follows. In Sec. II, the mathematical models of the electro-hydraulic actuated wet clutch are presented. Sec. III presents the two control strategies and the simulation results are discusses, and the concluding remarks are given in Sec. IV.

II. MODELING OF AN ELECTRO-HYDRAULIC ACTUATED WET CLUTCH AS A SUBSYSTEM OF AN AUTOMATED MANUAL TRANSMISSION

Control valves are the mechanical (or electrical) to fluid interface in hydraulic systems, and the knowledge of their performance characteristics is essential. Pressure control valves employ feedback and may be properly regarded as servo control loops. Because of that, a proper dynamic design is necessary to achieve stability. Starting from equations found in research work, where a single stage pressure reducing valve is modeled, in this paper, the concept of modeling an electro-hydraulic actuated wet clutch is presented.

A pump produces the line pressure P_s used as input for the electro-hydraulic actuator represented by a pressure reducing valve. This valve releases a pressure depending on the current i in the solenoid, which will have as consequence the magnetic force F_{mag} exerted on the valve plunger, which moves linearly within a bounded region under the effect of this force. Such a force is generated by a solenoid placed at

Department of Automatic Control and Applied Informatics,
Gheorghe Asachi Technical University of Iasi : { abalau,
clazar }@ac.tuiasi.ro

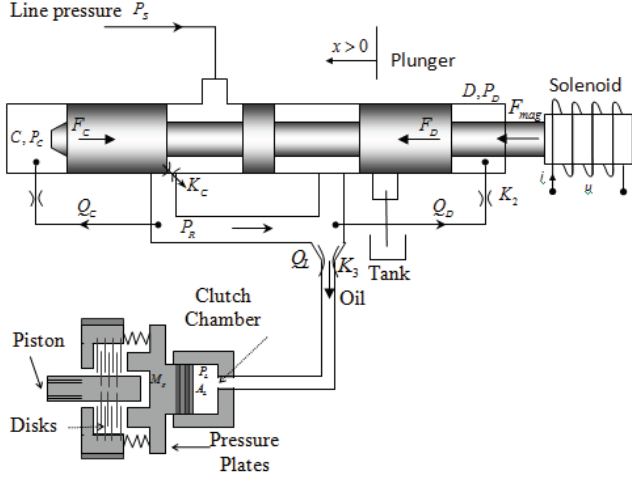


Fig. 1. Charging phase of the actuator-clutch system.

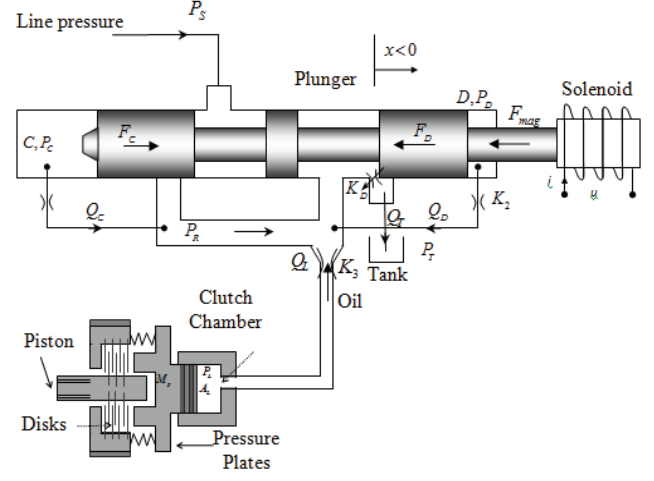


Fig. 2. Discharging phase of the actuator-clutch system.

one boundary of the region. The magnetic force is a function of the solenoid current and the displacement x , defined by:

$$F_{mag} = f(i, x) = \frac{k_a i^2}{2(k_b + x)^2}; \quad L \frac{di}{dt} + Ri = u, \quad (1)$$

where k_a and k_b are constants, L is the solenoid induction, R the resistance and u is the supply voltage.

The pressure to be controlled P_R is sensed on the spool end areas C and D and compared with the magnetic force which actuates on the plunger. The feedback force $F_{feed} = F_C - F_D$ is the difference between the force applied on the left sensed pressure chamber F_C , and the force applied on the right sensed pressure chamber F_D .

Depending on the valve plunger position, there are two phases of the actuator-clutch system: the charging phase, when the magnetic force is greater than the feedback force and the valve plunger is moved to the left, connecting the source with the hydraulic actuated clutch (Fig. 1), and the discharging phase, when the magnetic force is switched off or has a lower value than the feedback force so that the valve plunger is moved to the right, connecting the hydraulic actuated clutch to the tank (Fig. 2).

A. Input-output model

Using the magnetic force and the feedback force it results a force balance which describes the spool motion and the output pressure. This equation of force balance is the same for both positive and negative displacement of the spool:

$$F_{mag} - CP_C + DP_D = M_v s^2 X + K_e X, \quad (2)$$

where P_C represents the pressure in the left sensed chamber that acts on the (C) area, P_D represents the pressure in the right sensed chamber that acts on the (D) area, M_v is the spool mass, $K_e = 0.43w(P_{S0}P_{R0})$ represents the flow force spring rate, w represents the area gradient of the main orifice,

$X = X(s)$ is the Laplace transform of the spool displacement and s represents the Laplace operator.

The linearized continuity equation from [14] was used to describe the dynamics from the sensed pressure chambers:

$$Q_C = K_1 (P_R - P_C) = \frac{V_C}{\beta_e} s P_C - C_s X, \quad (3)$$

$$Q_D = K_2 (P_R - P_D) = \frac{V_D}{\beta_e} s P_D + D_s X, \quad (4)$$

where K_1, K_2 are the flow-pressure coefficients of restrictors, V_C, V_D are the sensing chamber volumes and β_e represents the effective bulk modulus.

Using the flow through the left and right sensed chambers, the flow through the main orifice (from the source to the hydraulic load) and the load flow, the linearized continuity equation at the chamber of the pressure being controlled is:

$$K_C (P_S - P_R) - Q_L - k_l P_R - K_1 (P_R - P_C) - K_2 (P_R - P_D) + K_q X = \frac{V_t}{\beta_e} s P_R, \quad (5)$$

where Q_L is the load flow, K_C is the flow-pressure coefficient of main orifice, K_q is the flow gain of main orifice, k_l is the leakage coefficient and V_t represents the total volume of the chamber where the pressure is being controlled.

These equations define the valve dynamics and combining them into a more useful form, solving (3) and (4) w.r.t. P_C and P_D and substituting into (5) yields after some manipulation ($V_C \ll V_t$ and $V_D \ll V_t$):

$$\begin{aligned}
& (K_C P_S - Q_L) \left(\frac{s}{\omega_1} + 1 \right) \left(\frac{s}{\omega_2} + 1 \right) + K_q X \left[1 + \right. \\
& \quad \left. + \left(\frac{1}{\omega_1} + \frac{1}{\omega_2} + \frac{C}{K_q} - \frac{D}{K_q} \right) s + \right. \\
& \quad \left. + \left(\frac{1}{\omega_1 \omega_2} + \frac{C}{K_q \omega_2} - \frac{D}{K_q \omega_1} \right) s^2 \right] = \\
& \quad = P_R K_{ce} \left(\frac{s}{\omega_1} + 1 \right) \left(\frac{s}{\omega_2} + 1 \right) \left(\frac{s}{\omega_3} + 1 \right), \tag{6}
\end{aligned}$$

where $\omega_1 = \frac{\beta_e K_1}{V_C}$ and $\omega_2 = \frac{\beta_e K_2}{V_D}$, are the break frequency of the left and right sensed chambers, $\omega_3 = \frac{\beta_e K_{ce}}{V_t}$ is the break frequency of the main volume and $K_{ce} = K_C + k_l$ represents the equivalent flow-pressure coefficient.

A negative displacement of the pressure reducing valve spool allows connection between the hydraulic load and the tank, while the channel that connects the source with the hydraulic load is kept closed.

The linearized continuity equations at the sensed pressure chambers for the discharging phase of the valve, illustrated in Fig. 2, are:

$$-Q_C = K_1 (P_C - P_R) = -\frac{V_C}{\beta_e} s P_C + C s X, \tag{7}$$

$$-Q_D = K_2 (P_D - P_R) = -\frac{V_D}{\beta_e} s P_D - D s X. \tag{8}$$

Using the flow through the left and right sensed chambers, the flow through the main orifice (from the hydraulic load to the tank) and the load flow, the linearized continuity equation obtained for the chamber of the pressure being controlled is:

$$\begin{aligned}
& Q_L + K_1 (P_C - P_R) + K_2 (P_D - P_R) - \\
& - K_D (P_R - P_T) - k_l P_R + K_q X = \frac{V_t}{\beta_e} s P_R, \tag{9}
\end{aligned}$$

where K_D is the flow-pressure coefficient of main orifice and P_T represents the tank pressure.

Combining these equations into a more useful form, solving (7) and (8) for P_C and P_D and substituting into (9) yields after some manipulation ($V_C \ll V_t$ and $V_D \ll V_t$ like for the charging phase model and considering $K_D = K_C$):

$$\begin{aligned}
& (K_D P_T + Q_L) \left(\frac{s}{\omega_1} + 1 \right) \left(\frac{s}{\omega_2} + 1 \right) + \\
& + K_q X \left[1 + \left(\frac{1}{\omega_1} + \frac{1}{\omega_2} + \frac{C}{K_q} - \frac{D}{K_q} \right) s + \right. \\
& \left. + \left(\frac{1}{\omega_1 \omega_2} + \frac{C}{K_q \omega_2} - \frac{D}{K_q \omega_1} \right) s^2 \right] = \\
& = P_R K_{ce} \left(\frac{s}{\omega_1} + 1 \right) \left(\frac{s}{\omega_2} + 1 \right) \left(\frac{s}{\omega_3} + 1 \right). \tag{10}
\end{aligned}$$

Considering the resulting force between the magnetic and the feedback force:

$$F_1 = F_{mag} - C P_C + D P_D, \tag{11}$$

solving P_C and P_D from the linearized continuity equations (3), (4) and substituting in the force balance equation (2), the following equation was obtained:

$$\begin{aligned}
& F_{mag} - C \frac{K_1 P_R + C s X}{\left(\frac{s}{\omega_1} + 1 \right) K_1} + D \frac{K_2 P_R - D s X}{\left(\frac{s}{\omega_2} + 1 \right) K_2} = \\
& = M_v s^2 X + K_e X. \tag{12}
\end{aligned}$$

After some manipulations, where it was considered that $\omega_m = \sqrt{\frac{K_e}{M_v}}$, representing the mechanical natural frequency, and substituting (11) into (2) yields:

$$\begin{aligned}
& F_1 - \left(\frac{\frac{C^2}{K_1} s}{\left(\frac{s}{\omega_1} + 1 \right)} + \frac{\frac{D^2}{K_2} s}{\left(\frac{s}{\omega_2} + 1 \right)} \right) X = \\
& = K_e \left(\frac{s^2}{\omega_m^2} + 1 \right) X, \tag{13}
\end{aligned}$$

where

$$F_1 = F_{mag} - \left(\frac{C}{\frac{s}{\omega_1} + 1} + \frac{D}{\frac{s}{\omega_2} + 1} \right) P_R. \tag{14}$$

The wet clutch is a chamber with a piston as represented in (Fig. 1). In the charging phase when the valve plunger is moved to the left and the displacement is considered positive, the oil flows from the source through the valve to the clutch and the piston in the clutch moves towards the clutch plates compressing them. In the discharging phase, when the valve plunger is moved to the right and the displacement is negative, the clutch piston moves to the left and the oil flows from the clutch chamber through the valve to the tank.

For the clutch model, the first equation arises by applying Newton's second law to the forces on the piston, resulting:

$$A_L P_L = M_p s^2 x_p + K x_p, \tag{15}$$

where A_L is the area of piston, P_L the pressure from the piston chamber, x_p the piston displacement, M_p the total mass of the piston and K is the load spring gradient of the piston.

Applying the continuity equation to the piston chamber yields:

$$Q_L = K_3 (P_R - P_L) = \frac{V_L}{\beta_e} s P_L + A_L s x_p, \tag{16}$$

for the charging phase of the system, and:

$$-Q_L = K_3 (P_L - P_R) = -\frac{V_L}{\beta_e} s P_L - A_L s X, \tag{17}$$

for the discharging phase of the system, where K_3 is the flow-pressure coefficient of the pipe from valve actuator to the clutch and V_L is the piston chamber volume.

Equations (1), (2), (3), (4), (5) and (6) for the charging phase of the valve, and equations (1), (2), (7), (8), (9) and

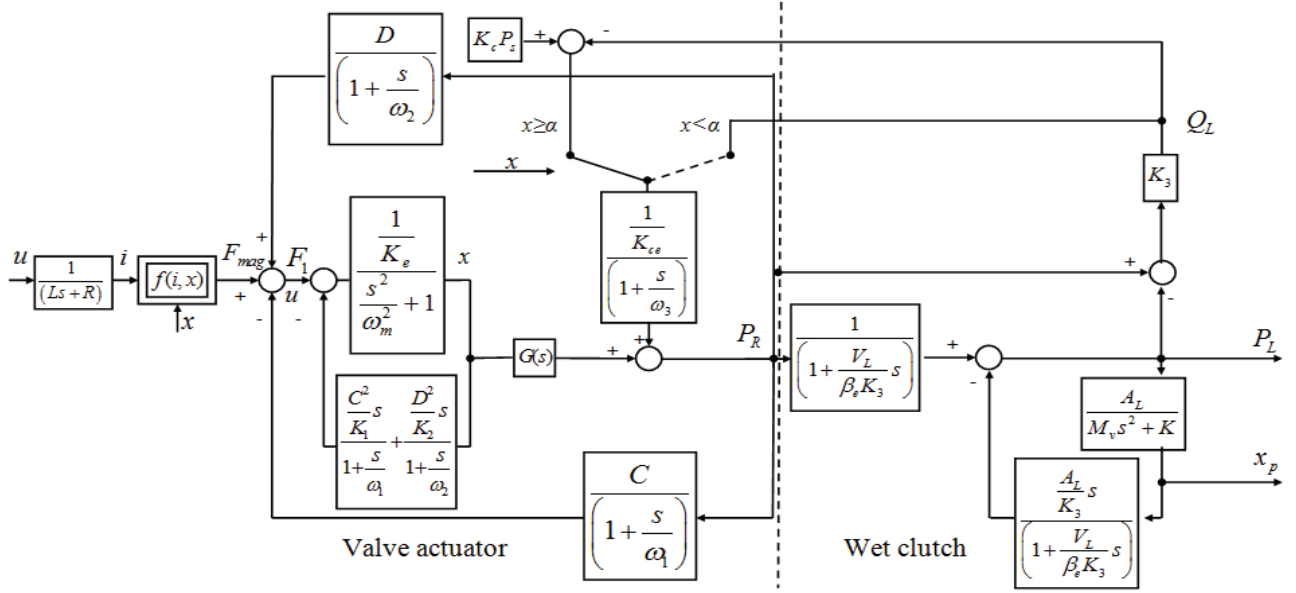


Fig. 3. Transfer function block diagram of the actuator-clutch system.

(10) for the discharging phase of the valve, together with (15) and (16), for the charging phase of the clutch, and (15) and (17) for the discharging phase of the clutch, define the electro-hydraulic actuated wet clutch dynamics and can be used to construct the transfer function block diagram represented in Fig. 3.

The representation was obtained after some manipulations, considering (13) and (14) and also, the following notation was made:

$$G(s) = \frac{\frac{K_q}{K_{ce}} \left[1 + \left(\frac{1}{\omega_1} + \frac{1}{\omega_2} + \frac{C}{K_q} - \frac{D}{K_q} \right) s \right]}{\left(1 + \frac{s}{\omega_1} \right) \left(1 + \frac{s}{\omega_2} \right) \left(1 + \frac{s}{\omega_3} \right)} + \frac{\frac{K_q}{K_{ce}} \left(\frac{1}{\omega_1 \omega_2} + \frac{C}{K_q \omega_2} - \frac{D}{K_q \omega_1} \right) s^2}{\left(1 + \frac{s}{\omega_1} \right) \left(1 + \frac{s}{\omega_2} \right) \left(1 + \frac{s}{\omega_3} \right)}. \quad (18)$$

It can be seen that a switch block was used in order to commute between the two phases that describe the functionality of the electro-hydraulic actuated wet clutch, the charging and the discharging phase. The sign of the displacement of the plunger was used as the switch parameter, with the threshold α , thus selecting different perturbations for positive or negative displacements.

B. State-space model

Combining the equations (1), (2), (3), (4), (5), (15) and (16) that describe the dynamics of the system in the charging phase, and equations (1), (2), (7), (8), (9), (15) and (17) that describe the dynamics of the system in the discharging phase, the state-space model of the electro-hydraulic actuated clutch is design according to:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \quad (19)$$

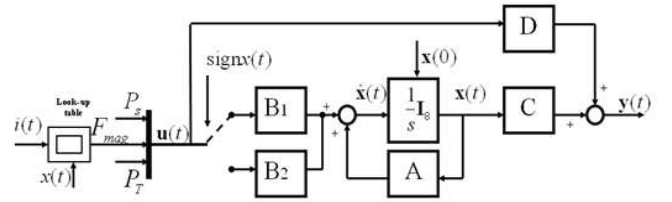


Fig. 4. State-space block diagram of the actuator-clutch system.

The state vector is represented by $x(t)$, $u(t)$ is the input vector, and $y(t)$ is considered to be the output of the system:

$$\begin{aligned} u(t) &= \begin{bmatrix} P_S(t) & P_T(t) & F_{mag}(t) \end{bmatrix}^T, \\ x(t) &= \begin{bmatrix} x_1(t)^T & x_2(t)^T \end{bmatrix}^T, \\ x_1(t) &= \begin{bmatrix} v(t) & x(t) & v_p(t) & x_p(t) \end{bmatrix}^T, \\ x_2(t) &= \begin{bmatrix} P_C(t) & P_D(t) & P_R(t) & P_L(t) \end{bmatrix}^T, \\ y(t) &= \begin{bmatrix} x(t) & x_p(t) & P_R(t) & P_L(t) \end{bmatrix}^T. \end{aligned} \quad (20)$$

The matrix A is the same for both charging and discharging phases of the actuator-clutch, while different values of the B matrix are used: B_1 for the charging phase of the system and B_2 for the discharging phase of the electro-hydraulic actuated clutch, and the matrices can be found in [15].

Starting from the equations that illustrates the mathematical model, a block diagram for the actuator-clutch system was created and represented in Fig. 4. It can be seen that a switch block was used, relative to the sign of the plunger displacement, in order to commute between the two phases that describe the functionality of the actuator, B_1 for the charging phase of the system and B_2 for the discharging phase.

III. CONTROL STRATEGY

Starting from the electro-hydraulic actuated wet clutch system model presented in the previous section, an analytical designed PID controller, with the help of the pole placement method and a GPC controller are presented and the results are compared and discussed.

The controllers have as input a reference value of the clutch piston displacement and, in order to apply a control strategy, the input of the system was considered to be the v voltage, while the output of the system is the clutch piston displacement.

A. Generalized predictive control

Predictive control techniques are of a particular interest from the point of view of both broad applicability and implementation simplicity, being applied on large scale in industry processes, having good performances and being robust at the same time.

Consider the plant described by the CARIMA (Controlled AutoRegressive Integrated Moving Average) model [16]:

$$A(z^{-1})y(k) = B(z^{-1})u(k-1) + \frac{e(k)C(z^{-1})}{D(z^{-1})}, \quad (21)$$

where $e(k)$ is white noise with zero mean value, u is the input voltage, $y(k)$ is the clutch displacement, $A(z^{-1})$ and $B(z^{-1})$ are the system polynomials with the degrees n_A and n_B , and $C(z^{-1}) = 1$ and $D(z^{-1}) = 1 - z^{-1}$ are the disturbances polynomials.

The electro-hydraulic actuated wet clutch system was identified with an ARX equivalent one employing the simulation model, utilizing as input a PRBS (PseudoRandom Binary Sequence) signal:

$$\begin{aligned} A(z^{-1}) &= 1 - 1.781z^{-1} + 0.8039z^{-2}, \\ B(z^{-1}) &= 0.00003312z^{-1} + 0.0001122z^{-2}, \end{aligned} \quad (22)$$

and the disturbances polynomials were considered to be $C(z^{-1}) = 1$ and $D(z^{-1}) = 1 - z^{-1}$ for obtaining a zero steady-state error.

The prediction model is given by

$$\begin{aligned} \hat{y}(k+j|k) &= G_j(z^{-1})D(z^{-1})z^{-1}u(k+j) + \\ &+ \frac{H_j(z^{-1})D(z^{-1})}{C(z^{-1})}u(k-1) + \frac{F_j(z^{-1})}{C(z^{-1})}y(k), \end{aligned} \quad (23)$$

with $j = \overline{hi, hp}$, where hi is the minimum prediction horizon and hp is the prediction horizon. $u(k+j-1|k)$, $j = \overline{1, hc}$ is the future control, computed at time k and $\hat{y}(k+j|k)$ are the predicted values of the output, hc being the control horizon.

The two Diophantine equations presented in [16] are used to determine the polynomials $F_j(z^{-1})$, $G_j(z^{-1})$ and $H_j(z^{-1})$.

Considering as inputs $D(z^{-1})u(k)$ and collecting the j -step predictors in a matrix notation, the prediction model can be written as

$$\hat{y} = Gu_d + \hat{y}_0, \quad (24)$$

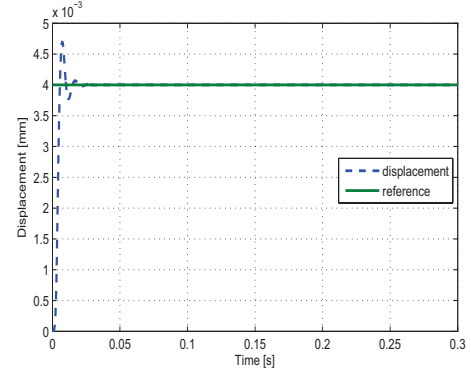


Fig. 5. GPC results.

where $\hat{y}$ represents the free response and matrix G is given in [16].

The objective function is based on the minimization of the tracking error and on the minimization of the controller output, the control weighting factor λ being introduced in order to make a trade-off between these objectives

$$J = (Gu_d + \hat{y}_0 - w)^T (Gu_d + \hat{y}_0 - w) + \lambda u_d^T u_d, \quad (25)$$

subject to $D(z^{-1})u(k+i) = 0$ for $i \in [hc, hp-1]$, where w is the reference trajectory vector with the components $w(k+j|k)$, $j = \overline{hi, hp}$. Minimizing the objective function ($\partial J / \partial u_d = 0$), the optimal control sequence yields as

$$u_d^* = (G^T G + \lambda I_{hc})^{-1} G^T [w - \hat{y}_0]. \quad (26)$$

Using the receding horizon principle and considering that γ_j , $j = \overline{hi, hp}$ are the elements of the first row of the matrix $(G^T G + \lambda I_{hc})^{-1} G^T$, the following control algorithm results:

$$D(z^{-1})u(k) = \sum_{j=hi}^{hp} \gamma_j [w(k+j|k) - \hat{y}_0(k+j|k)]. \quad (27)$$

The determined controller of the electro-hydraulic actuated wet clutch system was implemented in Matlab/Simulink. A step signal was applied as the reference for the clutch piston displacement and it was desired that the system tracks the reference signal as fast as possible, Fig. 5 showing the controlled output and the reference signal.

It can be seen that when using this predictive control strategy, the system reaches the reference value very fast, the rising time is small, but there is an overshoot of about 17%.

B. PID control

Having the discrete model of the plant given by:

$$G_f(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{0.00003312z^{-1} + 0.0001122z^{-2}}{1 - 1.781z^{-1} + 0.8039z^{-2}}, \quad (28)$$

and taking into account the performances imposed to the automated system ($\zeta = 0.707$ and $T = 0.08s$), we obtain the coefficients $\alpha_1 = -1.9001$, $\alpha_2 = 0.9048$, $\alpha_3 = 0.7022$ and the characteristic polynomial in the form of:

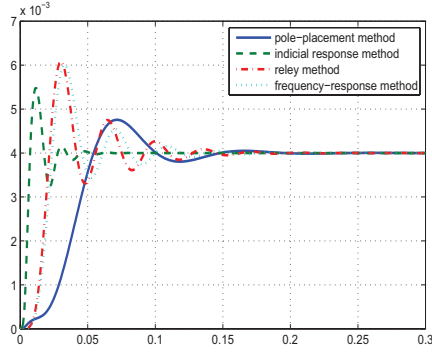


Fig. 6. PID controller results.

$$P_{cd}(z) = (z^2 + \alpha_1 z + \alpha_2)(z - \alpha_3)(z - \alpha_4). \quad (29)$$

Next step is to build the characteristic polynomial of the closed loop system:

$$P_{c0}(z^{-1}) = P(z^{-1})A(z^{-1}) + Q(z^{-1})B(z^{-1}). \quad (30)$$

Because $m = 2$, the system is undetermined and we select the PID controller with filtering of the derivative component:

$$G_R(z^{-1}) = \frac{Q(z^{-1})}{P(z^{-1})} = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{(1 - z^{-1})(1 - \alpha_d z^{-1})}. \quad (31)$$

The tuning parameters of the discrete PID controller are chosen by solving $P_{c0}(z^{-1}) = P_{cd}(z^{-1})$. This yields:

$$\begin{bmatrix} 1 & b_1 & 0 & 0 \\ a_1 - 1 & b_2 & b_1 & 0 \\ a_2 - a_1 & 0 & b_2 & b_1 \\ -a_2 & 0 & 0 & b_2 \end{bmatrix} \begin{bmatrix} \alpha_d \\ q_0 \\ q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} -a_1 + 1 - 3.3044 \\ 4.0661 - a_2 + a_1 \\ -2.2074 + a_2 \\ 0.4461 \end{bmatrix}, \quad (32)$$

and the following tuning parameters for the discrete PID controller are obtained:

$$\begin{bmatrix} \alpha_d \\ q_0 \\ q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} -0.5295 \\ 184.8801 \\ -364.0633 \\ 182.0876 \end{bmatrix}. \quad (33)$$

The PID controller was then implemented in Matlab/Simulink for controlling the electro-hydraulic actuated wet clutch system. Also, different controllers were experimentally tuned using the relay tuning method, and the Ziegler-Nichols method based on indicial-response and on frequency-response. In Fig. 6 the responses obtained with the four methods are presented.

It can be seen that when using the pole placement method to design the PID controller the set-point response has the smallest overshoot but with a downside concerning the rising time. When using the relay method or the frequency-response method, the system reaches the reference value almost in the same time like the pole placement controller, the rising time is smaller but the value of the overshoot is more than double. A faster response is obtain when using the indicial-response

method, but with a higher value of the overshoot that in the case of the pole-placement controller.

IV. CONCLUSIONS

In this paper two models for an electro-hydraulic actuated clutch system used in the automotive control systems for automatic transmission are presented: a linearized input-output model and a state-space model. After that, a GPC and a PID controller are designed in order to control the output of the electro-hydraulic actuated clutch system: the clutch piston displacement. Comparing the results of the two control strategies, it can be clearly seen that the GPC has better result, concerning both the overshoot and the response time of the system.

V. ACKNOWLEDGMENTS

The work was supported by National Center for Programs Management from Romania under the research grant SICONA - 12100/2008.

REFERENCES

- [1] J. Ma, G. Zhu, A. Haartsig, and H. Schock, "Model-based predictive control of an electro-pneumatic exhaust valve for internal combustion engines," in *American Control Conference*, Seattle, Washington, USA, 2008.
- [2] J. MA, G. Zhu, A. Haartsig, and H. Schock, "Modeling and control of electromechanical valve actuator," *Society of Automotive Engineers*, 2002.
- [3] K. S. Peterson, A. G. Stefanopoulou, Y. Wang, and T. Megli, "Virtual lash adjuster for an electromechanical valve actuator through iterative learning control," in *International Mechanical Engineering Congress*, Washington, D.C., USA, 2003.
- [4] S. D. Gennaro, B. C. Toledo, and M. D. Benedetto, "Non-linear control of electromagnetic valves for camless engines," vol. 80(11), pp. 1796–1813, 2007.
- [5] H. Nemeth, "Nonlinear modelling and control for a mechatronic protection valve," Ph.D. dissertation, Budapest, Hungary, 2004.
- [6] S. Liu and B. Yao, "Coordinative control of energy saving programmable valves," *IEEE Transactions on Control Systems Technology*, vol. 16, pp. 34–45, 2008.
- [7] H.-H. Liao, M. J. Roelle, and J. Gerdes, "Repetitive control of an electro-hydraulic engine valve actuation system," in *American Control Conference*, Seattle, Washington, USA, 2008.
- [8] R. Morselli and R. Zanasi, "Automotive control systems using power oriented graphs," in *The 32nd Annual Conference on Industrial Electronics (IECON)*, Paris, France, 2006.
- [9] R. Morselli, R. Zanasi, R. Cirrone, R. Sereni, R. Bedogni, and E. Sedoni, "Dynamic modeling and control of electro-hydraulic wet clutches," in *IEEE Intelligent Transportation Systems*, Shanghai, China, 2003.
- [10] M. J. W. H. Edelaar, "Model and control of a wet plate clutch," Ph.D. dissertation, Eindhoven University of Technology, The Netherlands, 1997.
- [11] A. C. Van Der Heijden, A. F. A. Serrarens, M. K. Camlibel, and H. Nijmeijer, "Hybrid optimal control of dry clutch engagement," *International Journal of Control*, vol. 80, pp. 1717–1728, 2007.
- [12] H. Langjord, T. A. Johansen, and J. P. Hespanha, "Switched control of an electropneumatic clutch actuator using on/off valves," in *27th American Control Conference*, Seattle, USA, 2008.
- [13] V. A. Neelakantan, "Model predictive control of a two stage actuation system using piezoelectric actuators for controllable industrial and automotive brakes and clutches," *Journal of Intelligent Material Systems and Structures*, vol. 19, pp. 845–857, 2008.
- [14] H. E. Merritt, *Hydraulic control systems*. USA: John Wiley & Sons, 1967.
- [15] A. E. Balau and C. Lazar, "State-space model of an electro-hydraulic actuated wet clutch," in *IFAC Symposium Advances in Automotive Control*, Munchen, Germany, 2010.
- [16] E. F. Camacho and C. Bordons, *Model Predictive Control*. Springer Verlag, 2004.

Identification and Sensitivity Analysis of a Trickling Biofilter Viewed as a Distributed Parameters System

Marian Barbu, Viorel Mînză, Daniela Carp, Emil Ceangă

Abstract— The paper aims to establish the frequency models and sensitivity analysis of a biological filter in an intensive recirculating aquaculture system when changing the functioning regime of the plant. The starting point is an analytical model of a trickling biofilter, a distributed parameter model identified from experimental data. The software implementation of this model is used as a process emulator, various functioning regimes of the trickling biofilter being simulated. Based on the simulated data sets and using adaptive filters the frequency models for each input - output channel have been identified. The sensitivity analysis of the trickling biofilter upon changing the functioning regime has led to important conclusions regarding the control solutions to be adopted for each input – output channel. Finally, based on the analysis conducted, a strategy for the automatic control of the intensive recirculating aquaculture system is proposed.

I. INTRODUCTION

The intensive recirculating aquaculture systems make it possible to obtain fairly high productivity, given that the aquaculture tanks are densely populated. This high fish density in the aquaculture tanks, together with an intensive feeding process, lead to a high concentration of ammonium in the wastewater. Under these conditions, the wastewater treatment systems within the intensive recirculating aquaculture systems should be highly efficient and at the same time the efficiency should be constant in time. This makes necessary the automatic control of the wastewater treatment system.

Since the main dissolved component of the wastewater is ammonium, the recirculated water quality depends above all on the performance of the nitrification process [1]. Therefore, there is a justified interest for the nitrification modelling and simulation in the wastewater treatment plants [2], [3]. An approach that offers significant solutions to the modelling problem of the nitrification process consists in describing the nitrification through a modified version of the model ASM3_2N [4], including the situation when the nitrification is treated as a controlled process [5].

A constructive version of the biological filter, essential in recirculating aquaculture systems, is the trickling filter. Its modelling is very difficult because the basic nitrification processes occur at a biofilm scale [6], [7]. The model of this

filter is given by a set of partial derivatives equations with respect to two spatial coordinates [7]. In order to implement the automatic control of the trickling biofilter the use of such a model is inappropriate. In literature there are few papers that address the problem of proposing mathematical models for trickling biofilters control and studying the dynamic properties of these models [8], [9].

This paper aims to establish frequency models and sensitivity analysis of a biological filter in an intensive recirculating aquaculture system when changing the plant functioning regime. The paper is structured as follows: section 2 briefly presents the experimental recirculating aquaculture system; section 3 provides the description of the trickling biofilter modelling; in section 4 the properties of the trickling biofilter treated as a dynamic system are analyzed; in section 5 the main approaches regarding the design of the trickling biofilters control systems are reviewed and the last section is dedicated to the conclusions.

II. INTENSIVE RECIRCULATING AQUACULTURE PILOT PLANT

In this study an intensive recirculating aquaculture pilot plant is used (Figures 1a and 1b). The pilot plant consists of two components: the first component is for the growth of various fish species, starting with the less demanding ones (e.g. carp) and ending with "difficult" species such as trout, sturgeons (beluga, sevruga etc.); the second component is for the wastewater treatment in order to ensure optimal conditions for the growth of fish species in the recirculating system [10].

The system contains four octagonal tanks of 1 m³ capacity each, in which the intensive growth of the fish takes place. The tank aeration is done as follows: through the ejection effect the air from the atmosphere in the influx is transferred with special valves which bring water into the tanks. The wastewater from the four aquaculture tanks is collected through natural flow in the mechanical treatment system (Drum filter). Here the separation and storage of the solid substances are achieved. Together with the solids removal, a part of the water is also removed, this being used to clean the filter. The compensation of this quantity of water is achieved by approximation and it is done manually from the public water network. In the next phase the water flows into a tank with a buffer accumulator role. From this tank, the water is transferred using a pump to two filters (with sand and coal respectively) and it is brought to the top of the biological filter of trickling type.

This work was supported by CNCS-UEFISCDI, project 9 PN II-RU 79/2010. Daniela Carp acknowledges the support of POSDRU Project 107/1.5/S/76822, Acronym TOP ACADEMIC.

Marian Barbu, Viorel Mînză, Daniela Carp and Emil Ceangă are with "Dunărea de Jos" University of Galați, Domnească 47, 800008, Galați, România. E-mail: Marian.Barbu@ugal.ro.



a)



b)

Fig. 1. The recirculating aquaculture system

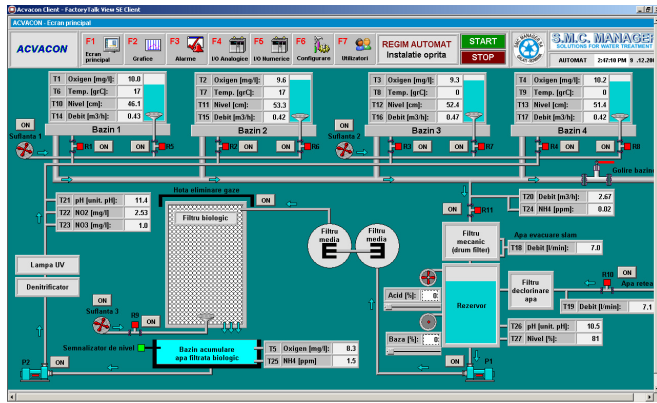


Fig. 2. The general synoptic scheme of the human-machine interface.

The trickling biofilter plays the major role of bringing the ammonium concentration in the recirculating aquaculture system to the characteristics imposed for a normal fish growth. It performs, under aerobic conditions, the transformation of the ammonium in nitrites and further in nitrates. In the tank below the biofilter, the water is collected at atmospheric pressure. From this point, the water is circulated through the chemical (for denitrification) and UV filters, being transferred under pressure in the aquaculture tanks. A constant level of water is maintained in these tanks through an overflow system.

The recirculating intensive aquaculture plant is equipped with a monitoring and control system that contains an HMI presented in Figure 2. For controlling the dissolved oxygen concentration in the aquaculture tanks and in the trickling biofilter ON/OFF controllers are used. The information regarding the dissolved oxygen concentrations are given by dissolved oxygen transducers from the aquaculture tanks and from the tank below the biofilter. The aeration is made using air generators.

III. MODELLING OF THE TRICKLING BIOFILTER

A. Analytical Modeling of the Trickling Biofilter

A trickling-type biofilter consists of many vertically distributed solid bodies. The trickling biofilter is fed at the top with the water to be treated through the nitrification process. Each body in the composition of the filter provides a large contact surface with water so that on these bodies biofilm is formed at a microscopic scale which enables the nitrification process. In the biofilter model two spatial coordinates are involved: a space coordinate relative to the biofilter height, z , related to the route of the processed water, and a space coordinate relative to the biofilm thickness, ξ , related to the biofilm processes. Obviously, the scales of the two space coordinates differ by several orders of magnitude. The plastic material on which microorganisms get attached, thus forming the biofilm, are not flooded but their surface is moist and aerated. It results that the biofilm composition may be considered to have three compartments to be modelled: a biofilm compartment, a bulk liquid compartment and a gaseous compartment. We will consider that the substance flow from gas to the biofilm is null and only the biofilm and the bulk liquid zones will be modelled in terms of transfer of components from the effluent. The gaseous zone will contribute only to the biofilm aeration.

Let us consider the fundamental equations of the concentration of one component (ammonium, nitrate etc.), in the biofilm and the bulk liquid. The model of biofilm concentration is:

$$\frac{\partial c}{\partial t} = \frac{\partial^2 c}{\partial \xi^2} - r(c) \quad (1)$$

where: c is the concentration of the considered component, ξ is the space coordinate relative to biofilm thickness, and $r(c)$ is the conversion rate of component c .

The boundary conditions with respect to equation (1) are:

$$\left(\frac{\partial c}{\partial \xi} \right)_{\xi=0} = 0; \quad (c)_{\xi=1} = c^b \quad (2)$$

where c^b is the concentration in the bulk liquid.

The concentration model in bulk liquid is:

$$v \frac{\partial c_i^b}{\partial t} = q \frac{\partial c_i^b}{\partial z} + a J_{f,i} + a_g J_{g,i}, \quad i = 1, 2, \dots, n \quad (3)$$

with:

$$v = V / (A_r h); \quad q = Q / A_r; \quad a = A / (A_r h); \quad a_g = A_g / (a_r h) \quad (4)$$

where c_i^b is the concentration of component i in bulk liquid (n components are considered), $J_{f,i}$ is the substance flow from biofilm to bulk liquid, $J_{g,i}$ – substance flow from gas to biofilm, z – space coordinate along the biofilm, A – total area of the biofilm, V – total volume of liquid, A_g – total area of gas-liquid interface, A_r – biofilter cross-section area, h – biofilter height, Q – liquid flow crossing the biofilm.

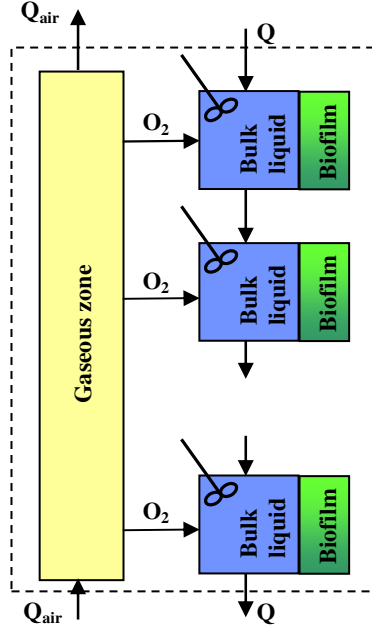


Fig. 3. Trickling biofilter structure

Further it is admitted that equation (3) is analyzed for a certain component, the same as for equation (1), and index i is no longer written. In equation (3) the spatial coordinate z is discretized into N finite zones. This corresponds to the approximation of the distributed system model in relation with z with a number of N concentrated subsystems. The concentrated subsystems are series connected as in Figure 3 (the gaseous zone is considered to be common). In each concentrated subsystem in Figure 3, the mass balance equation for the considered component can be written in the non-dimensional form:

$$\tau \frac{dc^b}{dt} = c_m^b - c^b - \gamma \left[\frac{\partial c}{\partial \xi} \right]_{\xi=1} \quad (5)$$

where τ and γ parameters are: $\tau = \frac{V}{Q} \lambda$, $\gamma = \frac{AD}{QL}$.

The general model of the biofilter is considered to consists of N equations of the type (5), for each finite element obtained from the discretization of the space coordinate z , and N equations of the type (1), each having to provide the factor $[\partial c / \partial \xi]_{\xi=1}$, which comes into eq. (5) of each zone defined on the biofilter height.

B. Simulation and Experimental Identification of the Trickling Biofilter

It was further considered a trickling biofilter structure of the type shown in Figure 3, the filter being divided into $N = 5$ zones. The analytical model of this trickling biofilter was developed and simulated in the AQUASIM software [11]. It was assumed that the main component of wastewater is ammonium, the organic substrate being considered negligible. The conversion rates of the components are taken from the mathematical models used in wastewater treatment systems with suspended biomass: ASM1 (Activated Sludge Model nr. 1) [12].

Using the routines provided by AQUASIM, the process was simulated and a sensitivity analysis of the analytical model with respect to its parameters has been done. This analysis showed that the main parameters that influence the model are: maximum specific growth rate for autotrophic biomass (μ_A), density of autotrophic biomass in biofilm (ρ_X) and yield for autotrophic biomass (Y_A). Finally, using AQUASIM identification routines and data taken from the HMI interface of the intensive aquaculture pilot plant, the analytic model with distributed parameters of the trickling biofilter has been identified. The values of the parameters considered in the identification are: $\mu_A = 1.14$, $\rho_X = 18800$ and $Y_A = 0.257$, and the identification results are illustrated in Figure 4 [13].

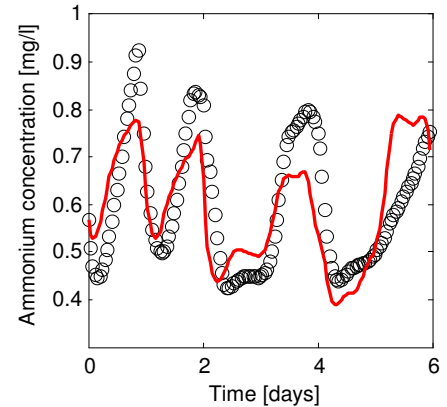


Fig. 4. Ammonium concentration in effluent with ASM1 based model (black circles – experimental data, red line – identified model)

C. Modeling of the Trickling Biofilter Using an Adaptive Filter

Although the trickling biofilter is a nonlinear system with distributed parameters, for control purpose it is sufficient to know its mathematical model linearized around the current functioning point. Obviously changing the functioning regime of the biofilter, which involves changing the functioning point, it is necessary to obtain the updated linear model. In these circumstances, the trickling biofilter can be treated as a variant dynamic system with distributed parameters. A powerful tool for identifying these systems is the adaptive filter.

Let $h_a(t)$ and $h_o(t)$ be the biofilter impulse response for

channels $NH_{4,in}(t) \rightarrow NH_{4,out}(t)$ and $Q_{in}(t) \rightarrow NH_{4,out}(t)$, where t is the discrete time ($t = 0, 1, 2, \dots$). Denoting by T_s the sampling period, the samples of the impulse responses $h_a(t)$ and $h_o(t)$ are:

$$h_a(0.T_s) \equiv h_{a,0}; h_a(1.T_s) \equiv h_{a,1};$$

$$h_a(2.T_s) \equiv h_{a,2}; \dots; h_a((n-1).T_s) \equiv h_{a,n-1} \quad (6)$$

$$h_o(0.T_s) \equiv h_{o,0}; h_o(1.T_s) \equiv h_{o,1};$$

$$h_o(2.T_s) \equiv h_{o,2}; \dots; h_o((n-1).T_s) \equiv h_{o,n-1} \quad (7)$$

As the process is considered variant, the impulse responses at moment $t=kT_s$ shall be denoted $h_{ai}[k]$ and $h_{oi}[k]$, $i=1, n-1$, and the discrete values vectors of the impulse responses are:

$$\mathbf{h}_a[k] = [h_{a,0}[k] \ h_{a,1}[k] \ \dots \ h_{a,n-1}[k]]^T \quad (8)$$

$$\mathbf{h}_o[k] = [h_{o,0}[k] \ h_{o,1}[k] \ \dots \ h_{o,n-1}[k]]^T \quad (9)$$

Convolution-type model of the process can be written as:

$$NH_{4,out}[k] = \sum_{i=0}^{n-1} h_{a,i-1}[k].NH_{4,in}[k-i] + \sum_{i=0}^{n-1} h_{o,i-1}[k].Q_{in}[k-i] \quad (10)$$

Defining the vectors:

$$\mathbf{x}_{NH_{4,in}} = [NH_{4,in}[k] \ NH_{4,in}[k-1] \ \dots \ NH_{4,in}[k-n+1]]^T \quad (11)$$

$$\mathbf{x}_{Q_{in}} = [Q_{in}[k] \ Q_{in}[k-1] \ Q_{in}[k-2] \ \dots \ Q_{in}[k-n+1]]^T \quad (12)$$

$$\mathbf{x}[k] = [\mathbf{x}_{NH_{4,in}}^T[k] \ \mathbf{x}_{Q_{in}}^T[k]]^T \quad (13)$$

$$\mathbf{h}[k] = [\mathbf{h}_a^T[k] \ \mathbf{h}_o^T[k]]^T \quad (14)$$

the process model can be written as

$$y[k] = \mathbf{h}^T[k] \mathbf{x}[k] \quad (15)$$

where $y[k] = NH_{4,out}[k]$.

Adjustment of the parameter vector, $\mathbf{h}[k]$, is achieved by the well-known algorithm RLS [14]:

$$\mathbf{h}[k] = \mathbf{h}[k-1] + \mathbf{g}[k].(y[k] - \mathbf{h}^T[k-1]\mathbf{x}[k]) \quad (16)$$

where the adapting gain $\mathbf{g}[k]$ is calculated by the relation:

$$\mathbf{g}[k] = \frac{\mathbf{C}_{xx}^{-1}[k-1]\mathbf{x}[k]}{1 + \mathbf{x}^T[k]\mathbf{C}_{xx}^{-1}[k-1]\mathbf{x}[k]} \quad (17)$$

and the autocorrelation matrix inverse, $\mathbf{C}_{xx}^{-1}$, is updated iteratively:

$$\mathbf{C}_{xx}^{-1}(k) = \mathbf{C}_{xx}^{-1}(k-1) - \mathbf{g}(k).\mathbf{x}^T(k)\mathbf{C}_{xx}^{-1}(k-1) \quad (18)$$

Based on the impulse responses $\mathbf{h}_a[k]$ and $\mathbf{h}_o[k]$, determined by the algorithm RLS, the process frequency characteristics can be obtained and they represent the starting point of the interactive frequency design methods of the trickling biofilter control algorithms.

IV. SENSITIVITY ANALYSIS OF THE TRICKLING BIOFILTER MODEL WHEN CHANGING THE FUNCTIONING REGIME

With the trickling biofilter there are three variables that can change the functioning point: the inflow rate of the trickling biofilter (which is in fact the recirculation flow), Q_{in} , the ammonium concentration in the trickling biofilter influent (which is the ammonium concentration in the aquaculture tanks effluent), $NH_{4,in}$, and the dissolved oxygen concentration in the trickling biofilter effluent (as determined by the dissolved oxygen concentration in the aquaculture tank effluent and the aeration and biological processes in the trickling biofilter).

In order to highlight the changing of the adaptive model dynamical properties when changing the biofilter functioning regime, three functioning regimes have been considered.

- *High Flow Regime*: $Q_{in} = 4 \text{ m}^3 / \text{h}$ and $NH_{4,in} = 2 \text{ mgN} / \text{l}$;
- *Medium Flow Regime*: $Q_{in} = 3 \text{ m}^3 / \text{h}$ and $NH_{4,in} = 2.667 \text{ mgN} / \text{l}$;
- *Low Flow Regime*: $Q_{in} = 2 \text{ m}^3 / \text{h}$ and $NH_{4,in} = 4 \text{ mgN} / \text{l}$;

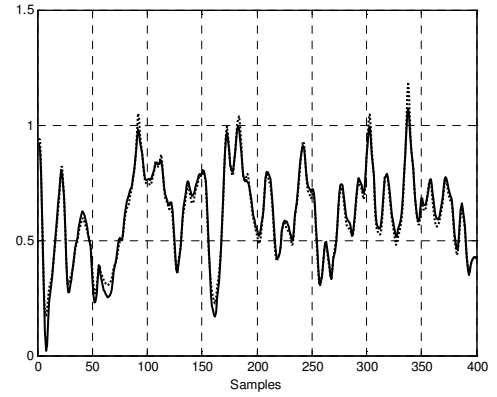


Fig. 5. Validation of identified model using adaptive filters (solid line – model output and dotted line – emulated process output)

It is noted that in all three functioning regimes in the aquaculture system there is the same amount of nitrogen: 8 g. It is also noted that in this situation a constant value for the dissolved oxygen concentration was considered: $DO_{in} = 4 \text{ mg } O_2 / \text{l}$. Software implementation of the analytical model obtained in Section II was used as process emulator, obtaining the process output for the three functioning regimes. An example of identification using adaptive filters is shown in Figure 5. From Figure 5 it can be seen a very good correspondence between the identified model output and the emulated process.

In Figures 6 and 7 Bode and Nyquist frequency characteristics of channel $Q_{in}(t) \rightarrow NH_{4,out}(t)$ are presented. The analysis of these characteristics shows that the water flow fed into the trickling biofilter has a significant influence on the process dynamics resulting in the changing of the gain and time constants of the frequency models identified in this channel.

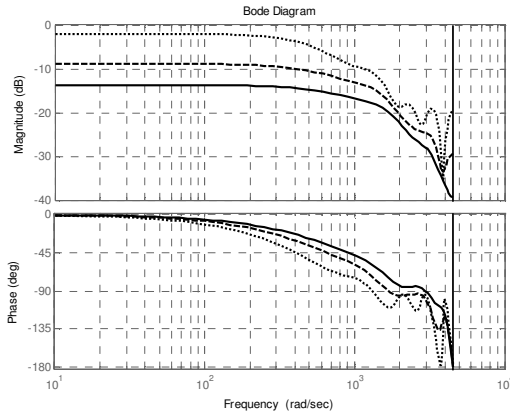


Fig. 6. Bode frequency characteristic of channel $Q_{in}(t) \rightarrow NH_{4,out}(t)$ (solid line – High Flow Regime, dashed line – Medium Flow Regime and dotted line – Low Flow Regime)

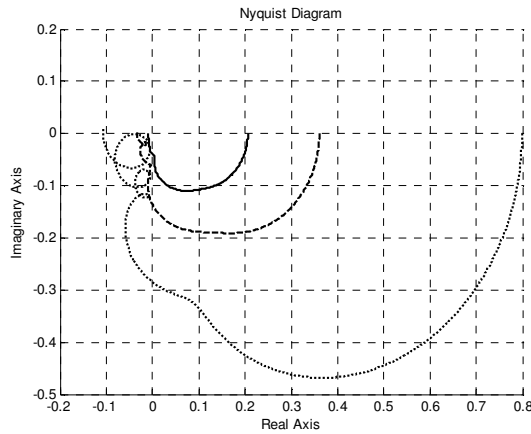


Fig. 7. Nyquist frequency characteristic of channel $Q_{in}(t) \rightarrow NH_{4,out}(t)$ (solid line – High Flow Regime, dashed line – Medium Flow Regime and dotted line – Low Flow Regime)

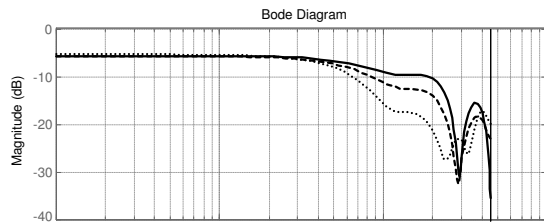


Fig. 8. Bode frequency gain characteristic of channel $NH_{4,in}(t) \rightarrow NH_{4,out}(t)$ (solid line – High Flow Regime, dashed line – Medium Flow Regime and dotted line – Low Flow Regime)

Figures 8 and 9 illustrate the Bode and Nyquist frequency characteristics of the channel $NH_{4,in}(t) \rightarrow NH_{4,out}(t)$. This is a disturbance channel, the ammonium concentration at the trickling biofilter influent being determined by metabolic processes occurring in the aquaculture tanks. Analyzing the mentioned figures it is observed that, despite a significant influence of this channel on the output, in the three functioning regimes this channel has similar dynamic properties at low and medium frequencies.

Finally an analysis of the dynamic properties of channel $DO_{in}(t) \rightarrow NH_{4,out}(t)$ is performed. For this three new functioning regimes were considered, where the flow and ammonium concentration in the trickling biofilter influent

are constant, varying only the concentration of dissolved oxygen by values: 3, 4 and 5 $mg\ O_2/l$. The frequency analysis of this channel is shown in Figure 10. From this figure it can be seen that on this channel there is no significant dynamics within the frequency range of interest.

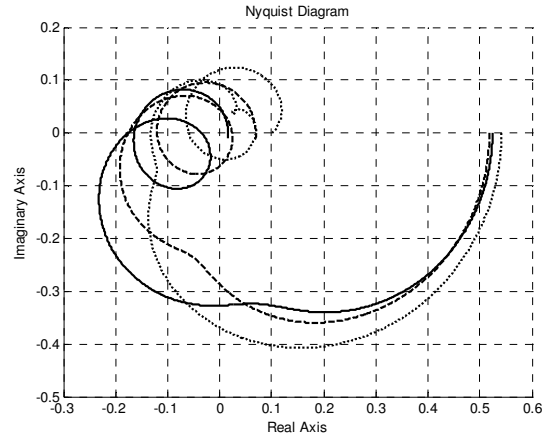


Fig. 9. Nyquist frequency characteristic of channel $NH_{4,in}(t) \rightarrow NH_{4,out}(t)$ (solid line – High Flow Regime, dashed line – Medium Flow Regime and dotted line – Low Flow Regime)

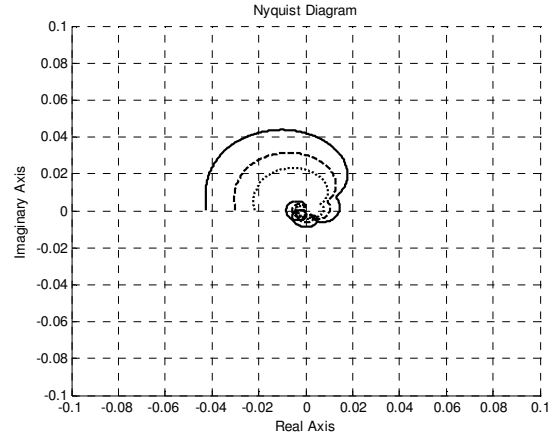


Fig. 10. Nyquist frequency characteristic of channel $DO_{in}(t) \rightarrow NH_{4,out}(t)$ (solid line – $DO_{in} = 3\ mg\ O_2/l$ Regime, dashed line – $DO_{in} = 4\ mg\ O_2/l$ Regime and dotted line – $DO_{in} = 5\ mg\ O_2/l$ Regime)

V. APPROACHES IN DESIGNING CONTROL SYSTEMS FOR INTENSIVE RECIRCULATING AQUACULTURE PLANTS

In the previous section it was highlighted the fact that the main control variable of trickling biofilters is the recirculated flow rate. The analysis showed that the dynamic properties of the command channel $Q_{in}(t) \rightarrow NH_{4,out}(t)$ varies widely, both in terms of gain factor and the time constants, with the functioning point. It is therefore necessary to use robust control techniques. It is proposed the approximation of the frequency models obtained for this channel with a variable parameters transfer function in which case the robust control technique QFT (Quantitative Feedback Theory) can be used [15]. Recirculated flow control can be done directly, if in the aquaculture plant there are recirculation pumps with variable flow rate, or indirectly via aquaculture tanks level control.

For the control channel $DO_{in}(t) \rightarrow NH_{4,out}(t)$ the lack of

significant dynamics has been pointed out. Also it has been observed in previous works that using a command to the trickling biofilter aeration does not lead to satisfactory results [9]. In these circumstances, the dissolved oxygen concentration in the trickling biofilter effluent will be indirectly controlled through direct control of the dissolved oxygen concentration in the aquaculture tanks.

The design of a trickling biofilter control system should take into account the disturbance channel $NH_{4,in}(t) \rightarrow NH_{4,out}(t)$. This channel is influenced by the food introduced into the aquaculture tanks and the metabolic processes of the fish population. These processes are the determining factor in setting the functioning regime of a recirculating aquaculture plant. Depending on the ammonium concentration in the aquaculture tanks the recirculation flow is established within the plant. Thus, an ammonium concentration of maximum 1 mgN/l in water must be achieved. For this channel, if there is an ammonium transducer for the trickling biofilter influent feed-forward control techniques can be used. If this disturbance is not measurable robust control techniques can be used.

Given the existing recirculating aquaculture pilot plant, the control system shown in Figure 11 is proposed. The dissolved oxygen control loop has as output the dissolved oxygen concentration in the trickling biofilter effluent, and as control variable the dissolved oxygen concentration in the aquaculture tanks. This control variable is a setpoint for the local control loop of dissolved oxygen in the aquaculture tanks. The disturbance $NH_{4,in}$ has a considerable influence on water quality at the biofilter outlet being influenced in turn by the metabolic processes of the fish population.

VI. CONCLUSION

In this present paper the frequency models of a biological filter from an intensive recirculating aquaculture plant have been established and sensitivity analysis when changing the functioning regime has been conducted. The starting point has been a trickling biofilter analytical model with distributed parameters identified using experimental data. Software implementation of this model was used as an emulator of the process being simulated various functioning regimes of the trickling biofilter. Based on the simulated data sets using adaptive filters, identification and sensitivity analysis of frequency models have been carried out for each input-output channel.

The sensitivity analysis of the trickling biofilter model when changing the functioning regimes led to the conclusion that the most important dynamics is that of $Q_{in}(t) \rightarrow NH_{4,out}(t)$ channel, while $DO_{in}(t) \rightarrow NH_{4,out}(t)$ channel features an insignificant dynamics. Finally, based on the conducted analysis, an automatic control strategy of the intensive recirculating aquaculture plant has been proposed.

REFERENCES

- [1] J. Van Rijn, "The potential for integrated biological treatment systems in recirculating fish culture - A review", *Aquaculture*, vol. 139, pp. 181-201, 1996.
- [2] S.W.H. Van Hulle, J. Verstraete, J. Hogue, P. Dejjans, A. Dumoulin, "Modelling and simulation of a nitrification biofilter for drinking water purification", *Water SA*, vol. 32, no. 2, pp. 252-264, 2006.
- [3] G. Baquerizo, J.P. Maestre, T. Sakuma, M.A. Deshusses, X. Gamisans, D. Gabriel, J. Lafuente, "A detailed model of a biofilter for ammonia removal: Model parameters analysis and model validation", *Chemical Engineering Journal*, vol. 113, no. 2-3, pp. 205-214, 2005.
- [4] I. Iacopozzi, V. Innocenti, S. Marsili-Libelli, E. Giusti, A modified Activated Sludge Model No. 3 (ASM3) with two-step nitrification-denitrification. *Env. Mod. Software*, vol. 22, pp. 847 – 861, 2007
- [5] M. Barbu, S. Caraman, E. Ceangă, A Modified ASM3 Model for a Trickling Filter, *Romanian Biotechnological Letters*, vol. 13, no. 5, pp. 39-48, 2008.
- [6] B.E. Logan, *Modeling and optimizing the performance of nitrifying trickling filters (NTFs)*, Available: www.engr.psu.edu/ce/enve/Pres-web/NTF-lecture_files/frame.htm, 2000.
- [7] H. Vanhooren, "Modelling for optimization of biofilm wastewater treatment process: a complexity compromise", PhD Thesis, University of Ghent, 2002.
- [8] T. Wik, Rational transfer function models for nitrifying trickling filters, *Wat. Sci. Tech.*, vol. 39, no. 4, pp. 121-128, 1999.
- [9] S. Caraman, M. Barbu, T. Ionescu, G. Ifrim, V. Cristea, E. Ceangă, The analysis of the dynamic properties of the wastewater treatment process in a recirculating aquaculture system, *Romanian Biotechnological Letters*, vol. 15, no. 4, pp. 5457-5466, 2010.
- [10] M. Barbu, S. Caraman, G. Bahrim, D. Carp, Results regarding the control of the dissolved oxygen concentration in wastewater treatment processes, *Romanian Biotechnological Letters*, vol. 16, no. 2, pp. 6096-6104, 2011.
- [11] P. Reichert, Design techniques of a computer program for the identification of processes and the simulation of water quality in aquatic systems, *Environmental Software*, vol. 10, no. 3, pp. 199-210, 1995.
- [12] M. Henze, W. Gujer, T. Mino, M. van Loosdrecht, *Activated Sludge Models ASM1, ASM2, ASM2d and ASM3*. IWA Publishing, 2000.
- [13] M. Barbu, C. Picioreanu, Model of Trickling Biofilter from an Intensive Recirculating Aquaculture System, *IWA Biofilm Conference 2011*, accepted for publication.
- [14] S. Haykin, *Adaptive Filter Theory*, Prantice Hall, 1995.
- [15] M. Barbu, S. Caraman, *QFT Robust Control of Wastewater Treatment Processes*, In: *Robust Control, Theory and Applications*, Intech, 2011.

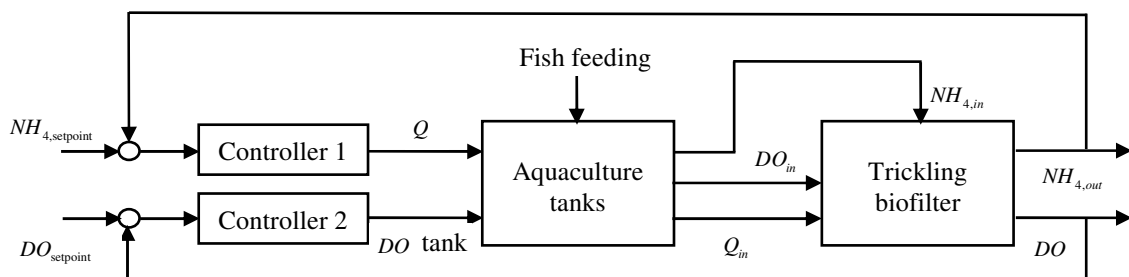


Fig. 11. Scheme of the automatic control of an intensive recirculating aquaculture system

Automatic Unsupervised Shape Recognition Technique using Moment Invariants

Tudor Barbu

Abstract — We approach the shape recognition domain in this paper. After an introduction in the image shape analysis domain, we describe a shape feature extraction technique using moment-based measures which are invariant to geometric transforms. Then, an automatic unsupervised feature vector classification approach is proposed. It is based on a sequence of hierarchical agglomerative region-growing clustering algorithms and a measure based on cluster validation indexes. The results of this provided recognition technique can be applied successfully in important domains, such as object recognition, shape-based image content indexing and retrieval.

I. INTRODUCTION

Shape recognition represents a very important computer vision domain, consisting of recognizing image objects, based on their shape information [1]. While supervised shape recognition classifies image objects by comparing them with some registered shape templates, unsupervised shape recognition groups a set of objects in several clusters, based on their shape similarity.

A successful shape recognition approach requires a robust shape descriptor, representing a measure that captures the shape features in a concise manner and is invariant to all geometric transformations. Shape analysis approaches are divided into two main categories: region-based and contour-based methods.

In region-based methods descriptors are derived using all the pixel information inside a shape region, while contour-based descriptors express the shape information of the object outline. The most important region-based recognition techniques are those based on the Angular Radial Transform Descriptor (ARTD) [2], Zernike moments [3] and Legendre moments [4]. Contour-based shape recognition includes techniques based on Fourier Descriptors [5, 6], Curvature Scale Space Descriptors (CSSD) [6], Contour Trees [7], Reeb Graphs [8] and invariant moments [9-11].

We propose an automatic unsupervised moment-based shape recognition approach in this paper. A shape detection process should be performed before shape recognition, but we do not insist on the image object detection problem in this paper.

In our previous works, we proposed object detection approaches based on color/texture segmentation or image binarization and foreground extraction, which can be used in this case [12]. Other shape detection solutions are based on

edge-detection [13], sliding-windows or generalized Hough transforms [14]. The identified image objects are then recognized by their shapes.

A shape feature extraction method using some moment invariants and producing strong shape descriptors is provided in the next section. In the third section we propose a novel unsupervised classification algorithm for the obtained shape feature vectors.

Some experiments performed using the proposed object shape recognition technique will be described in the fourth section. The paper will finalize with a conclusion section.

II. MOMENT-BASED SHAPE FEATURE EXTRACTION

We consider the following image recognition task. Let $\{Ob_1, \dots, Ob_n\}$ the image objects to be recognized by their shape information. These objects could belong to one or more images and have to be classified in a number of shape classes.

We propose a moment-based object shape feature extraction approach [15]. The computed feature vectors represent robust shape descriptors, being translation, scaling and rotation invariant measures.

So, if $f_k(x, y)$ represents the gray-value function of the object area Ob_k , then the area moment of that object is computed as $m_{ij}(k) = \sum_x \sum_y f_k(x, y) \cdot x^i \cdot y^j$. We perform

shape analysis only and we are not interested in object content given by gray level or color values, therefore a binary form could be used for the object function: $f_i(x, y) \in \{0, 1\}$.

Then, the centered area moments of Ob_k , which are translation invariant, are determined as:

$$\mu_{ij}(k) = \sum_x \sum_y f_k(x, y) \cdot (x - C_x)^i \cdot (y - C_y)^j, \quad i, j \geq 0 \quad (1)$$

where $k \in \{1, \dots, n\}$ and $C_x = \frac{m_{10}}{m_{00}}$, $C_y = \frac{m_{01}}{m_{00}}$ represent

the coordinates of the object's centroid. We propose the following feature vector for the image object shape description:

$$V(Ob_k) = [(\eta_{30}(k) + \eta_{12}(k))^2 + (\eta_{03}(k) + \eta_{21}(k))^2, \eta_{24}(k) + \eta_{42}(k)] \quad (2)$$

Manuscript received May 10, 2011. Tudor Barbu is with the Institute of Computer Science of the Romanian Academy, Iasi branch (e-mail: tudbar@iit.tuiasi.ro).

where $\eta_{ij}(k) = \mu_{ij}(k) / \mu_{00}^{(i+j)/2+1}(k)$ represents a scaling invariant moment.

The resulted feature vector $V(Ob_k)$ is invariant to the geometric transforms: scaling, translation, rotation and mirroring. The first component of the feature vector, $(\eta_{30}(k) + \eta_{12}(k))^2 + (\eta_{03}(k) + \eta_{21}(k))^2$ represents a Hu moment used to express the shape eccentricity [11]. The second component, $\eta_{24}(k) + \eta_{42}(k)$, is also an invariant moment-based parameter [12,15].

The modeled feature vector represents a powerful shape descriptor of the image object, its effectiveness being demonstrated by the classification results. The Euclidean distance is used for these feature vectors, in the classification process.

III. AUTOMATIC FEATURE VECTOR CLASSIFICATION TECHNIQUE

The moment-based shape feature vectors obtained in the previous section have to be clustered in a number of classes. Therefore, a classification process has to be performed within the feature set, $\{V(Ob_1), \dots, V(Ob_n)\}$.

We propose a novel automatic classification approach that uses an unsupervised agglomerative hierarchical clustering algorithm and some validation indexes [16]. Our method applies a region-growing procedure repeatedly until the optimal number of clusters is determined.

First, the considered region growing algorithm sets a number of final clusters, let it be $N \leq n$. It starts initially with n clusters, which are

$$C_1 = \{V(Ob_1)\}, \dots, C_n = \{V(Ob_n)\} \quad (3)$$

Then, at each step, the algorithm computes the overall minimum distance between clusters and merges the clusters being at that distance from each other. Thus, for $i < j$, we obtain:

$$d(C_i, C_j) = d_{\min} \Rightarrow C_i = C_i \cup C_j, C_j = \emptyset \quad (4)$$

where $d_{\min} = \min_{i \neq j \in [1, n]} d(C_i, C_j)$ and the distance between clusters is computed through *average linkage clustering*:

$$d(C_i, C_j) = \frac{\sum_{x \in C_i} \sum_{y \in C_j} d_E(x, y)}{|C_i| \cdot |C_j|} \quad (5)$$

where d_E represents the Euclidean distance and $|C_i|$ is the cardinality of that cluster. The region-growing clustering process continues until the number of clusters becomes N [16]. Let $C_1, \dots, C_N$ be the resulted feature vector clusters.

The optimal N value is computed using a combined measure based on cluster validation indexes that must be minimized [17,18]. We consider a threshold T , then run the described algorithm for each $N \in [1, T]$. The ideal threshold value would be $T = n$, but that could produce a great computation complexity for a high number of objects, so we have chosen $T = \lceil n/2 \rceil$ for our experiments.

The proper number of clusters is determined as a combination of the values of Dunn and Davies-Bouldin validity indexes. The Dunn index is used to maximize the inter-cluster distances and minimize the intra-cluster distances [17]. Therefore, the number of clusters that maximize that measure is taken as the optimal number of clusters.

The validity index of Davies and Bouldin is a function of the ratio of the sum of within-cluster scatter to between-cluster separation [18]. Its lowest value indicates an optimal clustering operation. Thus, the optimal N value is computed as follows:

$$N_{\text{optim}} = \arg \min_{N \in [1, T]} \left(DB(N) + \frac{1}{D(N)} \right) \quad (6)$$

where the Davies-Bouldin index is determined as

$$DB(N) = \frac{1}{N} \sum_{i=1}^N \max_{i \neq j} \frac{d(C_i) + d(C_j)}{d(Cen_i, Cen_j)} \quad (7)$$

and the Dunn validity index is obtained as

$$D(N) = \min_{i \in [1, N]} \left\{ \min_{j \neq i} \left\{ \frac{d(Cen_i, Cen_j)}{\max_{t \in [1, K]} d(C_t)} \right\} \right\} \quad (8)$$

where Cen_i and $d(C_i)$ are the centroid and the intra-cluster distance of C_i . The obtained set of feature vector clusters $C_1, \dots, C_{N_{\text{optim}}}$ represents the final shape recognition result.

IV. EXPERIMENTS

We have performed numerous experiments using the described shape recognition approach. The proposed technique has been tested on various image datasets and satisfactory results have been obtained.

A high recognition rate, of approximately 85 %, is achieved by our method. It represents a better rate than those of many other object recognition approaches. We have used the threshold value $T = \lceil n/2 \rceil$ in our experiments. The object recognition rate rises if T value is increased, but the procedure complexity and time execution rise as well.

We obtain high values for the performance parameters (*Precision*, *Recall* and F_1) of this recognition process. Thus, we get *Precision* = 0.85 and *Recall* = 0.84. That means our

recognition method produces very few missed hits or false positives.

A shape recognition example is described in the next figures. Thus, in Fig. 1 there is displayed an image containing objects which can be segmented easily. Each Ob_i is marked with the i value in the picture, $i = 1, \dots, 9$.

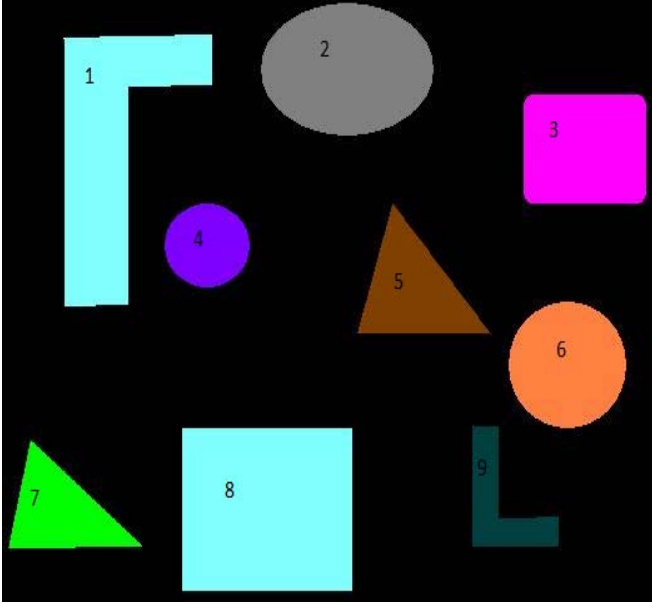


Fig. 1 Image containing several objects

The image is then converted in the binary form, using a threshold value of 45, that is determined empirically, then the binary image is processed using some morphological operations, to eliminate the small white and black spots and retain only the important image regions.

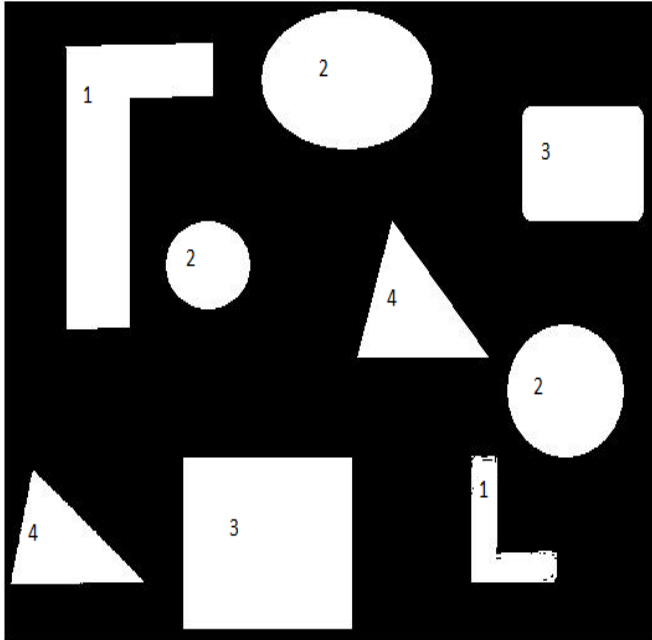


Fig. 2 The resulted shape classes

The connected components of the enhanced binary image are detected, thus the image foreground, containing the main objects, being extracted. The feature extraction process is then applied on the detected shapes.

The obtained shape feature vectors are then classified using the presented automatic clustering algorithm. A threshold value $T = 5$ is used in the process.

The shape classes corresponding to the image from Fig. 1 are represented in Fig. 2. As one can see in that figure, each shape is labeled with the number of its class. The four detected shape classes, representing the final recognition result, are: $\{Ob_1, Ob_9\}$, $\{Ob_2, Ob_4, Ob_6\}$, $\{Ob_3, Ob_8\}$ and $\{Ob_5, Ob_7\}$.

V. CONCLUSION

A novel automatic unsupervised shape recognition technique has been presented in this paper. As we have mentioned in the introduction, our article has focused on object shape recognition, and not on the image object detection issue, that was investigated in our previous works [12,15].

We consider that our paper brings some important contributions to the shape analysis domain. Novelty elements are brought in both the shape feature extraction stage and the feature vector classification stage.

Thus, we propose an original moment-based feature extraction approach, combining the well-known Hu moments with another measure based on invariant moments, to create a robust shape descriptor. The automatic feature vector classification method represents the main original contribution of this paper. Thus, we have considered a region-growing clustering algorithm and created a combined cluster validation measure to determine the optimal number of classes.

The identification of the proper number of shape clusters automatizes the classification method, which represents a very important thing. The automatic character of the provided recognition approach eliminates the interactivity from the process, a human user not being required to intervene.

That means our recognition technique can be used successfully for very large sets of images, containing a high number of shapes. Therefore, its automatic and unsupervised character makes our shape recognition approach very useful in some important computer vision application areas, such as shape-based indexing of large image databases and object retrieval from these databases.

The performed experiments, producing a satisfactory recognition rate, prove the effectiveness of the proposed technique. Improving the recognition rate and reducing the process complexity represent some task of our future work in the shape analysis domain.

Thus, we will compute some other moment-based shape descriptors and will consider using feature vectors based on other invariant characteristics. Also, we intend to investigate some other automatic unsupervised feature vector classification solutions.

Thus, we consider using an unsupervised K - means algorithm [19], or some variants of it like *Fuzzy K-Means* [20], instead of the region-growing procedure. We could also use in the shape recognition process an automatic clustering approach that is not based on validation indexes at all.

Obviously, another important application area of this shape recognition technique is represented by the image object recognition domain [12]. The results of our method can be used in combination with some color and texture analysis results to produce successful object recognition.

Our future research will focus on the image object recognition area, but also on content-based image indexing and retrieval domains [21].

REFERENCES

- [1] L. Floriani, M. Spagnuolo, *Shape Analysis and Structuring*, Springer, 2007, 978-3540332640.
- [2] R. J. Couerjolly, D. Baskurt, "Generalizations of angular radial transform for 2D and 3D shape retrieval", *Technical report, laboratoire LIRIS*, Claude Bernard University, Lyon, 2004.
- [3] A. Khotanzad, Y. H. Hong, "Invariant Image Recognition by Zernike Moments", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 5, May 1990.
- [4] V. P. D. Kumar, T. Tessamma, "Performance Study of an Improved Legendre Moment Descriptor as Region-based Shape Descriptor", *Pattern Recognition and Image Analysis*, Vol. 18, No. 1, Jan. 2008, pp. 23-29.
- [5] E. Persoon, K. Fu, "Shape Discrimination Using Fourier Descriptors", *IEEE Trans. On Systems, Man and Cybernetics*, Vol. SMC-7, No. 3, 1977, pp.170-179.
- [6] D. Zhang, G. Lu, "A comparative study of curvature scale space and Fourier descriptors for shape-based image retrieval", *Journal of Visual Communication and Image Representation*, Vol. 14, No. 1, 2003, pp. 39-57.
- [7] X. Zhang, "Complementary Shape Comparison with Additional Properties", *Volume Graphics*, 2006, pp. 79-86.
- [8] V. Natarajanb, H. Doraiswamy, "Efficient algorithms for computing Reeb graphs", *Computational Geometry*, No. 42, 2009, pp. 606 – 616.
- [9] S. O. Belkasim, M. Shridhar, M. Ahmadi, "Pattern recognition with moment invariants: a comparative study and new results", *Pattern Recognition*, Vol. 24, No. 12, 1991, pp. 1117-1138.
- [10] I. Sommer, O. Müller, F. S. Domnigues, O. Sander, J. Weickert, T. Lengauer, "Moment invariants as shape recognition technique for comparing protein binding sites", *Oxford Journals, Bioinformatics*, Vol. 23, No. 23, 2007, pp. 3139-3146.
- [11] M. K. Hu, "Visual pattern recognition by moment invariants", *IEEE Transactions on Information Theory*, Vol. 8, 1962, pp. 179–187.
- [12] T. Barbu, "An Approach to Image Object Recognition and Interpretation", *Proceedings of the Romanian Academy, Series A*, Vol. 4, No. 3, sept.-dec. 2003, pp.217-223.
- [13] M. Sarifuddin, R. Missaoui, J. Vaillancourt, M. Paindavoine, "An Optimal Edge Detector For Automatic Shape Extraction", In *International Conference On Signal-Image Technology & Internet-Based Systems*, Hammamet, Tunisia, 2006.
- [14] V. F. Leavers, "The Dynamic Generalized Hough Transform", *Lecture Notes in Computer Science*, Vol. 427, 1990, pp. 592 - 594.
- [15] T. Barbu, *Multimedia Information System Modelling* (in Romanian), Publishing House of the Romanian Academy, Bucharest, 2006, 224 pages.
- [16] T. Barbu, "An Automatic Unsupervised Pattern Recognition Approach", *Proceedings of the Romanian Academy, Series A*, Vol. 7, No. 1, 2006, pp. 73-78.
- [17] J. Dunn, "Well separated clusters and optimal fuzzy partitions", In *Journal of Cybernetics*, Vol. 4, 1974, pp. 95-104.
- [18] D. L. Davies, D. W. Bouldin, "A cluster separation measure", In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 1, No. 4, 2000, pp. 224-227.
- [19] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, A. Y. Wu, "An Efficient K-Means Clustering Algorithm: Analysis and Implementation", In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 7, 2002, pp. 881-892.
- [20] M. Sarkar, T. Y. Leong, "Fuzzy K-means clustering with missing values", In *Proceedings of 2001 AMIA Symposium*, 2001, pp. 588 - 592.
- [21] O. Starostenko, J. Rodríguez-Asomoza, S. E. Sánchez-López, J. A. Chévez-Aragón, "Shape Indexing and Retrieval: A Hybrid Approach Using Ontological Descriptions", *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*, 2008, pp. 381-386.

Stabilization of Navier–Stokes equations by oblique boundary feedback controllers

Viorel Barbu

I. EXTENDED ABSTRACT

Consider the Navier–Stokes system

$$\begin{aligned} \frac{\partial Y}{\partial t} - \nu \Delta Y + (Y \cdot \nabla)Y &= \nabla p + f_e & \text{in } (0, \infty) \times \mathcal{O}, \\ \nabla \cdot Y &= 0 & \text{in } (0, \infty) \times \mathcal{O}, \\ Y &= v & \text{on } (0, \infty) \times \partial\mathcal{O}, \\ Y(0) &= y_0 & \text{in } \mathcal{O}, \end{aligned} \quad (1)$$

in a bounded open domain $\mathcal{O} \subset \mathbb{R}^d$, $d = 2, 3$, with a smooth boundary $\partial\mathcal{O}$ which, for simplicity, we assume simple connected. Here $\nu > 0$, f_e is a given smooth function and v is a boundary input. If y_e is an equilibrium solution to (1), then (1) can be, equivalently, written as

$$\begin{aligned} \frac{\partial Y}{\partial t} - \nu \Delta Y + (y_e \cdot \nabla)Y + (Y \cdot \nabla)y_e + (Y \cdot \nabla)Y &= \nabla p \\ &\text{in } (0, \infty) \times \mathcal{O}, \\ \nabla \cdot Y &= 0 & \text{in } (0, \infty) \times \mathcal{O}, \\ Y &= u & \text{on } (0, \infty) \times \partial\mathcal{O}, \\ Y(0) &= y_0 - y_e & \text{in } \mathcal{O}. \end{aligned} \quad (2)$$

Our main concern here is the design of an oblique boundary feedback controller which stabilizes exponentially the equilibrium state y_e , or, equivalently, the zero solution to (2). The main step toward this end is the stabilization of the linear system corresponding to (2) or, more generally, of the Oseen–Stokes system

$$\begin{aligned} \frac{\partial Y}{\partial t} - \nu \Delta Y + (Y \cdot \nabla)a + (b \cdot \nabla)Y &= \nabla p & \text{in } (0, \infty) \times \mathcal{O}, \\ \nabla \cdot Y &= 0 & \text{in } (0, \infty) \times \mathcal{O}, \\ Y &= u & \text{on } (0, \infty) \times \partial\mathcal{O}, \end{aligned} \quad (3)$$

where $a, b \in (C^2(\overline{\mathcal{O}}))^d$, $\nabla \cdot a = \nabla \cdot b = 0$ in $\mathcal{O}$. Besides its significance as first order linear approximation of (2), this system models the dynamics of a Stokes flow with inclusion of a convection acceleration $(b \cdot \nabla)Y$ and also the disturbance flow induced by a moving body in a Stokes fluid flow.

In its complex form, the main result of this work amounts to saying that there is a boundary feedback controller of the form

$$u(t, x) = \eta \sum_{j=1}^N \mu_j \left(\int_{\mathcal{O}} Y(t, x) \overline{\varphi_j^*}(x) dx \right) (\phi_j(x) + \alpha(x) \vec{n}(x)), \quad t \geq 0, x \in \partial\mathcal{O}, \quad (4)$$

which stabilizes exponentially system (1). Here $\phi_j = \text{lin} \left\{ \frac{\partial \varphi_i^*}{\partial n} \right\}_{i=1}^N$ and $\{\varphi_j^*\}_{j=1}^N$ is an eigenfunction system for

the adjoint of the Stokes–Oseen operator

$$\begin{aligned} \mathcal{L}\varphi &= -\nu \Delta \varphi + (a \cdot \nabla)\varphi + (\varphi \cdot \nabla)b - \nabla p, \quad \varphi \in D(\mathcal{L}), \\ D(\mathcal{L}) &= \{\varphi \in (H^2(\mathcal{O}))^d \cap (H_0^1(\mathcal{O}))^d; \nabla \cdot \varphi = 0 \text{ in } \mathcal{O}\}. \end{aligned} \quad (5)$$

It turns out that this feedback controller also stabilizes the Navier–Stokes system (2) in a neighborhood of the origin.

In (4), N is the number of the eigenvalue λ_j of $\mathcal{L}$ with $\text{Re } \lambda_j < 0$, $\alpha \in C(\overline{\mathcal{O}})$ is an arbitrary function with zero circulation on $\partial\mathcal{O}$, that is,

$$\int_{\mathcal{O}} \alpha(x) dx = 0 \quad (6)$$

and $\phi_j = \text{lin span} \left\{ \frac{\partial \varphi_j^*}{\partial n} \right\}_{i=1}^N$. (In its real version, the stability controller u is of the form (4) but with $\{\varphi_j^*\}$ replaced by $\{\text{Re } \varphi_j, \text{Im } \varphi_j\}$.)

Taking into account that the vectors $\frac{\partial \varphi_j^*}{\partial n}(x)$ are tangential at each $x \in \partial\mathcal{O}$, that is, $\frac{\partial \varphi_j^*}{\partial n} = \nu \vec{\tau}$, where $\vec{\tau}$ is the tangent vector (see [11], p. 35), we see that u is an oblique vector field on $\partial\mathcal{O}$.

More precisely, we have

$$u(t, x) \cdot \vec{n}(x) = \alpha(x), \quad \forall x \in \partial\mathcal{O}, \quad (7)$$

$$|\cos \langle u(t, x), \vec{n}(x) \rangle| \geq 1 - \frac{C}{C + |\alpha(x)|}, \quad \forall x \in \partial\mathcal{O}, \quad (8)$$

where $C > 0$ is independent of α . This means that the “stabilizable boundary controller u can be chosen “almost” normal to $\partial\mathcal{O}$. However, for technical reasons the limit case $|\alpha| \equiv +\infty$, that is, u normal is excluded from our discussion.

It should be said that in the stabilization literature, only in a few situations was designed a normal stabilizable controller for equation (1) and this for periodic flows in 2– D channels (see, e.g., [1], [2], [3], [24], [25], [26]). However, even in this case, the feedback controller is not given in explicit form and sometimes one assumes restrictive conditions on ν or on the spectrum of the operator $\mathcal{L}$.

It should be said that there is a large body of results obtained in recent years on boundary stabilization of system (1) and here the works [10], [11], [13], [14], [16], [17], [19], [20] should be primarily cited. (See, also, [7], [14], [15], [21].) The approach used in these works can be described in a few words as follows; one decomposes system (1) in a finite-dimensional unstable part which is exactly controllable and an infinite-dimensional part which is exponentially stable and proves so its stabilization by open loop boundary controller with finite-dimensional structure. Then one designs in a standard way a stabilizable feedback controller via

V. Barbu is with the Octav Mayer Inst. of Math. of Romanian Academy, Iași, România

the infinite-dimensional algebraic Riccati equation associated with an infinite horizon quadratic optimal control problem. Our construction of boundary stabilizable controller for (1) avoids the Riccati equation based approach which though provides a robust controller it is, however, untreatable from computational point of view. Instead, we propose an explicit feedback controller of the form (4) easy to implement into system. It should be said that this construction resembles the form of stabilizable noise controllers recently designed in the author's works [4], [5], [6], [8], [9], which seem to be, however, more robust to stochastic perturbations.

REFERENCES

- [1] Aamo O.M., Krstic M., Bewley T.R., Control of mixing by boundary feedback in 2D-channel, *Automatica*, 39 (2003), 1597-1606.
- [2] Balogh A., Liu W.-L., Krstic M., Stability enhancement by boundary control in 2D channel flow, *IEEE Trans. Autom. Control*, 11 (2001), 1696-1711.
- [3] Barbu V., Stabilization of a plane channel flow by wall normal controllers, *Nonlinear Anal. Theory – Methods Appl.*, 56 (2007), 145-168.
- [4] Barbu V., The internal stabilization by noise of the linearized Navier–Stokes equation, *ESAIM COCV*, 2 (2009).
- [5] Barbu V., Stabilization of a plane periodic channel flow by noise wall normal controllers, *Syst. Control Lett.*, 50 (10) (2010), 608-618.
- [6] Barbu V., Exponential stabilization of the linearized Navier–Stokes equation by pointwise feedback controllers, *Automatica*, 46 (12) (2010), 1890-1895.
- [7] Barbu V., *Stabilization of Navier–Stokes Flows*, Communications and Control Engineering, Springer, London, 2010.
- [8] Barbu V., Internal stabilization of the Oseen–Stokes equations by Stratonovitch noise, *Systems & Control Letters*, doi:10.1016/j.sycont.2011.04.019
- [9] Barbu V., Da Prato G., Internal stabilization by noise of the Navier–Stokes equation, *SIAM J. Control Optim.*, 49 (1) (2010), 1-21.
- [10] Barbu V., Lasiecka I., Triggiani R., Abstract setting for tangential boundary stabilization of Navier–Stokes equations by high and low-gain feedback controllers, *Nonlinear Anal.*, 64 (2006), 2704-2746.
- [11] Barbu V., Lasiecka I., Triggiani R., Tangential boundary stabilization of Navier–Stokes equations, *Mem. Am. Math. Soc.*, 852 (2006), 1-145.
- [12] Barbu V., Lasiecka I., Triggiani R., The unique continuations property of eigenfunctions to Stokes–Oseen operator (submitted).
- [13] Barbu V., Triggiani R., Internal stabilization of Navier–Stokes equations with finite dimensional controllers, *Indiana Univ. Math. J.*, 53 (2004), 1443-1494.
- [14] Badra M., Feedback stabilization of the 2–D and 3–D Navier–Stokes equations based on an extended system, *ESAIM COCV*, 15 (2009), 934-968.
- [15] Badra M., Lyapunov functions and local feedback boundary stabilization of the Navier–Stokes equations, *SIAM J. Control Optim.*, 48 (2009), 1797-1830.
- [16] Fursikov A.V., Real processes of the 3–D Navier–Stokes systems and its feedback stabilization from the boundary. In: Agranovic M.S., Shubin M.A. (eds.), AMS Translations. Partial Differential Equations, M. Vishnik Seminar, 2002, 95-123.
- [17] Fursikov A.V., Stabilization for the 3–D Navier–Stokes systems by feedback boundary control, *Discrete Contin. Dyn. Syst.*, 10 (2004), 289-314.
- [18] Munteanu I., Normal feedback stabilization of periodic flows in a 2–D channel, *JOTA* (to appear).
- [19] Raymond J.P., Feedback boundary stabilization of the two dimensional Navier–Stokes equations, *SIAM J. Control Optim.*, 45 (2006), 790-828.
- [20] Raymond J.P., Feedback boundary stabilization of the three dimensional inequnations, *J. Math. Pures Appl.*, 87 (2007), 627-669.
- [21] Raymond J.P., Thevenet L., Boundary feedback stabilization of the two dimensional Naier–Stokes equations with finite dimensional controllers (to appear).
- [22] Temam R., *Navier–Stokes Equations and Nonlinear Functional Analysis*, SIAM, Philadelphia, 1985.
- [23] Temam R., *Navier–Stokes Equations*, North–Holland, Amsterdam, 1979.
- [24] Triggiani R., Stability enhancement of a 2–D linear NavierStokes channel flow by a 2–D, wall-normal boundary controller, *Discrete Contin. Dyn. Syst.*, Series B (DCDS-B) 8 (2) (2007), 279-314.
- [25] Vazquez R., Krstic M., A closed-form feedback controller for stabilization of linearized Navier–Stokes equations: The 2D Poiseuille flow, *IEEE Trans. Automatica*, 2005, 10.1109/CDC.2005.1583349
- [26] Vazquez R., Tvelat E., Coron J.M., Control for fast and stable Laminar-to-High-Reynolds-Number transfer in a 2D channel flow, *Discrete Contin. Dyn. Syst.*, SB 10 (2008).

An algorithm designed to determine the optimum number of communication channels in a modular simulator

Lucian-Florentin M. Barbulescu

Abstract— The distributed modular application represent the next step in developing simulators for industrial applications. In this way several simple software components can be created, placed on different machines within a high-speed network and linked together in order to obtain a simulator for an installation. The main problem found in this approach is determining the optimum way of linking together those software components. The current paper offers an algorithm that can be used to determine the optimum number of communication channels in a modular simulator based on the type of data exchanged within the system. This algorithm is part of a complete distributed simulation framework written in Java.

I. INTRODUCTION

THE development of computer hardware and computer networks in our society has conducted to the possibility of developing applications that use the computation power of several machines in order to fulfill their purpose. This had impact in all domains and led to the development of a specific category of simulators, namely the distributed modular simulators[1]. Those simulators are composed of modules that run independently and can even be placed on different machines, as long as they can exchange data between them.

To create a distributed simulator for an industrial installation one approach is to split it into functional components (for example: pumps, engines, pipes etc.) and to create simulators for each one of them. At the end, by interconnecting those modules, a simulator for the entire system will be created. The method of interconnecting those modules is very important for the performance of the system and this is a place where performance can be fine-tuned.

Before analyzing the methods of interconnection let's take a look at the modules. Those are monolithic simulators for different physical systems and their state evolves in time. Those being said it is easy to see that the modules are in fact dynamic systems [2] and are characterized by input variables, output variables, state variables and parameters. While the state variables and parameters are internal to the module, the input variables originate from the outside and

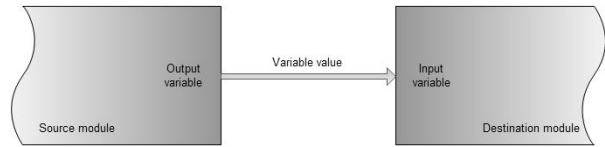


Fig. 1. The value of an output variable from the source module is transferred as the value of an input variable of the destination module.

the output variables are sent to the outside environment. To link together two modules A and B, a method of transporting the value of an output variable from module A as the value of an input variable of module B must be decided.

One solution is to use some software components called communication channels[3] to act as bridges between the modules that need to communicate. Those components have a simple role of reading any value that a module connected to them will send and transport it to any module connected to them that waits for data. It is the responsibility of the receiving module to interpret the data and decide if it is useful for it or it can be discarded. Another function of the communication channel is that of machine separation, that is, if required, the source and destination modules can be placed on different machines and the channel is responsible with their interconnection.

The problem with this approach is that new software components, with no direct link to the system are introduced and this can have a negative effect on the system performance. In the analysis performed in [3] it was discovered that the size (in bytes) of the exchanged data has little effect on the transmission time. In other words, the time needed to send 1 byte through a communication channel is comparable to the time needed to send 32 bytes. This means that the only thing that matters in computing the time needed to exchange the variables is the number of communication channels. Although the communication channels are performing fast[3], they are still software components that use CPU time and computer memory in order to function. To improve the performance of the system an optimum number of communication channels must be determined. The minimum number of communication channels is one and all modules send their output variables to it and receive their input variables from it. This approach has the advantage of a small memory footprint for the communication channels but a lot of CPU time is wasted by the modules with discarding the data they receive and do not require.

Lucian-Florentin M. Barbulescu is with the University of Craiova, of Automation, Computers and Electronics, Computer Science Department, Bvd. Decebal, Nr. 107, 200440, Craiova, Dolj, ROMANIA (phone: +40745343516; fax: +40251438198; e-mail:lucian.barbulescu@cs.ucv.ro).



Fig. 2. A special component called communication channel is added between the source and destination modules.

The alternative is to use one communication channel for each transfer of one variable from a module to another. In this way each module receive only the variables it requests and it will never waste CPU time, but the number of communication channels, and implicitly the CPU time and memory allocated to them, will be at maximum. The solution is between those values and the following algorithm is suggested.

II. THE ALGORITHM

This algorithm can be used to obtain the optimum number of communication channels, which is a value between the minimum and maximum number presented above. This number is defined as the minimum number of communication channels for which every module sends the value of a variable only once and receives only the values of the variables it requires.

The algorithm has four steps:

--Step 1: Break the simulator into components and create a table containing the input and output variables for each component.

Table I represents the schema of a simulator for which seven components are needed and those components exchange between them eight different variables.

TABLE I
COMPONENTS

Component	Input variables ¹	Output variables ¹
Component 1	V ₁ , V ₂	V ₅
Component 2	V ₁ , V ₂ , V ₄	-
Component 3	V ₃	V ₆
Component 4	V ₃	V ₄
Component 5	V ₄	V ₁ , V ₇
Component 6	-	V ₂ , V ₈
Component 7	V ₄ , V ₅ , V ₆ , V ₇ , V ₈	V ₃

¹ The input and output variables represent the data received by each component and the data emitted by each component, respectively.

--Step 2: Create a representation of the table as an oriented graph in which the nodes are the simulator's components and the edges represent values exchanged between them. This representation corresponds to the situation where for each exchanged value there is a channel allocated by the system.

--Step 3: Unite the edges with the same source and value resulting edges with one source and multiple destinations. After this step the number of communication channels is reduced to the number of different values exchanged by the components, as each component will send each value on

one and only one communication channel.

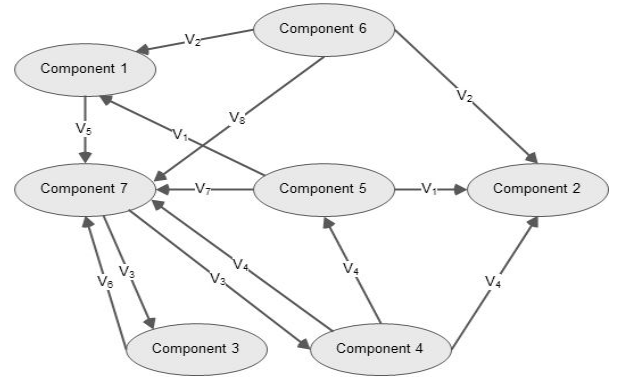


Fig. 3. Simulator components represented as a directed graph in which each node represents a component and each edge represents a value exchanged.

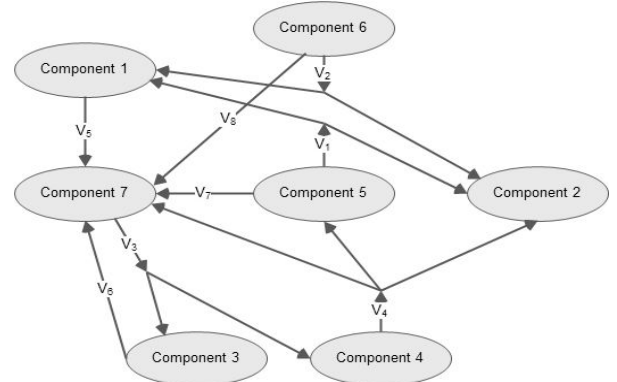


Fig. 4. The edges with the same source and value are united to form links with one source and several destinations.

--Step 4: Unite the edges that have the exact same destinations resulting edges with several sources and destinations.

After the final step of the algorithm the optimum number of communication channels is determined. By using the channels determined by this algorithm it is guaranteed the fact that a component will send a value only once and will receive only the values that it actually needs.

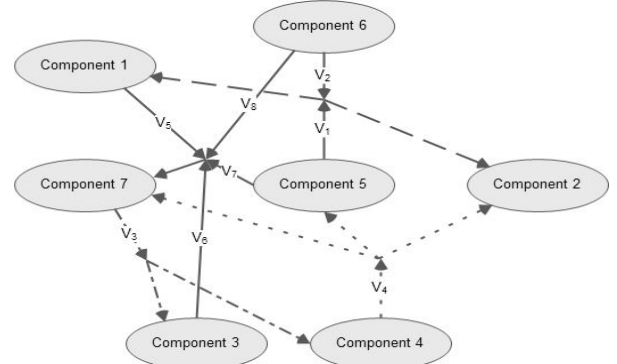


Fig. 5. The edges with the same destinations are united to form links with several sources and destinations. The number of those links represents the optimum number of communication channels.

III. AN EXAMPLE

For the purpose of demonstrating the effectiveness of this algorithm a simulator for a small installation is created. The installation consists of two basins, filled a different quantity of water, connected with a valve.

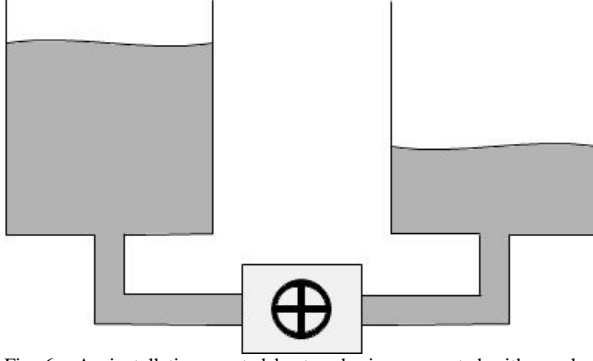


Fig. 6. An installation created by two basins connected with a valve. Initially one basin contains a higher level of water than the other basin.

The first step in creating the modular simulator for this installation is to determine the components involved. The installation in Fig. 4 can be divided in three functional components: basin 1, basin 2 and the valve connecting them. The two basins have the same behavior and thus are instances of the same component. For each of those components there must be determined the input and output variables. To do this will start with the analysis of the physical processes involved.

A. The basin

For the basin we can compute the pressure at the bottom based on the quantity of liquid it contains based on the formula[4]:

$$p = p_a + \rho gh \quad (1)$$

where:

- p : the pressure at the bottom of the basin,
- p_a : the atmospheric pressure,
- ρ : water density,
- g : gravitational acceleration,
- h : the water level.

The water level in the basin will vary based on the quantity received or lost, based on the following equation:

$$h_f = h_i + qs \quad (2)$$

where:

- h_f : the final level of water,
- h_i : the initial level of water,
- q : the quantity of water received or lost,
- s : the surface of the basin.

Based on (1) and (2) it can be determined that, from the functional point of view, for the basin simulator, the input variable is the quantity of water received or lost and the output variable is the pressure at the bottom of the basin.

B. The valve

The valve can be first considered a pipe for which we can compute the volumetric flux using the Hagen-Poiseuille formula[5]:

$$Q = \frac{\pi R^4}{8\mu} \frac{(p_l - p_r)}{L} \quad (3)$$

where:

- Q : the volumetric flux of water (left to right),
- R : the radius of the pipe,
- μ : the water viscosity[6],
- p_l : the pressure at the left end of the pipe,
- p_r : the pressure at the right end of the pipe,
- L : the length of the pipe.

For a very short time interval the quantity of liquid that passes the valve can be computed with the formula:

$$q = Q \times \Delta t \times f \quad (4)$$

where:

- Q : the volumetric flux of water (left to right),
- q : the volume of water that passes the vane,
- Δt : the time interval (very small),
- f : the opening of the valve, in the interval [0,1].

From (3) and (4) we can determine that, from the functional point of view, at constant temperature, for the vane, the input variables are the left and right side pressure, the opening factor and the time interval and, for the output variable, the quantity of water that passes the vane from left to right.

For the system to work the time interval and valve opening factor must be provided. To do this two more components are added to the simulator: the time interval generator and the user interface.

C. The time interval generator

This component is used to generate periodically a time interval, with a very small value, that is used to compute the quantity of water that passes through the valve. To start functioning it needs a start command and it will receive it from the user interface components. This command acts as an input variable for the component, while the output variable is the time interval.

D. The user interface

This component encapsulates a friendly interface that is displayed to the user. In this way the state of the system is

presented to the human operator and he can send commands to the simulator. The input variables for this component are the water level and the pressure at the bottom of the left and right basins, the existence of water in the left and right sides of the valve and the time interval, used to display the state of the time generator component. The output variables are represented by the opening factor of the valve and the command to start the time interval generator.

TABLE II
COMPONENTS OF THE SIMULATOR

Component	Input variables ¹	Output variables ¹
Left basin	q	p_l, h_l
Valve	p_l, p_r, f, T	q, A_l, A_r
Right basin	q	p_r, h_r
Time generator	S	T
User interface	$p_l, p_r, h_l, h_r, A_l, A_r, T$	F, S

¹The variables exchanged within the system are: q : the quantity of water lost or gained by a basin, p_l : the pressure at the bottom of the left basin, h_l : the water level in the left basin, p_r : the pressure at the bottom of the right basin, h_r : the water level in the right basin, f : opening factor of the valve, T : the time interval, A_l : signal the presence of water in the left section of the valve, A_r : signal the presence of water in the right section of the valve, S : the start command for the time interval generator.

After the analysis of the input and output variables for all five components the first step of the algorithm can be applied and the components and variables table created.

By applying the second step of the algorithm a graphical representation of the system is obtained.

In this representation the number of communication channels is 14. The next step of the algorithm can be applied in order to reduce the number of communication channels.

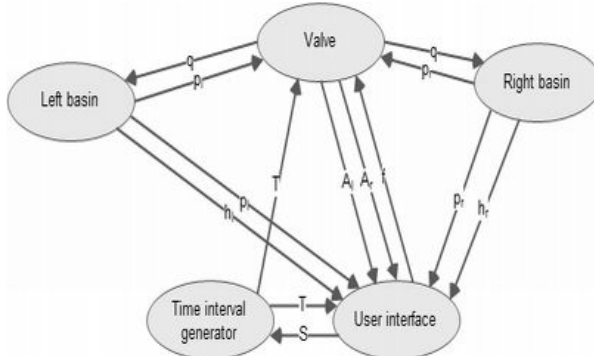


Fig. 7. Simulator components represented as a directed graph in which each node represents a component and each edge represents a value exchanged.

By applying the third step of the algorithm the variable q which was sent to two different modules is now sent on only one channel. The same goes for variables p_l , p_r and T respectively. The number of communication channels is reduced at 10. The last step of the algorithm can now be performed to obtain the optimum number of communication channels.

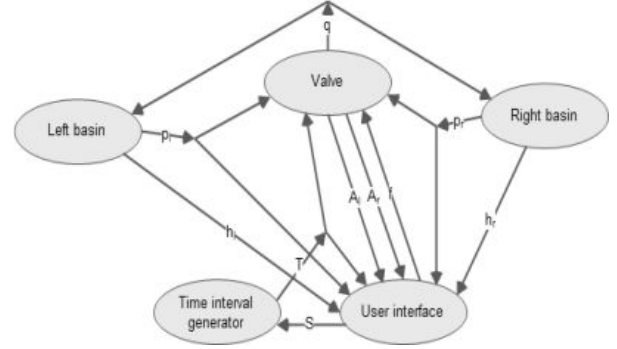


Fig. 8. The edges with the same source and label are merged and the number of communication channels is reduced.

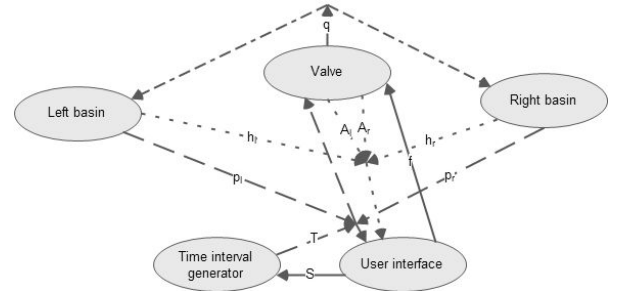


Fig. 9. The edges with the same destinations are merged to obtain the optimum number of communication channels.

From the figure it can be seen that the variables p_l , p_r and T have the exact same destinations, namely the modules “Valve” and “User interface”. Also, the variables h_l , h_r , A_l and A_r have the same destination, the module “User Interface”. This conducts to obtaining the optimum number of communication channels, five, almost 36% of the initial number. In this way no component will send a variable on two difference communication channels and it will never receive a value that it does not need.

TABLE III
COMMUNICATION CHANNELS

Channel	Variables ¹	Source modules	Destination modules
Channel 1	q	Valve	Left basin Right basin
Channel 2	p_l, p_r, T	Left basin Right basin Time generator	Valve User interface
Channel 3	h_l, h_r, A_l, A_r	Left basin Right basin Valve	User interface
Channel 4	f	User interface	Valve
Channel 5	S	User interface	Time generator

¹The variables exchanged within the system are: q : the quantity of water lost or gained by a basin, p_l : the pressure at the bottom of the left basin, h_l : the water level in the left basin, p_r : the pressure at the bottom of the right basin, h_r : the water level in the right basin, f : opening factor of the valve, T : the time interval, A_l : signal the presence of water in the left section of the valve, A_r : signal the presence of water in the right section of the valve, S : the start command for the time interval generator.

IV. CONCLUSIONS

This algorithm has proven to be very useful for creating the schema for the distributed modular simulator. The result of this algorithm always represents the minimum number of channels needed by the system such that no component will send the value of a variable twice nor it receives a variable that it does not need. Starting from the result of this algorithm, the user can minimize the number of communication channels even more at the cost of some redundancy. For example, for the studied installation, channels 4 and 5 can be united, the cost being that the components “Valve” and “Time interval generator” will receive some variables that are ignored. However, since those variables are the result of the user interaction with the system and their value will not be changed very often, the advantage of this modification, that is one less communication channel meaning less processor and memory usage, can be acceptable over the disadvantage that, from time to time, one component will discard a small amount of data.

REFERENCES

- [1] R. M. Fujimoto, *Parallel and Distributed simulation Systems*, New York, John Wiley & Sons, 2000
- [2] B. T. Kulakowski, J. F. Gardner, J. L. Shearer, *Dynamic Modeling and Control of Engineering Systems 3rd ed.*, New York, Cambridge University Press, 2007, ch. 1.
- [3] L. F. Barbulescu, “Functional analysis of a communication framework used in a modular simulator”, in *Proc. of the 14th International Conference on System Theory and Control*, Sinaia, Romania, October 17-19, 2010, pp. 61–66.
- [4] *Fluid statics* [Online], Available: http://en.wikipedia.org/wiki/Fluid_statics.
- [5] *Hagen–Poiseuille equation* [Online], Available: http://en.wikipedia.org/wiki/Poiseuille's_law
- [6] *Viscosity* [Online], Available: http://en.wikipedia.org/wiki/Viscosity#Viscosity_of_water

Current Controlled Optimal Electrical Drive System

C. Botan and F. Ostafi

Abstract— The optimal control from the energetic point of view of the transient state of electrical drive systems is presented. The performance criteria consider only Joule losses, since they significantly overcome other ones in the transient state. The paper refers to a structure with current control of the drive.

I. INTRODUCTION

THE optimal control [1], [2] of the electrical drive systems represents an important way for energy saving and there are numerous studies dedicated to this problem, for different types of motors, control strategies, criteria, or used methods (for instance, we mention [3], [4], [5], [6], [7], but many other papers can be indicated). Moreover, the optimization is appreciated as a main direction of the developing of the electrical drive systems in the future [8].

It should be also noticed that the optimal control is useful not only for energy saving, but in many cases offers the possibility to reduce the motor rated power and therefore the weight and volume. Indeed, the motor rated power is chosen from heat consideration and the optimal control leads just to a diminished Joule losses. Moreover, the decrease of the weight of certain sub-ensemble leads in certain applications to the decrease of the energy consumption of the all plant.

In spite of the mentioned advantages, the number of applications is nowadays small and we appreciate that a cause of the reluctance in this direction is the complexity of the algorithms. Taking into account this fact, we consider that simple implementable solutions for optimal control are necessary. The previous results obtained by authors indicate that it is possible to obtain a significant diminution of the energy losses in the motor windings in the electromechanical transient process using simple implementable optimal controllers.

It is important in many applications to achieve a control which ensure small energy losses and, at the same time, an acceptable behaviour of the drive system. However, the demands for different applications are not the same and certain differences in the adopted optimization criteria can occur. For instance, the minimization of the global power

losses is a goal for the steady state operation and the establishing of an adequate flux level is the most used procedure in this direction. Although a similar criterion can be also adopted for the transient period (but referring to energy and not to power), one has to take into account that the Joule losses significantly overcome all other losses, since the start currents have very great values. Another aspect which must be considered refers to the control structure of the drive system: with current or with voltage control, depending on the type of the power electronic converter. The presence of an inner current control loop indicates that such a structure can be assimilated to a current controlled one.

It is possible in many cases to adopt one or two control variables. The last variant is useful for optimal control if the drive operates in many situations with a reduced load and it is compulsory in the optimal control of the steady state.

The authors have presented in some previous papers different variants for optimal control problems for transient period of the electrical drives. For instance, the voltage control case was studied in [9] and the current control variant is presented in [7] and [10]. In this last case, the optimization performance index contains only the terms referring to the cooper losses. Two interesting results, valid for different motor types, were obtained in this case for some reasonable assumptions:

- the minimum cooper losses are obtained when the electromagnetic torque is twice the load one;
- if two control currents (or two current components) are used, the corresponding cooper losses for these currents must be equal.

In addition, an important advantage of the proposed current optimal control is a very simple implementation, based on a usual cascade structure. An inconvenient of this optimal control is the long time transfer when the load torque is small. The limitation of the time leads to the increase of the cooper losses in comparison with the unconstrained optimal control. The present paper indicates an alternative for this situation: the introducing of a term depending on the speed error in the performance index. The advantage of the easy implementation is maintained in this case.

Only the case of one control variable is considered. The results are valid for different motor types, because the mathematical model is the same with adequate assumptions.

C. Botan is with the Technical University of Iasi, Romania (corresponding author to provide phone: +40232 278683; fax: +40232 230751; e-mail: cbotan@ac.tuiasi.ro).

F. Ostafi is with the Technical University of Iasi, Romania (phone: +40232 278683; fax: +40232 230751; e-mail: fostafi@ac.tuiasi.ro).

II. MAIN OPTIMALITY CONDITIONS

The mechanical equilibrium equation of the drive system is

$$\dot{\omega}(t) = [m_e(t) - m(t)] / J, \quad (1)$$

where $\omega(t)$ is the speed, J is the inertia, $m_e(t)$ is the electromagnetic torque and $m(t)$ is the load torque (the dependence of m on the speed is neglected).

The state variable $\omega(t)$ results from (1) and, for $t = T$, one obtains

$$\omega_d = \omega(T) = \omega(0) + T(m_{em} - m_m) / J, \quad (2)$$

where

$$m_{em} = \frac{1}{T} \int_0^T m_e(t) dt \text{ and } m_m = \frac{1}{T} \int_0^T m(t) dt \quad (3)$$

are the mean values of the electromagnetic and of the load torque, respectively.

The relation (3) shows that the proposed control problem can be solved if the mean value of the load torque is beforehand known. At least the shape of the load torque must be known and its amplitude will be estimated at the beginning of the optimal control process.

The electromagnetic torque of the (non-saturated) motor with one control variable is considered to be in the form

$$m_e(t) = ci, \quad (4)$$

where c is a constant of the machine and i is the rotor current in the d.c. with constant flux case, and it is the i_q component of the current in a vector controlled a.c. machine case (the component i_d is supposed to be constant).

The system equation results from (1) and (4) in the form

$$\dot{\omega}(t) = bi(t) - \mu(t), \quad (5)$$

where

$$b = c/J \text{ and } \mu(t) = m(t) / J. \quad (6)$$

The adopted performance criterion is

$$I = \int_0^T [i^2(t) + q^2(\omega(t) - \omega_d)^2] dt, \quad (7)$$

where ω_d is the desired final value of the speed $\omega(t)$ and

q is a weight coefficient for the speed error. Of course, the first term in (7) corresponds to the energy losses. An optimal control problem containing only the first term in (7) will be referred in the sequel as a minimum energy (ME) problem. The paper refers especially to a combined optimal criterion (COC), when the criterion (7) contains both terms.

The optimal control problem is to find the control variables $i(t)$ which transfer the system (5) from the initial state $\omega(0)$ (for simplification, one consider $\omega(0) = 0$) to the final state $\omega_d = \omega(T)$ so that the performance index (7) to be minimized.

The Hamiltonian [1], [2] of the problem is

$$H = i^2(t) + q^2(\omega_d - \omega(t))^2 + \lambda(bi(t) - \mu(t)), \quad (8)$$

where $\lambda \in \mathbf{R}$ is a co-state variable.

The necessary optimality conditions [1] $\partial H / \partial i = 0$ and $\partial H / \partial \omega = -\dot{\lambda}(t)$ lead to

$$i(t) = -\frac{b}{2} \lambda(t) \quad (9)$$

and

$$\dot{\lambda}(t) = -2q^2(\omega(t) - \omega_d). \quad (10)$$

The solution to the system (5) and (10) for a constant load torque is:

$$\omega(t) = C_1 e^{qbt} + C_2 e^{-qbt} + \omega_d, \quad (11)$$

$$\lambda(t) = \frac{2q}{b} C_1 e^{qbt} - \frac{2q}{b} C_2 e^{-qbt} - \frac{2}{b^2} \mu. \quad (12)$$

The terminal conditions $\omega(0) = 0$ and $\omega(T) = \omega_d$ lead to

$$C_1 + C_2 = -\omega_d, \quad (13)$$

$$C_1 e^{qbT} + C_2 e^{-qbT} = 0.$$

The constants C_1 and C_2 result immediately from (13).

The optimal control variable results from (9) and (12):

$$i(t) = -C_1 q e^{qbt} + C_2 q e^{-qbt} + \frac{1}{b} \mu. \quad (14)$$

The desired velocity ω_d is achieved at the final moment

T and for $t > T$ it has to maintain this value. Therefore, the acceleration must be zero, i.e. the control law becomes

$$i = m / c . \quad (15)$$

If the final moment T is not imposed, it is useful to choose an optimal value for T . The corresponding transversality condition [1] can be used for this aim:

$$H(\omega(T), i(T), \lambda(T)) = 0 . \quad (16)$$

Taking into account that $\omega(T) = \omega_d$, the condition (16) leads to

$$i(T) = 2\mu / b . \quad (17)$$

It results finally from (14) and (17) the optimal value of the final time:

$$T^* = \frac{1}{bq} \ln \frac{\mu}{2bq\omega_d} . \quad (18)$$

Remark 1: The optimal control in the ME problem is obtained [7] when

$$m_e = 2m_m , \quad (19)$$

or, equivalently,

$$i = 2\mu / b , \quad (20)$$

where m_m is the mean value of the load torque on the interval $[0, T]$. For this constant value of the current, the speed has a linear variation and the final value ω_d is achieved in the time

$$T = J\omega_d / m_m . \quad (21)$$

Therefore, the final time results implicitly in the ME problem and it is not possible to choose or impose its value. If this value is not convenient (especially for small m_m), a supplementary restriction have to be introduced. Of course, the energy losses increase in this case.

Remark 2: The results for COC problem are closely related to ones of the ME problem if the weight coefficient q has a small value. Another correlation between the two problems is the fact that final value (17) of the control variable in the COC problem is the same as the constant value (20) of the same variable in the ME problem. This implies greater values for current on the interval $[0, T]$ (see, for illustration, the presented results in the next section). This

fact has the following explanation: the drive system is forced to diminish faster the speed error and therefore the time transfer is smaller, but the energy losses increase.

III. SYSTEM IMPLEMENTATION AND SIMULATION RESULTS

The above established control is an *ideal optimal control*. We say that the control is ideal because of the assumption referring to the currents: namely, we supposed these could have abrupt, non-inertial variations.

The fact that we know the ideal optimal values of the currents suggests an easy implementation: the controller must ensure the desired value of the currents with a great accuracy. A better accuracy ensures a better proximity to ideal optimal control. Of course, a task for the load torque estimation and for computing of the desired current values has to be introduced in the control algorithm.

There are different ways for obtaining the proper optimal controller, described above. In this respect we shall present only a structure based on a cascade control with PI controllers: the output of the speed controller is transmitted to the input of the current controller via the optimal control (OC) block. If the speed error is very small, the transfer is direct ($i^* = i^0$), otherwise the optimal control i^* must be computed with (14) for $t \leq T$ and (15) for $t > T$.

This specific part (the internal loop of the cascade) is presented in Fig.1, where OC represents the optimal controller block, described above, C is a PI controller, P is the controlled plant, and i^0 represents the output of the external (speed) controller.

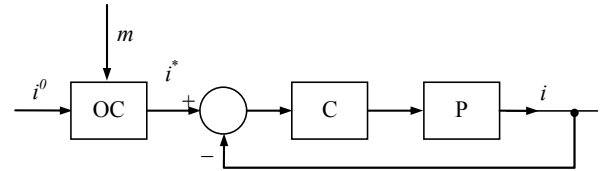


Fig. 1 Structure for suboptimal current control

The figures 2,...,7 present a selection of simulation tests, performed for the proposed structure of a drive system for different operation conditions. We have supposed in all cases that the load torque is known. The drive system has a motor with rated current 3.3A, rated torque 0.78 Nm and the inertia $J=0.028$ Nms/rad. In all cases, ω is expressed in rad/s, i in A, m in Nm, and q in As/rad.

The presented figures indicate the behaviour of the optimal drive system for different load torques, values of the final time and weight coefficients q in the criterion..

The current has unacceptable high values for q greater then 0.5 As/rad. Such situations are not presented in figures. For small values of the coefficient q , the behaviour is closely related to one corresponding to ME problem. For instance, one can remark in Fig. 2 that the current is almost constant on

the interval $[0, T]$ and the speed has an almost linear variation, as in ME problems (see, (20)). The currents and energy losses are smaller for low values of q . On the other hand, the duration of the acceleration period is bigger in these cases (see, for instance, Fig. 3 and 4).

The performance index was computed in each case. From practical point of view is more interesting to compare the energy losses in different cases. Of course, the energy losses are minimal in ME problem, but the acceleration time can be too big for small load torque. If a smaller final time T is imposed, the losses increase. For the same T , the losses in COC problem are with $(0.4...4)\%$ greater for $q = 0.1$ and with about 20% greater for $q = 0.4$.

If the transient period is forced and the final time is imposed to be smaller then the corresponding optimal value T^* , the control current increases; this fact can be observed comparing Fig. 2 with 4 and Fig. 6 with 7.

The increase of the load torque leads, as it is expected, to an increase of the current (see, Figs. 6 and 7 in comparison with Figs. 5 and 3, respectively). However, the difference between the two cases is not so big, corresponding to the difference of the loads. The biggest part of the current and therefore, of the losses corresponds to transient (accelerating) component.

The performed simulations showed that the energy losses can be diminished with $(5,...,30)\%$ in comparison with a usual cascade control structure (the greater reductions are obtained for small load torques).

Summarizing, one can remark that the implementation of the ME problem is more advantageous from the energetic and acceleration time points of view, even if the time transfer is restricted. If the weight coefficient q in the COC problem is small, the performances in the two problems are similar.

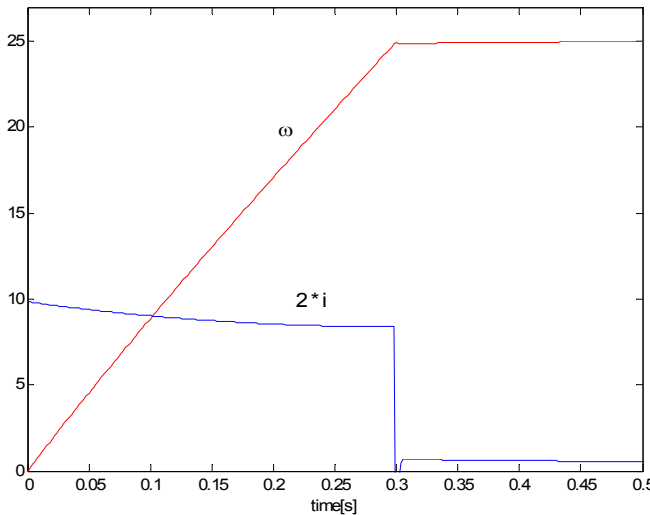


Fig. 2. The behaviour of the optimal system for $m=0.15$, $q=0.1$, $T=0.3$

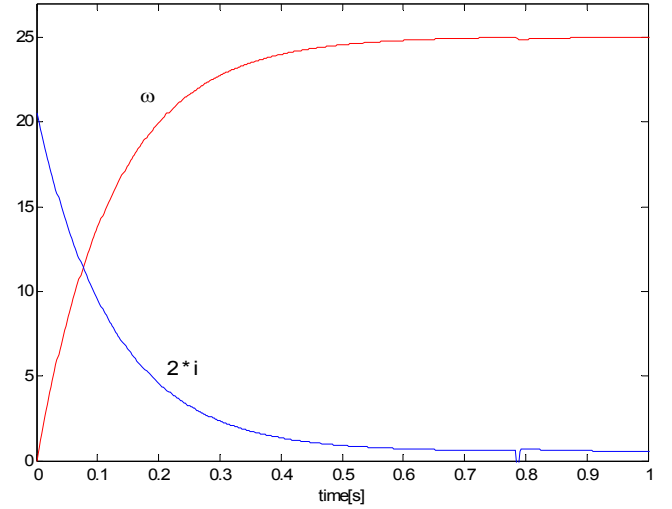


Fig. 3. The behaviour of the optimal system for $m=0.15$, $q=0.4$, $T=T^*=0.78$

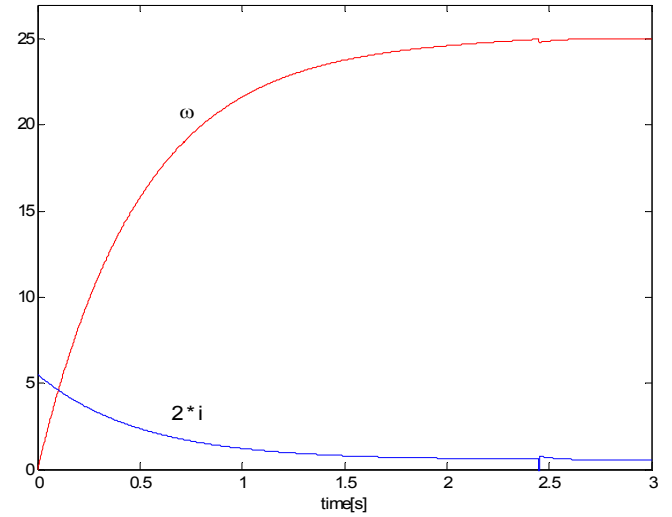


Fig. 4. The behaviour of the optimal system for $m=0.15$, $q=0.1$, $T=T^*=2.45$

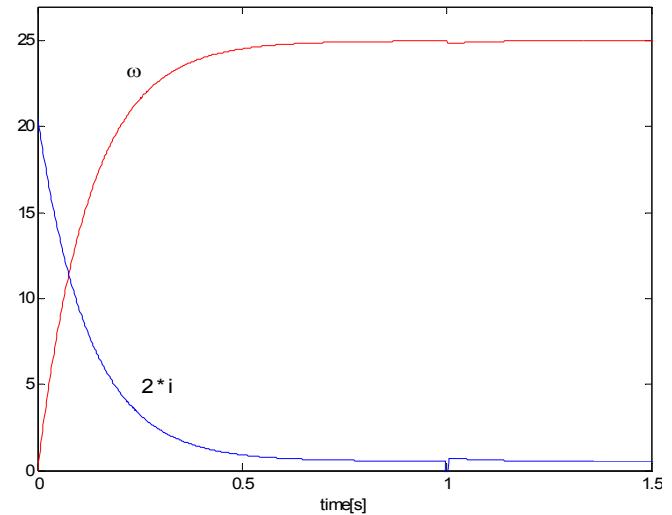


Fig. 5. The behaviour of the optimal system for $m=0.15$, $q=0.4$, $T=1$

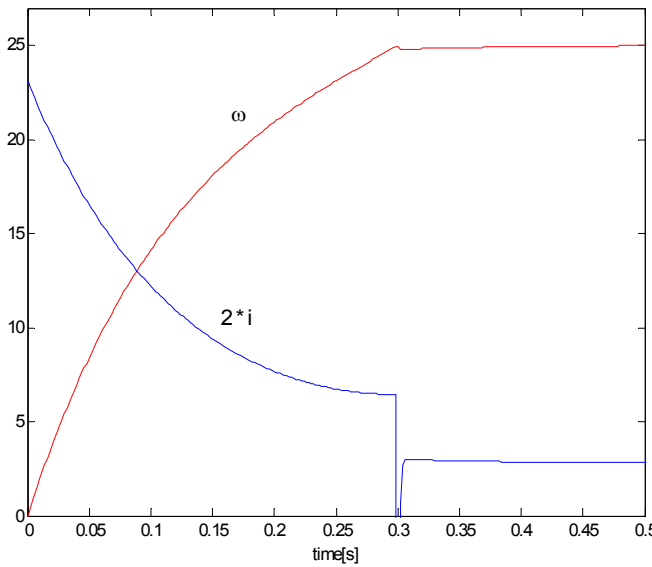


Fig. 6. The behaviour of the optimal system for $m=0.78$, $q=0.4$, $T=0.3$

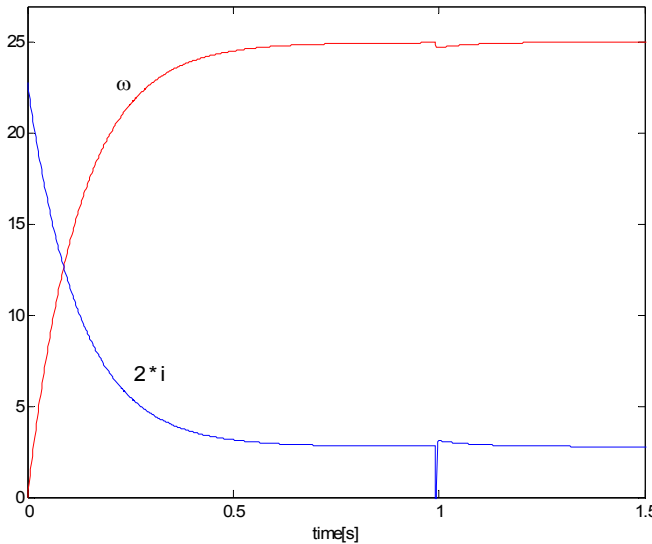


Fig. 7. The behaviour of the optimal system for $m=0.78$, $q=0.4$, $T=T^*=0.99$

IV. CONCLUSION

The optimal control of the electrical drives has the doubtless advantages from energetic point of view in comparison with other control strategies. In addition, a good behaviour in the transient state can be achieved.

The paper presents a simple implementable structure for optimal control of electrical drives with different motor type. This structure is based on a cascade one with certain modifications.

The proposed strategy uses one control variable – a current or a current component in $d-q$ frame. A quadratic performance index referring the copper energy losses and speed error is adopted.

The performed simulation tests indicate a decrease of the energy losses and a good behaviour of the drive system. However, the behaviour of the system which minimizes only the energy losses (even if the transfer time is restricted) is more convenient from different points of view. Only for a small weight of the speed error in the criterion, the behaviours in the two cases are similar.

REFERENCES

- [1] B.D.O. Anderson and J.B. Moore, *Optimal Control. Linear Quadratic Methods*. Prentice Hall, 1990.
- [2] M. Athans and P.L. Falb, *Optimal Control*. New York: Mc Graw Hill, 1966.
- [3] C. De Capua and C. Landi, "Measurement Station Performance Optimization for Testing on High Efficiency Variable Speed Drives," *IEEE Trans. on Instrumentation and Measurement*, vol. 48, no. 6, pp. 1149-1154, 1999.
- [4] B. Canudas de Wit and J. Ramirez, "Optimal torque control for current-fed induction motors", *IEEE Trans. Automat. Contr.*, vol. 44, pp. 1084-1089, May 1999.
- [5] G. Acampora, C. Landi, M. Luiso, and N. Pasquino: "Optimization of Energy Consumption in a Railway Traction System," in *Proc. of International Symposium on Power Electronics, Electrical Drives, Automation and Motion, SPEEDAM 2006*, Taormina, Italy, 2006, CD-ROM.
- [6] I.T. Wallace, D.W. Novotny, R.D. Lorenz, and D.M. Divan, "Verification of Enhanced Dynamic Torque per Ampere Capability in Saturated Induction Machines," *IEEE Transactions on Industry Applications*, vol. 30, no. 5, pp. 1193-1201, Sept./Oct. 1994.
- [7] C. Botan, V. Horga, F. Ostafi, M. Albu, M. Ratoi, "General Aspects of the Electrical Drive Systems Optimal Control," in *Proc. of 12th European Conf. on Power Electronics and Applications, EPE 2007*, Aalborg, Denmark, Sept. 2007, CD-ROM.
- [8] R.D. Lorentz, "Future Motor Drive Technology Issues and their Evolution," in *Proc. of 12th Int. Power Electronics and Motion Conference, EPE-PEMC 2006*, Portoroz, Slovenia, 2006, CD-ROM.
- [9] A. Onea and C. Botan, "Real Time LQ-Optimal Control of a Drive System," in *Proc. of IEEE – IECON'98*, Aachen, Vol. 3, Sept. 1-4 1998. pp. 1417-1421.
- [10] C. Botan, V. Horga, "Optimal control of the electrical drives with induction motors," in *Proc. of International Symposium on Industrial Electronics, ISIE 2008*, IEEE Catalog Number CFP08ISI-CDR, Cambridge, UK, July 2008, CD-ROM.

Design and Control of a Biped Robot

Alexandru Ionut Botas and Razvan Solea, *Member, IEEE*

Abstract—This paper aims to present the practical realization of a biped robot and its remote control via bluetooth communication using C++ language. With a height of about 25 centimeters, the robot has six degrees of freedom, given by the six servomotors that are used to move it. Dimensions and the small number of motors are an advantage over conventional robots, because it does not need external energy sources and big hardware or software resources. The design allows the robot's straight movement and turning left or right. The presence of infrared sensors allows the robot to avoid static obstacles and stop at the edges of the surface displacement to avoid any accidents. The robot is able through an infrared proximity sensor used to avoid obstacles, to achieve grayscale images of objects is used a line-by-line scanning algorithm. The software allows the live creation of the images pixel by pixel. Those images can be stored or used to identify shapes or objects.

I. INTRODUCTION

Since ancient times, mankind has tried by every method to recreate the complex mechanism that makes up the human body. Natural bipedal locomotion involves a large number of degrees of freedom. Human locomotion involves more than 200 degrees of freedom [1]. Beside this, there are many problems that involves kinematics, dynamics, balance, stability and many others. All this makes the study of bipedal robot a complex subject and research often lead to not entirely satisfying results. However, with technological development based on theoretical and experimental research, the man managed to copy or imitate some parts of the body.

So far the best known and most advanced humanoid robot in the world is Honda's ASIMO. In addition to the advanced control of locomotion that allows it to walk, run, climb stairs and avoid both static and dynamic obstacles while walking ASIMO is equipped with voice recognition algorithms and visual recognition of gestures. In fact the main characteristic of studying humanoid robots is locomotion. Only after they gain a natural ability to walk into a human specific environment the studies will focus on socialization and interaction with people [2].

In this work we aimed to build the lowerside, the locomotion part of a biped robot and its control so that it will be able to walk, avoid static obstacles and provide information on the environment in which it is located (see Fig. 1).

This work was supported by CNCIS-UEFISCSU, Romania, project PNII-IDEI 506/2008

A.I. Botas is student at the Faculty of Automatic Control, Computers, Electric and Electronic Engineering, Dunarea de Jos University of Galati, Romania botas.ionut@yahoo.com

R. Solea is with the Faculty of Automatic Control, Computers, Electric and Electronic Engineering, Dunarea de Jos University of Galati, Romania razvan.solea@ugal.ro

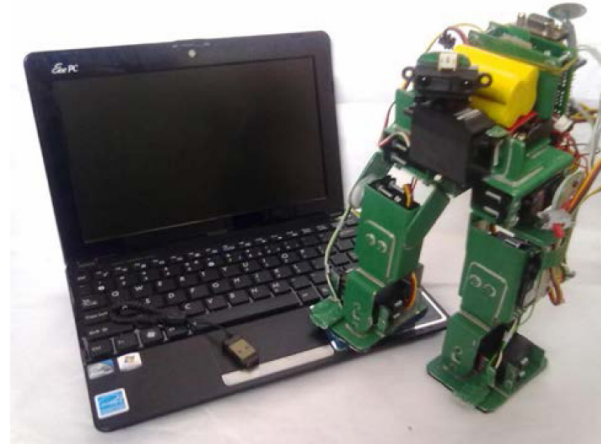


Fig. 1. Hardware equipment

II. DESIGN AND MOTION PLANING

Based on the analysis of the lower part of the human body it has been chosen the minimum number of degrees of freedom that can ensure the robot locomotion [3]. Totalizing six, these correspond according to Fig. 2 to the Y axis rotation of the ankles and to the X axis rotation of the knees and the whole leg from the hip. The small number of degrees of freedom is imposed by the size of the power source and the number of actuators — a large number of actuators means a large power source or even an external one which prevents free movement.

Biped walking is actually a repetition of steps. Therefore it is possible to separate and identify the movements that compose repetitive step. Based on those movements the actuators state for various positions are identified [4].

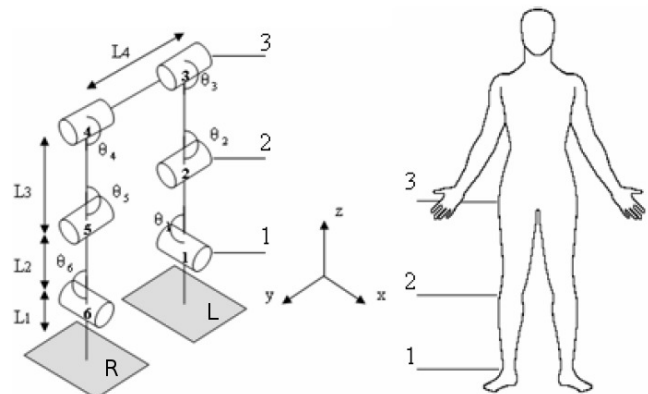


Fig. 2. Human-robot correspondence

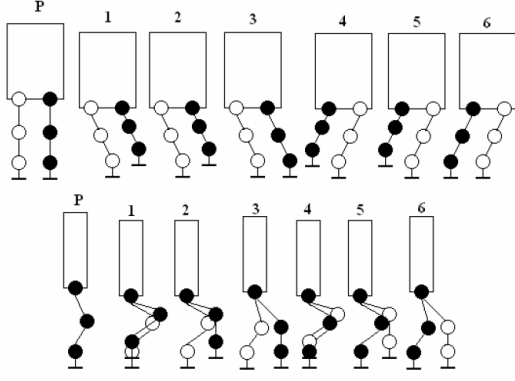


Fig. 3. Basic movements that compose a forward walking step

Forward walking consists in the repetition of six basic movements that compose a step. In Fig. 3 the partial movements that comprise a step for forward walking (front and side view) starting with the parking position P , the movement is possible by repeating steps 1–6 in this order.

The parking position P represents the initial position where the steady state is maximum. This position will be used for the cases in which the robot will scan the environment to identify any obstacles or to provide images or to identify shapes with known properties.

For the right and the left turns it is necessary for the robot to start from the parking position P and to reach to the same parking position P , this position being the transition between the different forward movements, the right or left turns. The same as the forward walking the turns are executed by repeating movements 1, 2 and 3 for the right turn and movements 4, 5 and 6 for the left turn, these being situated between two different P parking positions. The Figs. 4, 5 show the movements that compose the right and the left turns.

III. KINEMATIC MODEL

Based on the above planning movements the kinematic model can be deduced. Kinematic analysis is based on the basic equation of the geometric model that consists in determining the position and orientation of a point compared

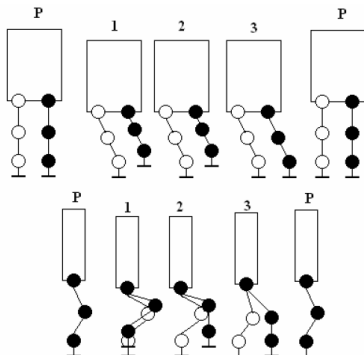


Fig. 4. Movements that compose a right turn

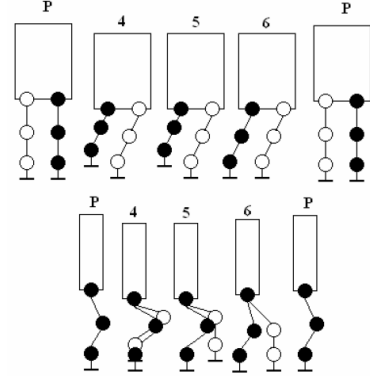


Fig. 5. Movements that compose a left turn

to a reference (all those characteristics are called kinematic attitude or configuration of the robot end effector at every moment of time [5]) when are known the relative coordinates (attitudes) of all segments of the kinematic chain that compose the robot, at that time. Therefore for any moment of time is considered that as the reference, the part of the robot that has contact with the ground. They were marked with X_l , Y_l , and Z_l the left foot position and with X_r , Y_r , and Z_r the right foot position. Thus in the parking position we can say that $X_l = X_r$, $Z_l = Z_r$ and $Y_l = Y_r + L_4$.

Because links 1 and 6 rotates only on the O_y axis and links 2, 3, 4 and 5 rotates only on the O_x axis the angles that correspond to those links will determine separately the coordinates Y_l , Y_r respectively X_l and X_r . Therefore we will consider in determining the X coordinates that the left foot is the reference, this being shown in Fig. 6.

The equation which calculates the current position X_r is:

$$X_r = L_3 \cdot \cos\theta_2 + L_3 \cdot (\cos\theta_4 - \cos\theta_3) + L_2 \cdot \cos\theta_5 \quad (1)$$

By analogy we have:

$$X_l = L_3 \cdot \cos\theta_5 + L_3 \cdot (\cos\theta_3 - \cos\theta_4) + L_2 \cdot \cos\theta_2 \quad (2)$$

To determinate the Y_l and Y_r coordinates we will consider the same case in which the left foot will be the reference, case presented in Fig. 6. Y_r are current value will be given by the next equation:

$$Y_r = L_4 \cdot \cos\theta_1 + L_1 \cdot \cos\theta_6 \quad (3)$$

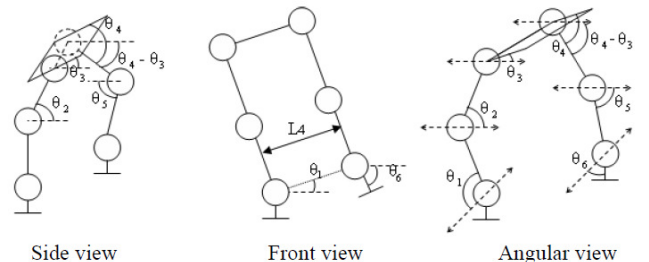


Fig. 6. Biped locomotion

By analogy we have:

$$Y_l = L_1 \cdot \cos\theta_1 + L_4 \cdot \cos\theta_6 \quad (4)$$

For the Z_r and the Z_l coordinates we must take into consideration all the links and as in the previous cases in order to calculate those coordinates we consider the reference as being the left foot, case presented in Fig. 6. The equation for calculating the current value of the Z_r coordinate will be:

$$Z_r = L_1 + L_2 \cdot \sin\theta_1 + L_3 \cdot \sin\theta_2 + L_3 \cdot \sin(\theta_4 - \theta_3) + L_2 \cdot \sin\theta_5 + L_1 \cdot \sin\theta_6 \quad (5)$$

By analogy we have:

$$Z_l = L_1 + L_2 \cdot \sin\theta_6 + L_3 \cdot \sin\theta_5 + L_3 \cdot \sin(\theta_3 - \theta_4) + L_2 \cdot \sin\theta_2 + L_1 \cdot \sin\theta_1 \quad (6)$$

IV. SOFTWARE AND HARDWARE EQUIPMENT

SSC-32 [6] is a pre-assembled servocontroller with high resolution ($1\mu s$) for precise positioning and smooth movement of the servos. This controller used an 8 bit low cost Atmega168-20PU microprocessor and is able to generate PWM signals with a width varying between $0.5ms$ and $2.5ms$ to get rotations between 0° and 180° . The servomechanisms can be controlled in the sense of immediate response, variable speed, variable time or combination of these. A special command allows a group of motors to begin and complete the movement at the same time even if their range of motion differ. This represents a real advantage in creating controls for robots that walk using multiple servomotors.

The servomotors used in this project are HS311-Standard from Hitec because these are able to develop a high torque of $4.5Kg/cm$ at a supply voltage of $6V$ being capable to support the weight of the entire robot. The servomotors are controlled by width variable rectangular pulses known in the specialty literature as PWM signals (pulse width modulated signals). The characteristics of these pulses are the minimum width, maximum width and repetition rate. Being given the constraints of rotation of the servo the neutral position is defined as the command pulse for the clockwise rotation is the same as the command pulse for the counterclockwise rotation. The angle of rotation is determined by the width of the PWM control signal (Fig. 7). For operating the servomotors needs the signal to be repeated every $20ms$. The pulse width determine the servos position. For example a signal with width of $1.5ms$ will cause the motor to rotate in the 90° position, known the neutral position.

The control software (Fig. 8) has been created with Visual Studio. NET 2003 [7] in C++ and contains a graphic user interface through which you can select the control options and in which you can view the information returned by the robot in the form of grayscale images. In the test phase it was used SSC32Poser software [8]. This is a special software developed in Visual Basic. NET that allows SSC-32 controller both reading data from inputs and outputs triggering.

To control the inputs and the outputs SSC-32 controller receives and responds to commands on a serial line compatible

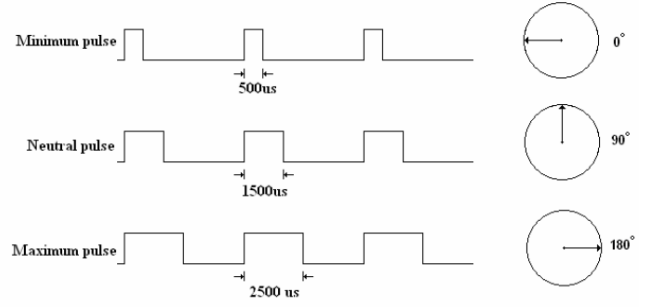


Fig. 7. PWM control signals

with both TTL logic levels as well as for standard RS232 communication. To transmit data between the host computer, on which will run the command software and the SSC-32 controller an OmniTeck CSR BlueCore4 Bluetooth [9] module was used in the transparent transfer of data working mode and a USB-Bluetooth [10] adapter from miniNova like in Fig. 9.

V. OBSTACLE AVOIDANCE

Sharp GP2Y0A21YK [11] is an infrared sensor used to measure distances on a range between $10cm$ and $80cm$ [12]. It is an analog sensor, which means that for a certain distance from that range it returns a voltage between $0V$ and $3V$. This voltage values are converted by the Atmega168-20PU [13] microprocessors Analog to Digital converter and returned on request on the serial line in integer value between 0 and 127. So these integer values correspond to a distance between the scanned item and Sharp sensor.

In the case of obstacles avoidance a horizontal scan is performed using a servomotor and the scan results are stored in an array to determine the optimal direction for turning. This is done by comparing the two values L_{left} and L_{right} representing the distance between the obstacle and the limits of the scanned area, for turning being chosen the direction corresponding to the greatest of them (Fig. 10).

In the case of edge detection are used two QRD1114 optical sensors from Fairchild [14] which operates on the same principle as the distance detection sensor from Sharp

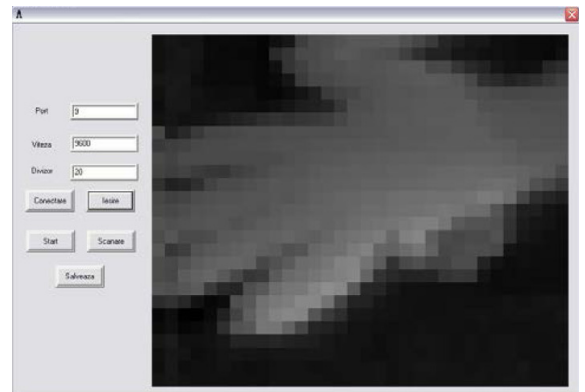


Fig. 8. Biped Robot Graphic user interface

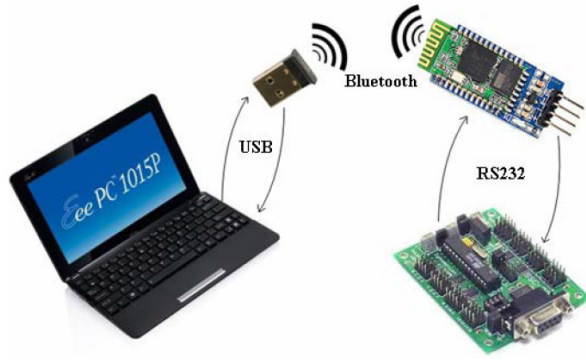


Fig. 9. Communication

but the detection range is between 0mm and 250mm and the returned analog voltage is between 0V and 5V.

VI. INFRARED 3D SENSOR SCAN

To receive information about the environment in which the robot operates the Sharp sensor is used. A line-by-line scanning is performed in order to create grayscale images of the objects that are near the robot. Scanning can be done with variable step fact that is reflected in various resolutions of the created images and different scanning times. Distance values are converted to grayscale values and the created image preview can be done in real time pixel by pixel in the window of the robot control program (Fig. 11).

Because scanning is done both vertical and horizontal the distance from the different characteristics of the scanned object varies according to the size of the step and the angle of scanning. This is calculated using the following formula:

$$Length = d \cdot \cos \left(\alpha \cdot \left| \frac{N}{2} - i \right| \right) \cdot \sin \left(\beta \cdot \left| \frac{M}{2} - j \right| \right) \quad (7)$$

where:

- N – is the number of steps on horizontal
- M – is the number of steps on vertical
- i – is the current step on horizontal
- j – is the current step on vertical
- α – is the horizontal scanning angle
- β – is the vertical scanning angle

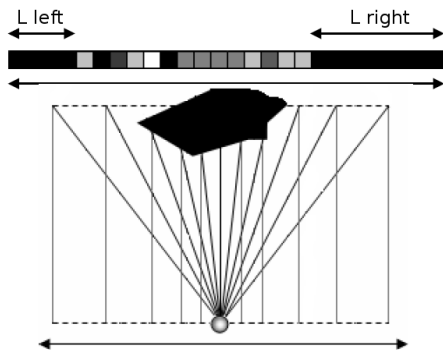


Fig. 10. Horizontal scan and the result in the obstacle avoidance case

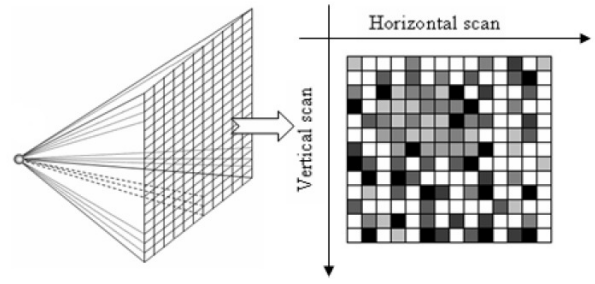


Fig. 11. 3D Scanning and distances to image conversion

$$d = \begin{cases} D_{max}, & \text{if } d_{measured} \geq D_{max} \\ d_{measured}, & \text{if } d_{measured} < D_{max} \end{cases} \quad (8)$$

In the Fig. 12 are presented some results of the 3D scanning of the same object at different resolutions.

This is a low cost method compared with the acquisition of images using a camera. Another advantage of this method is the possibility of acquiring images in the absence of the light and the easier processing of data. The information obtained can be used to identify shapes or objects. A classical application is that of finding the battery charger from multiple objects by knowing its characteristics.

VII. EXPERIMENTAL RESULTS

Figures 13, 14, shows the commands that are sent to the motors of the biped robot for forward walking.

In Fig. 15 are represented several captures images taken during the real experiments. This snapshots show the partial movements that comprise a step for forward walking.

VIII. CONCLUSIONS

In this work we aimed to build the lowside, the locomotion part of a biped robot and its control so that it will be able to walk, avoid static obstacles and provide information

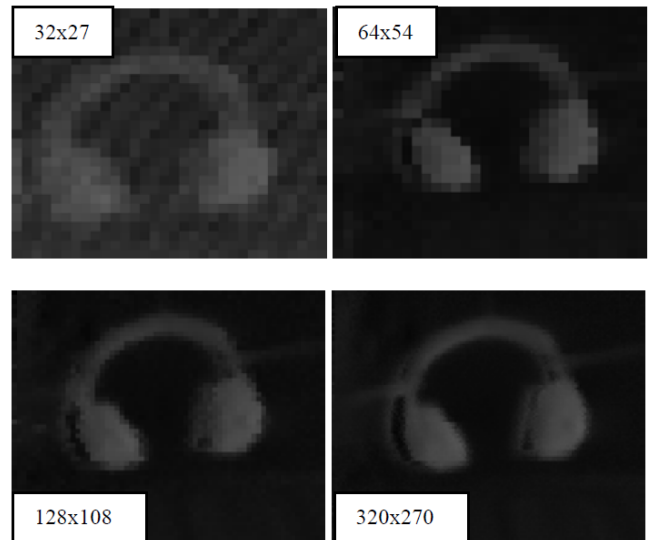


Fig. 12. The result of scanning the same object on different resolutions

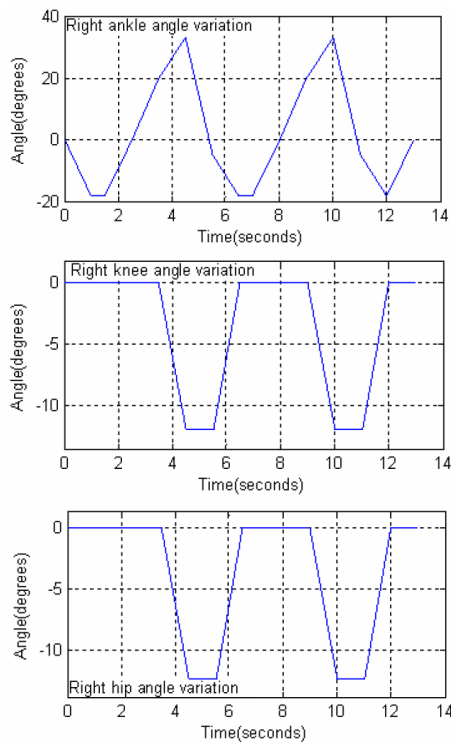


Fig. 13. Servomotors angle variation

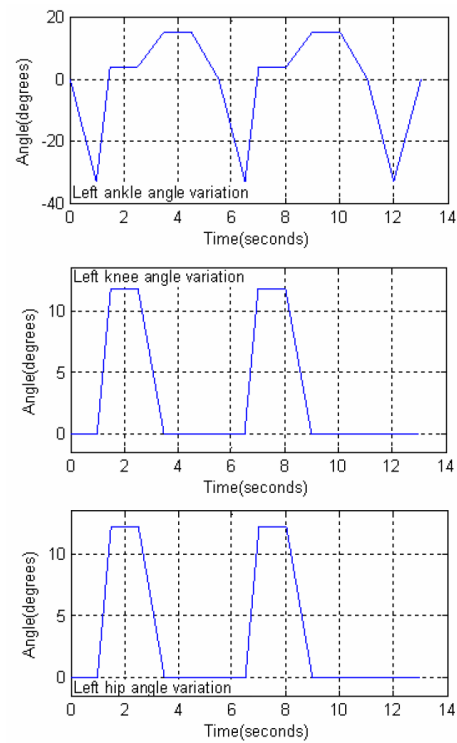


Fig. 14. Servomotors angle variation

on the environment in which it is located. The design allows the robot's straight movement and turning left or right. The presence of infrared sensors allows the robot to avoid static obstacles and stop at the edges of the surface displacement to avoid any accidents.

The robot is able through an infrared proximity sensor used to avoid obstacles, to achieve grayscale images of objects is used a line-by-line scanning algorithm. The software allows the live creation of the images pixel by pixel. Those images can be stored or used to identify shapes or objects. The advantage of this method is the possibility of acquiring images in the absence of the light and the easier processing of data.

IX. ACKNOWLEDGMENTS

This work was supported by CNCSIS-UEFISCSU, Romania, project PNII-IDEI 506/2008.

REFERENCES

- [1] A.Carlos de Pina Filho, *Biped Robots*, InTech, ISBN 978-953-307-216-6, 2011.
- [2] P.D.D. Teodoro, *Development of a simulation environment of an entertainment humanoid robot*, Master Dissertation, 2007.
- [3] F. R. Sias Jr. and Y. F. Zheng How many degrees-of-freedom does a biped need?, in *IEEE International Workshop on Intelligent Robots and Systems IROS '90*, 1990, pp.297-302.
- [4] S. Oh, S. Lee and K. Kim, Design of a Biped Robot Using DSP and FPGA, in *International Journal of Control, Automation, and Systems*, Vol. 1(2), 2003, pp.252-256.
- [5] S. Mcu . *Mechanics and Robots*, Course, Dunarea de Jos University of Galati.
- [6] J. Frye, *SSC-32 Manual*, V2.01XE, 2009.
- [7] <http://msdn.microsoft.com/en-us/visualc/default.aspx>
- [8] <http://www.otherrobots.com/Tutorials/SSC32ServoControllerSoftware/tabid/58/Default.aspx>
- [9] <https://www.bluetooth.org/apps/content>
- [10] *CSR BlueCore4 Bluetooth Module*, Manual Rev2.0, 2005.
- [11] *Sharp GP2Y0A21YK*, Datasheet.
- [12] <http://www.acroname.com/robotics/info/articles/sharp/sharp.htm>
- [13] *Atmega168-20PU*, Datasheet.
- [14] *Fairchild QRD1114*, Datasheet.

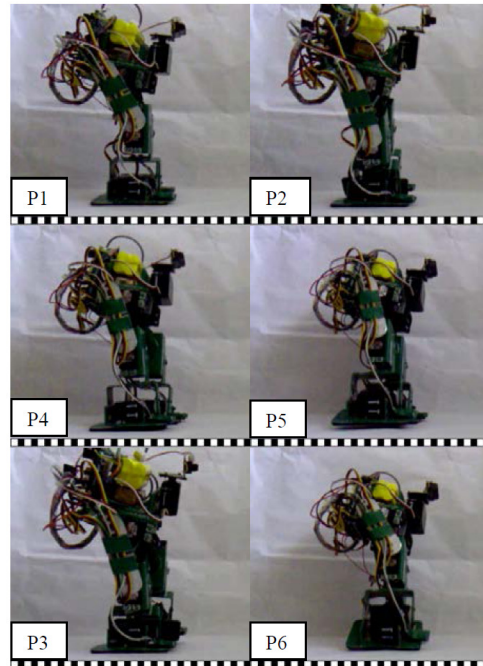


Fig. 15. Snapshots during one step walking

A Clustering Algorithm for Negotiating Security in Wireless Mesh Networks

N. Botezatu, V. Manta, and G. Vieriu

Abstract—Wireless networks rely greatly on security for authentication and confidentiality, due to the use of an open communication medium. The tendency to incorporate wireless capabilities in battery powered hand-held devices makes it difficult to enforce security because of computational resources and battery capacity limitations.

One approach to this problem consists in adaptable security mechanisms: different security algorithms may be enforced at any given time, to assure a proper level of protection, while respecting constraints on power and resource consumption. This paper presents such a mechanism (CLASS), which uses a clustering scheme to setup homogenous security areas in the network. The scheme is based on the available energy at every node and on the individual security requirements. By proper assignment of the nodes and by proper selection of security, we show that our method significantly reduces the average energy consumption compared to a heterogeneous network.

I. INTRODUCTION

THE research and technological advances that spurred the embedded systems world determined an increase in complexity, also resulting in the boost of security costs. The development of new, faster, feature rich embedded systems has boosted the emergence and improvement of security methods in parallel with ranges of new and more powerful types of attacks.

The security of embedded systems poses some specific problems due to the limited computational resources available and the overhead introduced by security. Moreover, battery powered systems generate new constraints on power consumption which translate to stricter limitations on computational performance. Also, systems with wireless capabilities tend to have more security requirements, due to the use of an unsecure communication medium [1]; and are more power hungry, due to the radio subsystem [2].

These problems may be addressed by leveraging the security-energy ratio. In general, it can be safely assumed

that by doing more computations a higher level of security can be achieved [3]. Thus, selection of an optimal security level translates to desired energy consumption and performance levels.

The increasing interest in flexibility of operation and minimum user intervention has led to the development of self-adaptable embedded systems. The main characteristic of these systems is the ability to adapt to variable internal and external conditions, resulting in high performance usage scenarios in various environments. Research topics have been proposed regarding the use of self-adaptable mechanisms in delivering security [4]-[7]. The main idea is to vary the enforced security at run-time based on variable constraints. Although the usage of such methods increases performance and decreases energy consumption when concerning a system node, the heterogeneity introduced by the adaptability mechanism has a negative impact when concerning a network of nodes. In the case of wireless mesh networks, where nodes with different security requirements try to communicate, the negotiation of security method to be used to secure data transmissions can lead to higher levels of power consumption and to the degradation of security for some nodes.

As our research work focuses on the latter problem, this paper proposes a mechanism to negotiate the security settings used in a wireless mesh network, with the direct goal of reducing energy consumption. We use a clustering scheme to setup areas where security is chosen based on the security and energy characteristics of the constituent nodes. The algorithm copes with run-time changes in nodes security and energy and adapts accordingly with imposed constraints.

In Section II we briefly outline related work; Section III presents some prerequisites concerning the field of our work and Section IV details the proposed method and explains its behavior. Section V presents a case study used to evaluate the performances of our algorithm and Section VI highlights the contributions of this work and proposes future research perspectives.

II. RELATED WORK

We focus on giving an overview on papers that cover two subjects: adaptable security with respect to energy considerations; and adaptable security in wireless networks.

This paper continues our previous work. In [4] we

Manuscript received June 14, 2011.

N. Botezatu is a PhD student at “Gh. Asachi” Technical University of Iasi, Faculty of Automatic Control and Computer Engineering (e-mail: nbotezatu@cs.tuiasi.ro)

V. Manta is with the “Gh. Asachi” Technical University of Iasi, Faculty of Automatic Control and Computer Engineering (e-mail: vmanta@cs.tuiasi.ro)

G. Vieriu is with the “Gh. Asachi” Technical University of Iasi, Faculty of Automatic Control and Computer Engineering (e-mail: gvieriu@cs.tuiasi.ro).

covered the aspects of self-adaptable security at a theoretical level. The paper proposes a security architecture for embedded systems and discusses the guidelines used for constructing an adaptable mechanism. In [5] we propose an energy-aware method to determine the security resources consistent with the requirements of the system. The proposed solution uses a decision mechanism to evaluate the relations between different security solutions and to select the appropriate one based on variable runtime requirements.

In [6] the authors propose the use of Analytic Hierarchy Process (AHP) to define priorities among different system requirements and to compare different security solutions. The approach relies on the solution of an optimization problem which is addressed by a Linear Programming formulation. The presented mechanism is used to secure mobile nodes communications.

AHP was also used in [7], where a context aware and adaptive security manager (CASM) is presented. The CASM is used to secure 802.11 wireless networks.

Reference [8] introduces a new wireless sensor network management and security platform. Alongside the description of the proposed mechanism that is used for managing public key based secure communications, the authors focus on presenting the security analysis and requirements of WSNs.

In [9] the authors propose the ANGEL security architecture, based on IEEE 802.15.4 security requirements. The architecture allows the deployment of ZigBee networks in a secure manner and facilitates the user access to these networks in a secure and transparent manner.

III. BACKGROUND

This section presents the information needed in order to better understand the proposed algorithm. It concerns the network model used and the security resources of every network node.

A. Network Model

This work assumes a wireless mesh network composed of N nodes, randomly deployed in a two-dimension field. The following list gives details and assumptions, similar to those in [8]-[12]:

- 1) All nodes are homogenous, meaning they can perform the same functions regarding our algorithm.
- 2) Every node has a unique ID.
- 3) The network is energy heterogeneous at start-up, as the nodes have different residual energy levels.
- 4) The links are symmetric; every node can determine the distance to another node if it is covered by its transmission range.
- 5) The network is always connected and any two nodes can communicate either directly or through routing nodes, according to a routing protocol.

- 6) All nodes are location unaware, meaning they do not know their coordinates in the field.

B. Security model

The security mechanism implemented on every node is based on the concept of security resources. A security resource is represented by different types of security algorithms used to secure data (e.g. symmetric key ciphers, public key ciphers, hash functions) and any node may implement more than one security resource.

Every node chooses at run-time which security resource to use, based on implementation of a self-adaptable mechanism (e.g. any of the mechanisms presented in [3], [5]-[7]). This mechanism reacts only to local conditions and constrains without considering a global network status. Based on the implemented security resource, every node creates a descending ordered list, L , where the first entry represents the most suitable security resource to be used, where as the last entry represents the least suitable resource to be used.

For energy evaluation purpose, we used the energy model in (1).

$$e_{SR} = e_{setup} + e_{data} \cdot \beta \quad (1)$$

where e_{SR} represents the total energy consumed for a security resource, e_{setup} represents the energy used for enforcing the resource (e.g. key generation) and e_{data} is the energy used for securing one data byte. β is the number of bytes processed with the given security resource.

IV. CLASS: CLUSTERING ALGORITHM FOR SECURITY RESOURCES

In a network where the nodes implement heterogeneous security, every communication session, between two nodes that use different security resources, has the following disadvantages: an increase in energy consumption, due to the security setup phase; and contravenes on security constraints, due to the use of a misfit security resource, for one of the nodes. Even if the security could be negotiated per communication session between the two nodes, the energy consumption would increase due to the protocol overhead.

In order to prevent such situations, we propose CLASS: a self-adaptable clustering protocol that uses a stochastic process to evenly distribute the security resources utilized by the network nodes. The nodes organize into cluster zones and elect the security resource to be used for the respective zone. For inter-cluster communication, every zone has a cluster-head node (CH) that routes the data. As cluster-heads consume more energy due to the use of at least two security resources, the CH are dynamically updated in order to distribute the energy consumption among all the nodes in the cluster.

The operation of CLASS is divided into rounds. Each

round has a setup phase where the clusters and initial security settings are determined. It is followed by a running phase where the network is available and operating normally. In this phase further adaptations on cluster-heads and security resource usage are made.

The following subsections describe the stages performed in the setup phase and the cluster-head and security adaptations methodologies.

A. Cluster formation

This process is similar to the LEACH algorithm [10] advertisement phase. It is based on the number of clusters the network will form, established a priori, and on the number of times a node has been elected to form a cluster. Every node generates a random value between 0 and 1. If the value is below a threshold th , the node begins forming the cluster. The threshold value is presented in (2) and is similar to the ones used in [10]-[12].

$$th = \begin{cases} \frac{p}{1 - p \cdot (r \bmod \frac{1}{p})} & , \text{node} \notin CS \\ 0 & , \text{node} \in CS \end{cases} \quad (2)$$

where p is the percentage of cluster heads and is calculated as the ratio between the number of clusters and the total number of nodes, r is the current round, and CS is the set containing the nodes that have elected the clusters in the last $1/p$ rounds.

Once a node has elected itself to form a cluster, it broadcasts a message to the rest of the nodes. The message contains the node unique ID, which will be further used as the cluster ID by the joining nodes. Once the broadcasts are finished, a node decides which cluster to join based on the distance to the nodes that form the clusters.

The motivation for this stage is based on the goal of limiting energy consumption. A classic approach to security negotiation between multiple peers would put a high load on message transmissions, resulting in a shorter network lifetime. By using the above methodology, we save energy by sending a minimum number of messages and we ensure the uniform distribution of nodes to the clusters, after a number of rounds.

B. Cluster-head and security selection

After each node has decided to which cluster it belongs, it broadcasts a message with the following information: ID, cluster ID, residual energy and security resource it uses. In this way, every node in a cluster learns its neighbors and their corresponding energy and security parameters. Note that this step takes a certain amount of time, long enough for a node to receive messages from the rest of the cluster neighbors.

Each node performs a deterministic sort for the cluster residual energy values and the node with the highest energy level is elected cluster-head, in a distributed fashion.

Next, the cluster-head tries to determine the most suitable security resource for the cluster. This is done through counting the number of time a security resource is desired by the cluster nodes. The security resource with the highest relevance for the cluster is the one that has the most counts, representing the majority among the nodes. The cluster-head broadcasts the security resource elected (CSR – cluster security resource), this type of message being named SU (security update). This operation has two effects: first, the cluster nodes are informed which security resource they have to use; and second, the other cluster-heads learn about each other. This is useful for routing messages between clusters. This operation signals the end of the setup phase.

C. Runtime adaptation

During the running phase of every round, every cluster can perform two types of adaptations: cluster-head changes and security resource updates.

In order to better understand the adaptations, we present three parameters that characterize every node: $(\Delta e, \alpha_s, a)$. Δe represents the maximum amount of energy a node can use while being cluster-head, α_s represents the level of adaptation a node can withstand, and a dictates the action taken when a node cannot adapt.

1) *Cluster-head changes* occur when the energy consumed crosses the Δe threshold. The current cluster-head broadcasts a message similar to the one presented in sub-section B, triggering the same selection process for the cluster-head. At this stage the security resource used remains unchanged.

2) *Security resource updates* include two distinct operations, one on the cluster-heads and one on the other nodes, and implies the use of two message types: SU (security update) – signals the selection of a different security resource by the cluster-heads; and SR (security request) – signals the change of the locally desired security resource on a normal node.

As mentioned before, the first SU message is sent at the end of the setup phase. Upon reception, if the CSR does not match the locally desired security resource, the node tries to adapt based on the α_s parameter. It can take a value between 0 and ns , where ns is the number of implemented security resources. The value 0 means that the node cannot adapt its security. For the other value, the node searches the list L (described in Section III.B) from the first value to the value of index α_s . If the search does finds a match for the CSR, the node starts using it without announcing the cluster-head. If the search does not return a valid result, the node sends a SR message signaling once again the desired security resource. In the latter case, the node can react in two ways, based on the value of a : *FE* (forced execution) – the node continues its operation and uses all security resources needed to communicate with its peers; and *SW* (stopped and

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Network layout and dimension	Bidimensional – 200 x 200 meters
Number of nodes	100
Nodes deployment	Random – uniform distribution
Node transmission range	89 meters
Initial energy	Random – uniform distribution between 1 and 8 Joules
Data source	Random – Bounded Pareto distribution ($l = 1$, $L = 8k$, $a = 1$)

TABLE II
SECURITY RESOURCES CHARACTERISTICS

Security resource	Encryption method	Setup energy - E_{setup} (μJ)	Setup time - T_{setup} (μsec)	Encryption energy - E_{data} ($\mu J/\text{byte}$)	Encryption time - T_{data} ($\mu\text{sec}/\text{byte}$)
SR ₁	3DES	89	305	6.7	11.7
SR ₂	DES	29.6	99.2	2.1	4.1
SR ₃	AES	7.7	62.9	1.3	7.2
SR ₄	Blow Fish	3057	875	0.8	2

waiting) – the node does not send data, does not respond to any incoming communication request and waits for a CSR update.

When local conditions on a node determine a change in list L , the search process is repeated. In this case the SR message is sent regardless of the result. The cluster-heads listen for SR messages and store their content by keeping track how many nodes desire each security resource. If the CSR stops being dominant for the cluster, then the cluster-head changes its value correspondingly, and signals the cluster security update by broadcasting a SU.

V. SIMULATION RESULTS

In order to validate our work we developed a SystemC based simulation environment. We setup a wireless mesh network with the parameters in Table I and we evaluated the CLASS algorithm alongside a non-adaptable direct communication setup, which will be named NADC.

For every message transmitted between every two nodes, the NADC secures the data using the desired security resource of the source node. Beside its desired security resource, the sink node also sets up the one needed for reception of the message. The more peers it has, the more security resources it has to use, if the nodes do not desire the same security resources.

The two methods were evaluated in respect to network lifetime and aggregate throughput. The two parameters were calculated based on the consumption and speed

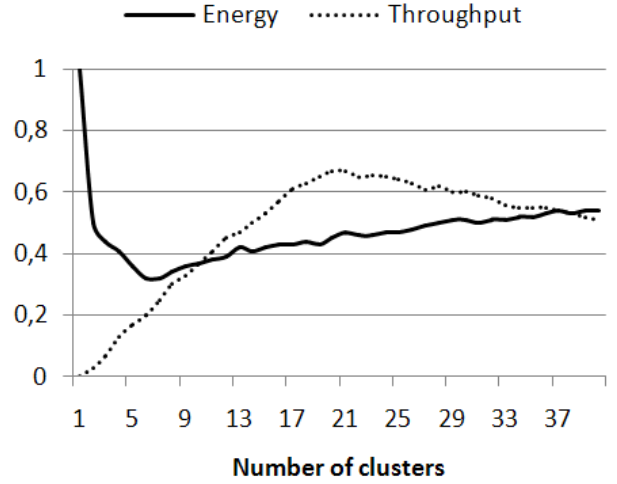


Fig. 1. Normalized aggregate energy consumption and normalized average throughput variation function of the number of clusters.

performance of the implemented security resources. We implemented four security resources presented in Table II. They were chosen due to their use among different implementations of TLS/SSL, for bulk data encryption. The mechanism used for choosing security resources locally is described in our previous work [5] and is not the scope of this paper.

Of the parameters that characterize the runtime adaptation of CLASS, Δe has a fixed value of 0.5 J, α has a value of 2; simulations were run for both values of a .

The amount of data considered for transmission is a stochastic input of the simulation and is generated using a Bounded Pareto distribution. The lower limit is set at one unit, the upper one is set at 8k units and the alpha parameter was chosen to be one. These values are consistent with both CLASS and NADC simulations.

In order to compare CLASS with the non-adaptable setup, first we determined the optimal number of clusters. Figure 1 presents the normalized aggregate energy consumption and the normalized average throughput. The number of clusters was varied between 1 and 100; the result obtained for the extreme values (1 and 100 clusters) resembled those obtained for NADC. The figure depicts an interval containing both optimal values for energy (6 clusters) and throughput (20 clusters). Due to these results, the rest of the simulations were run for different number of clusters between 6 and 20.

Figure 2 shows a comparison of system lifetime between CLASS and NADC. CLASS was run using 6 clusters for both *forced execution* and *stopped and waiting* values of a . We found that it takes approximately 7 times longer for the first node to die when $a=SW$ and 4.5 times longer when $a=FE$. Although the first case performs better, an average of 16 nodes are kept silent in every round, thus altering their

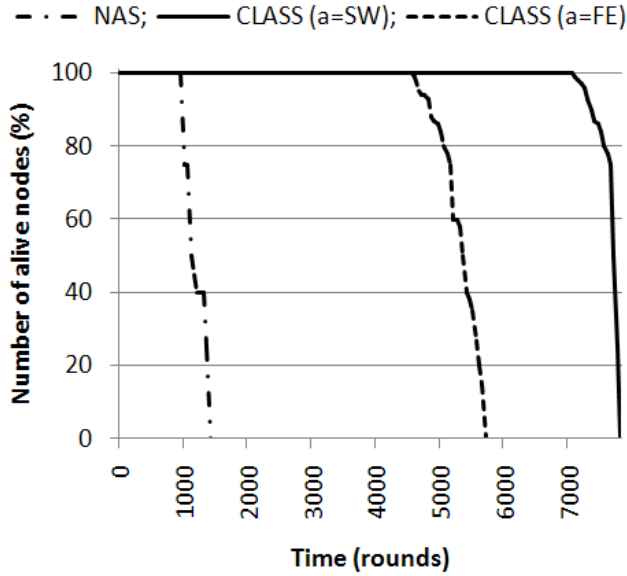


Fig. 2. Network lifetime expressed in number of cluster reconfiguration rounds.

TABLE III
NETWORK LIFETIME FOR DIFFERENT SIMULATION SCENARIOS

<i>a</i> parameter		Number of clusters	First node dies (round)	Last node dies (round)
CLASS algorithm	Forced execution	6	4562	5713
		10	4326	5355
		14	4240	5087
		17	3994	4941
		20	3811	4877
	Stopped and waiting	6	7078	7821
		10	6948	7702
		14	6899	7645
		17	6806	7587
		20	6752	7460
NADC setup	NA	NA	995	1376

performance. Table III presents the complete data for these experiments. The CLASS algorithm showed superior results in all simulation scenarios: in the worst case it took 3.8 times longer for the first node to die and when considering the difference between the first and last node death, on average the network lasted 2.3 times longer than in the NADC setup.

Next, we investigated how the performance of the network varies for CLASS (Figure 3). As we expected, the highest throughput was achieved for 20 clusters, for both values of a . The difference for the two values is explained by the impact of the nodes that do not communicate when $a=SW$. The result show that with an increase in number of clusters comes an increase in nodes that cannot adapt. Although the throughput reduces to about a half when

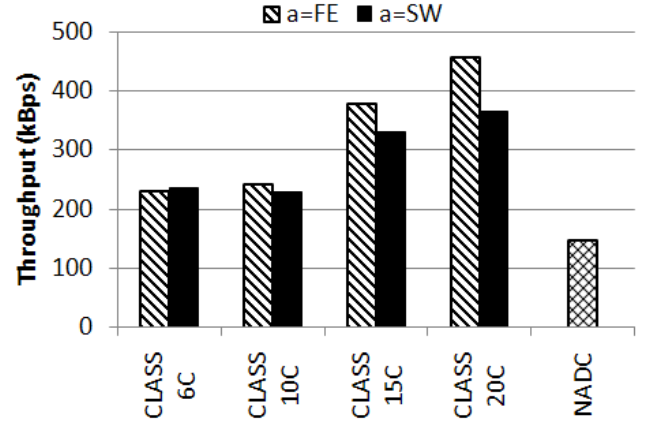


Fig. 3. Average network throughput for the CLASS algorithm having 6, 10, 15 or 20 clusters and for the NADC setup.

having 6 clusters, the throughput is still 1.5 times better than when using NADC.

The experimental results conclude the superiority of CLASS compared to NADC. They also show that the algorithms parameters must be tuned to the networks characteristics, in order to obtain the best result for network lifetime or throughput.

VI. CONCLUSION AND FURTHER WORK

In this paper, we proposed CLASS, a self-adaptable security algorithm for wireless mesh networks. It uses a clustering scheme to distribute the energy consumption among the network nodes and an adaptable mechanism to generate homogeneous security requirements for every cluster.

The simulation results show that CLASS outperforms a direct communication setup at both energy saving and aggregate throughput: the network lifetime was prolonged with at most 7 times and the throughput was boosted at least 1.5 times.

Future work includes the evaluation of our mechanism under real life scenarios, through prototype implementation of a wireless node platform. Also, we intend to study the energy consumption and processing overheads introduced by CLASS.

REFERENCES

- [1] S. Ravi, P. Kocher, and S. Hattangady, "Security in Embedded Systems: Design Challenges", in *ACM Transactions on Embedded Computing Systems*, vol. 2, no. 3, Aug. 2004, pp. 461–491.
- [2] M. Marcu, D. Tudor, and S. Fuiuc, "Towards a Network-Device Unified Framework for Power-Aware Wireless Applications", in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, Leipzig, Germany, 2009, pp. 963–967.
- [3] R. Chandramouli, et al., "Battery Power-aware Encryption" in *ACM Transactions on Information and System Security*, vol. 9, no. 2, May 2006, pp. 162–180.

- [4] N. Botezatu, V. Manta, and A. Stan, "Self-adaptable Security Architecture for Power-aware Embedded Systems", in *Proceedings of the 14th International Conference on System Theory and Control*, Sinaia, Romania, 2010, pp. 98–103.
- [5] N. Botezatu, V. Manta, and A. Stan, "Self-adaptability in secure embedded systems: an energy-performance trade-off", *The 2011 International Conference on Information Security and Internet Engineering, World Congress on Engineering*, London, UK, to be presented.
- [6] A. Taddeo, P. Marcon, and A. Ferrante, "Negotiation of security services: a multi-criteria decision approach", in *Proceedings of the 4th Workshop on Embedded Systems Security*, Grenoble, France, 2009, Paper 4.
- [7] C. Hager, "Context Aware and Adaptive Security for Wireless Networks", Ph.D. dissertation, Dept. Elect. Eng., Virginia Polytechnic Univ., Blacksburg, VA, 2004.
- [8] B. Lee, S. Bae, and D. Han, "Design of Network Management Platform and Security Framework for WSN", in *Proceedings of the 2008 IEEE International Conference on Signal Image Technology and Internet Based Systems*, Bali, 2008, pp. 640-645.
- [9] O. Garcia-Marchon and H. Baldus, "The ANGEL WSN Security Architecture", in *Proceedings of the Third International Conference on Sensor Technologies and Applications*, Athens, Greece, 2009, pp. 430-435.
- [10] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", in *Proceedings of the 33rd Hawaii International Conference on System Sciences*, Hawaii, 2000, vol.8, pp. 8020
- [11] H. Gou, G. Li, and Y. Yoo, "A Partition-Based Centralized LEACH Algorithm for Wireless Sensor Network Using Solar Energy", in *Proceedings of the International Conference on Convergence and Hybrid Information Technology*, Daejeon, Korea, 2009, pp. 60-66.
- [12] M.J. Handy, M. Haase, and D. Timmermann, "Low Energy Adaptive Clustering Hierarchy with Deterministic Cluster-Head Selection", in *Proceedings of the 4th International Workshop on Mobile and Wireless Communications Network*, Stockholm, Sweden, 2002, pp. 368-372.

Monitoring CAN Performances in Distributed Embedded Systems

Florin Catalin Braescu, Lavinia Ferariu and Andrei Franciuc

Abstract—The paper presents a new module devoted to provide a refined monitoring of the traffic on a CAN bus. The real-time application in charge with message CAN communication is developed in compliance with FreeRTOS and dsPIC33. It is connected by a serial interface with the PC host application. Unlike other similar existing devices, the new CAN monitor extracts enriched information regarding the error and overload frames, it detects the overdue RTR message responses and permits various display filtering options. In addition, the CAN monitor estimates the current CAN bandwidth, thus giving the possibility to foresee potential bus overloading situations. These services are targeted to facilitate the debugging and the exploitation of CAN-based applications, as experimentally illustrated.

I. INTRODUCTION

THE CAN protocol is widely used within distributed systems architectures [1]. Its most important benefits are related to the fact that it provides a reliable, high speed, event - triggered communication, as well as a simple and low-cost utilization. In automotive industry, CAN is mainly used to interconnect many Electronic Control Units (ECUs) that exchange information messages on bus. CAN's main competitors are FlexRay and local interconnected network (LIN). Even it is faster and more reliable than CAN, FlexRay is still rarely employed, mainly because of its much higher price. On the other hand, given its slow speed, LIN is almost exploited as a cheap sub-network of CAN bus, being targeted to integrate sensors and actuators in nowadays cars [2].

In order to support an efficient and robust development of CAN-based applications, various tools allowing CAN traffic monitoring have been produced, PcanView being probably the most popular one. Usually, a CAN monitor is connected to a host PC and to at least one CAN bus, whose traffic it illustrates by means of a PC host application. The common facilities include showing the type, the content and the properties of the exchanged CAN messages, creating logs which describe the CAN activity and sending customized CAN messages (periodically or at user's command). It is worth mentioning that almost of the

approaches connects the CAN monitor according to an open loop architecture, having in mind the monitoring goal, solely. However, the information provided by the CAN monitor can also be used for the on-line reconfiguration of the application, if the system is designed as a closed loop based one.

This paper presents a flexible, multi-features CAN monitor composed by a FreeRTOS embedded application (loaded on the CAN node which needs to be monitored) and a PC host application. The later receives the CAN traffic data from the embedded application and sends back user's commands. In comparison with the existing on-market CAN monitors, the proposed solution supervises the CAN bus by extracting additional information regarding the exchanged CAN messages, it detects overdue Remote Transmission Request (RTR) message responses and it allows refined filtering display options. Besides, this new CAN monitor is able to detect the actual bandwidth of the CAN bus signaling whenever the traffic exceeds the availability of the CAN bus. Therefore, the suggested CAN monitor allows an easier debugging of CAN based application, yet it offers valuable information about the anomalies detected in the distributed system, such as communication errors, slow answering, overloaded CAN devices. The information can be exploited for an adequate off-line re-design of the CAN based application (e.g., overloading could be solved by using multi-domain CAN-based distributed architectures or by means of adequate message prioritization). Besides, the information can be used for a flexible, on-line reconfiguration of the communication tasks.

The paper is organized as follows. Section II presents the main characteristics of the CAN protocol, while section III describes the most important facilities of FreeRTOS real-time operating system which was employed for the CAN-based embedded application. Details regarding the suggested CAN monitoring application are given in section IV and several preliminary experimental results which demonstrate the usability of the suggested approach are discussed in section V. Last section is devoted to conclusions.

II. CAN PROTOCOL OVERVIEW

The presented CAN monitor exploits the facilities of FreeRTOS real-time operating system and CAN communication protocol to allow an efficient observation of

This work was supported by The National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008.

F. C. Braescu, L. Ferariu and A. Franciuc are with the "Gheorghe Asachi" Technical University of Iasi, 27 Dimitrie Mangeron Bd., 70050, Iasi, Romania (e-mail: cbraescu@ac.tuiasi.ro, lferaru@ac.tuiasi.ro, don_franc@yahoo.com).

the data exchanged within the CAN-based distributed systems architectures [3].

CAN is a multi-master broadcast serial bus communication standard [4]. Each device is able to send or to receive messages on the bus, but not at the same time. All messages are broadcasted, each CAN receiver deciding to process or to ignore a certain message. The messages which are communicated can be of standard, extended, error, remote or overload frame.

In the case of standard or extended frame, the CAN message is composed of the following fields: start-of-frame (SOF), arbitration, control, data, cyclic redundancy check (CRC), acknowledge (ACK) and end-of-frame (EOF). The arbitration field is composed by the message's identifier (ID) and by the Remote Transmission Request bit. The control field provides more information about the contents of the CAN message, such as the number of data bytes present in the message and the frame's type (standard or extended). The messages are prioritized, the priority being given by the identifier field which is 11 bits long (standard frame) or 29 bits long (extended frame). The less is the ID's value, the higher is the priority of the message.

It should be also noticed that the CAN protocol offers an arbitration mechanism during message sending. If two messages are simultaneously sent on the bus and a collision is detected, the arbitration is won by the message with the highest priority. Its sending device will transmit the whole message, while the devices that lose the bus arbitration will hold the transmission and retry to send their messages later. The arbitration mechanism determines a very important restriction over the CAN message identifier field: it must be unique on the bus, as it will identify the message source or the message data content. All the receiving devices will use the identifier to detect whether the message is relevant for them or not. If messages with the same identifier are sent simultaneously by two CAN devices on the same bus, the arbitration mechanism fails, leading to communication errors.

A remote frame, also named RTR frame, allows a destination CAN node to request data from a source CAN node. It is transmitted by a node on the bus, to ask for the transmission of a data frame with the same identifier from another node. The nodes receiving a RTR frame check whether there is the corresponding transmitter defined or not. The node satisfying the condition sends the RTR response data frame. As the request and the response frames are completely different frames, one can notice that the response frame could be delayed due to messages with higher priorities exchanged on the CAN bus [5].

The CAN protocol provides an appropriate environment for an automatic detection of data transmission errors and for a centralized error management. The error frame is generated by any CAN node that detects a bus error. Once

received an error frame, all the CAN nodes drop the last received message and the transmitter resends the message. The error frame is composed by an error flag field and an error delimiter field. The CAN protocol implements mechanisms of error detection, error handling and error limitation. The error detection one allows the detection of the following error types: bit error, frame error, acknowledge error, CRC error and bit stuffing error. The error handling mechanism is based on the error flag field to signal the error's detection as well as the error indication receiving confirmation, whilst the error delimiter field permits the bus nodes to restart communications after an error has occurred. The error limitation mechanism permits the CAN bus to work correctly even if there are errors-generating faulty nodes. Therefore, each node can be in one of the following three error states: error active, error passive and bus off (the node can neither send nor receive). The CAN bus error limitation mechanism is graphically illustrated in Fig.1, where the transitions between the possible node states are made depending of the Transmission Error Counter (TEC) and Reception Error Counter (REC) values.

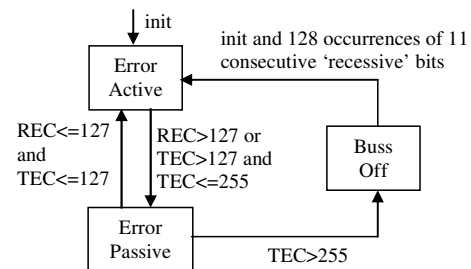


Fig. 1. The CAN bus error limitation mechanism.

The CAN protocol assures inter-frames spacing and bit stuffing mechanisms as well. The inter-frame spacing mechanism provides that standard, extended and remote frames are separated by an inter-frame space, whilst the bit stuffing one ensures that a bit of opposite polarity is inserted after five consecutive bits of the same polarity.

The overload frame indicates a detected overloading situation and is generated by a node either when a dominant bit is detected during inter-frames space or when a node is not yet ready to receive the next message. The last case could appear when the node is still reading the previous received message. The overload frame has the same format as an error frame but can only be generated during an inter-frame space. By means of the overload frame, the transmitter is requested to transmit the message again, the next transmission's start being delayed by the overload frame. A node can generate a maximum of two consecutive overload frames, in order to delay the start of the next message.

The CAN protocol does not provide a deterministic

response time for messages assigned with low priorities [6]. Therefore, in order to assure a good utilization of the CAN bus, the messages with short periods of occurrence should have allocated higher priorities [7]. The maximum transfer rate on CAN bus is 1 Mbit/s, whilst the typical transfer rates kept within automotive and industrial environments are 500 Kbit/s, 250 Kbit/s and 125 Kbit/s.

III. FREERTOS REAL-TIME OPERATING SYSTEM FACILITIES

The embedded CAN based application is built in compliance with FreeRTOS. FreeRTOS is especially designed to support an easy development of portable and modular embedded applications [8]. Its portability is mainly assured by means of a set of standard API services, which make the applications compliant with various hardware architectures. Moreover, FreeRTOS provides all the capabilities required for a multitasking execution, such as rapid commutation between different contexts, predictable interrupts handling, simple access to shared resources, easy-to-use inter-task communication and synchronization mechanisms, simple time management and various tracing facilities for monitoring the execution sequence and the memory usage.

The concurrent processes of the embedded application are defined as tasks, co-routines or interrupt service routines (ISRs). A task can have one of the following states: running (it is executed), ready (it demands to be executed), blocked (it is waiting for a temporal event) and suspended (it is not available for scheduling). Comparing with tasks, the ISR processes (used for providing specific responses to interrupts) accept only a limited range of API services. Usually, their priorities (higher than the priorities of any task) are not managed by FreeRTOS. However, the ISR processes that call API services must be assigned with a priority equal to the priority of the real time kernel interrupt, in order to allow a proper execution of necessary system operating services. On the other hand, the execution of tasks (preemptive or non preemptive), is supervised by a time-driven scheduler according to the allocated priorities. FreeRTOS real-time operating system includes API services that allow full control over the tasks priorities and states during application execution, like changing the tasks' priorities, starting, blocking, suspending or even deleting the tasks.

The FreeRTOS kernel allows preemptive or non preemptive scheduling. In the preemptive mode the implicit scheduling points are periodically distributed. The scheduling algorithm is "highest priority first". If more tasks in the ready state have the same highest priority, the round robin scheduling is employed. The scheduling points can also occur asynchronously, for instance when a task is blocked, suspended or gains a new priority (higher than the

priority of the running task), when a task of higher priority is created or when an explicit rescheduling is demanded.

The inter-task communication and synchronization mechanisms provided by FreeRTOS operating system are queues and semaphores. The queues are mainly used for communication purposes whilst the semaphores are employed for synchronization ones.

FreeRTOS provides three different memory allocation schemes that allow the programmer to implement a proper RAM allocation algorithm within an application [9]. Therefore, the API services required for memory allocation are located in the FreeRTOS portable layer. The differences between the available memory allocation schemes take into account the disadvantages brought by memory allocation functions (`malloc()` and `free()`) (e. g. memory fragmentation). While the first memory allocation scheme allows static allocation only, the second one allows the dynamic allocation of blocks of predefined sizes, and the third one provides complete freedom in using `malloc()` and `free()` functions.

The kernel functionalities and the API services availability are tailored to a particular application by means of a configuration file, `FreeRTOSConfig.h`. Its specific definitions indicate the kernel type (preemptive or cooperative), the microcontroller's frequency, the allowed tracing facilities, the employed inter-task communication and synchronization mechanisms, etc.

IV. CAN MONITOR APPLICATION

The solution proposed for monitoring the traffic on CAN buses includes a FreeRTOS embedded application (which is loaded at least on the CAN node needing to be monitored) and a PC host application connected to the monitored CAN node. These two modules are described in sequel.

A. FreeRTOS-Based Target Application

The FreeRTOS-based embedded application supervises the CAN bus, extracts information about the traffic and sends necessary information to the PC host application via the serial interface. As consequence, the application includes an Enhanced Controller Area Network (ECAN) driver compliant with dsPIC33 ECAN controller and a Universal Asynchronous Receiver/Transmitter (UART) driver.

The structure of the embedded application, together with the involved FreeRTOS specific data types is illustrated in Fig.2. The application is built as a set of tasks and ISR processes which communicate via queues of messages. The queue `xRXQueue` stores the information extracted by the `ECAN_RX_ISR` from the CAN bus, `xCharsForTxQueue` allows the sequential transmission of characters by means of UART interface from the embedded application to the PC host monitor, whilst the

`xRxdCharsQueue` keeps the commands received by UART from the PC application.

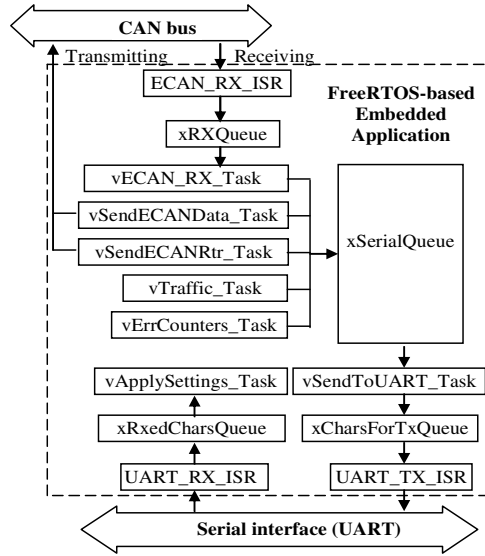


Fig. 2. The structure of FreeRTOS-based embedded application.

Although the ECAN module has a single interrupt vector, the `ECAN_RX_ISR` is specifically implemented to differentiate between possible multiple sources of interrupt, such as errors (error state warning, error state bus passive and error buss off), bus overload (reception buffer overwriting), data receiving or RTR request. The detected type of the interrupt source is stored in the elements of `xRXQueue` by means of two additional fields (namely `message_code` and `message_type`) which complement the implicit data embedded within the CAN message. The values allowed for `message_code` are `MSG_RX` (received message), `MSG_ERR` (bus error) or `MSG_OVF` (overflow error), while `message_type` gives refined information concerning the elements having the same `message_code` value (e.g. types of errors, RTR or data message, etc).

The communication on the serial interface is managed by an interrupt-driven driver compliant with FreeRTOS and dsPIC33. It solves data reception and transmission by means of `UART_RX_ISR` and `UART_TX_ISR`, respectively. The commands received from the PC host application are read by the `vApplySettings_Task` from `xRxdCharsQueue` and appropriate settings are made to implement requested message filtering and transmission buffers prioritization.

Using data available in `xRXQueue`, the task `vECAN_Rx_Task` determines the periods of the messages, updates and checks the overdue RTR message responses, and afterwards sends these results to the UART gatekeeper task. On the other hand, the `vTraffic_Task` task calculates the traffic on the CAN bus and sends the

determined value to the PC application, whilst `vErrCounters_Task` transmits the current values of the TEC and REC. Because the embedded application has multiple independent tasks that need to asynchronously send certain information on the UART interface, a resource sharing mechanism is implemented by means of `xSerialQueue`, having in mind to prevent UART communication corruption. To fit that end, the task `vSendToUART_Task` acts as a UART gatekeeper task and all the tasks which require sending data via UART, are forced to temporarily store the information in `xSerialQueue`.

The embedded application also includes two tasks, `vSendECANRtr_Task` and `vSendECANData_Task`, which can be used for customized testing purposes. They periodically send an RTR and a data message, respectively, on the CAN bus.

B. PC Host Application

The structure of the proposed CAN monitor is illustrated in Fig.3. As aforementioned, the FreeRTOS-based embedded application monitors the CAN traffic and sends the extracted information to the PC host application via the serial interface (selected by the user). The PC application is implemented by the authors in C++ within the MFC framework, being in charge with adequate displaying of all relevant received information and the transmission of user commands.

The serial communication is managed using an instance of `CSerialComm` class. In order to eliminate the risk that undesired communication errors will block the graphical interface, the communication is implemented as a separate thread. Poling techniques are employed, hence the reception of commands is made within an infinite loop which detects the available information coming from the FreeRTOS-based embedded device. The end of the monitor application is signaled to the communication thread via an event set from the main program; this thread is forcibly terminated if its execution is not ended within a timeout of 300ms.

A customized communication protocol was defined to allow an appropriate data transfer between the two modules of the application, namely the FreeRTOS embedded one and the PC host monitor. The PC application sends the user commands encrypted with a single character; the set of accepted commands includes: selecting the masks used for message filtering, selecting the threshold interval for the overdue RTR message responses, selecting the priorities for the transmission buffers or changing the period of the test messages sent by the monitor.

The FreeRTOS-based embedded application sends back the information about the messages exchanged on the CAN bus. The data is stored in messages with predefined structure, some examples being portrayed in Fig. 4.

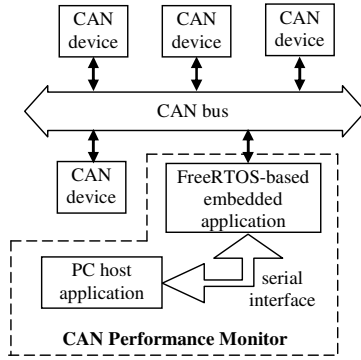


Fig. 3. CAN monitor system connected to a CAN bus.

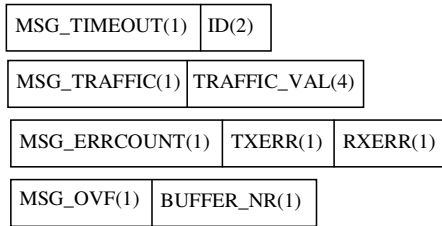


Fig. 4. Some of the fixed-structure based messages sent by the embedded application. The dimension of each filed is indicated between brackets as number of bytes.

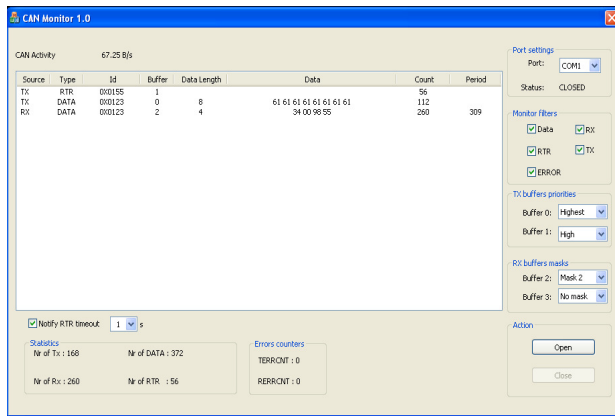


Fig. 5. Displaying the CAN performances within the PC host application.

The timeout for the RTR message responses is determined within the `vECAN_Rx_Task`. It stores in a specific array the arrival time of RTR messages and it detects if the corresponding response is overdue.

Usually the CAN monitors (including PcanView) display the monitor's error state ("Error Active", "Error Passive" and "Buss Off") without mentioning the number of transmitter or receiver errors. This information is useful for detecting a bus problem or even a node problem, so the presented CAN performance monitor displays both the node state and the errors counters values.

All the information received by the PC host application (message identifiers, message data, and the detected messages periods) and the bandwidth of the CAN bus are on-line updated within the graphical interface (Fig.5). The PC host application records the CAN bus activity in log file which can be used for future analysis.

V. EXPERIMENTAL RESULTS

Several experiments have been carried out to demonstrate the main functionalities of the proposed solution. The FreeRTOS-based embedded application was targeted to a dsPIC33FJ128MC802 microcontroller from Microchip. The microcontroller is composed of standard on-chip peripherals, including six asynchronous serial communications interfaces (SCI) and one ECAN module compatible with CAN 2.0 A, B. The ECAN has up to eight sending buffers and up to 32 reception buffers, with 16 receiving filters and three masks. It works according to First-In-First-Out (FIFO) mode by using the Direct Access Memory (DMA) and it employs automatic processing of the RTRs. As the microcontroller has a built-in ECAN controller without a CAN transceiver, a MCP2551-based CAN transceiver board was designed in order to connect the microcontroller to the CAN bus.

The periods and the priorities of the tasks running on the FreeRTOS-based embedded application are presented in the table I. Being in charge with CAN reception, `vECAN_Rx_Task` is set with the highest frequency and the highest priority. `vSendToUART_Task` switches aperiodically to the ready states, whenever a message is stored in `xSerialQueue`. The other tasks responsible with sending the CAN traffic/error related information are assigned with lower priorities and lower frequencies.

TABLE I
EMBEDDED APPLICATION TASKS CHARACTERISTICS

TASK NAME	TASK PRIORITY	TASK PERIOD (MS)
<code>vECAN_Rx_Task</code>	5	5
<code>vTraffic_Task</code>	3	500
<code>vErrCounters_Task</code>	2	1000
<code>vSendToUART_Task</code>	4	aperiodic
<code>vApplySettings_Task</code>	4	aperiodic
<code>vSendECANData_Task</code>	1	1000
<code>vSendECANRtr_Task</code>	1	2000

The detection of the overdue RTR responses represents an important feature of the suggested CAN monitor, because it allows the detection of disconnected or overloaded nodes. The PC host application permits selecting between three different thresholds for the RTR response time interval (1s, 1.5s and 2s). In order to test this feature, the proposed CAN monitor was connected to the same bus with an off-the-shelf CAN monitor employing PcanView software. Two messages with 0x0123 ID were created: an RTR message and the correspondent response data message. The thresholds for the RTR response time interval was set in the PC host application to 1s. The period of the RTR message sent by the PcanView software was set to 2s whilst the correspondent response message was set to be sent at user's command. As long as the RTR response data message was sent within 1s from RTR message

transmission, both messages were displayed. The overdue RTR response message was detected and the error message was displayed immediately when the response message delayed more than 1s. An example of detected overdue RTR response is illustrated in Fig.6 (for RTR message with 0x0123 ID).

In order to validate the functionalities related to the monitoring of errors counters and error messages generated by the PC host application, the CAN communication was interrupted by closing the PcanView application. The PC displayed the “Error State Warning” and the “Error state bus passive” (Fig.7) when the transmission errors counter reached the 96 and 128 values, correspondingly (Fig.8). The counter’s value started to decrease after re-starting the PcanView application and the CAN communication, the CAN performance monitor returning to the “Error State Warning” state and than to the normal operation state. Within a closed loop monitoring architecture, the exact information about the values of the errors registers would allow adopting the required actions for preventing the node entering the “Buss Off” state.

Source	Type	Id	Buffer	Data Length	Data
TX	RTR	0x0155	1		
TX	DATA	0x0123	0	8	61 61 61 61 61 61 61 61
RX	RTR	0x0123	2		
RX	DATA	0x0123			TIMEOUT

Fig. 6. The overdue RTR message responses detection.

Source	Type	Id	Buffer	Data Length	Data
TX	DATA	0x0123	0	8	61 61 61 61 61 61 61 61
TX	RTR	0x0155	1		
RX	DATA	0x0123	2	4	56 53 78 79
RX	RTR	0x0123	2		
ERR					Transmitter is in Error State Warning
ERR					Transmitter is in Error State Bus Passive

Fig. 7. The error detection/ limitation mechanism and the corresponding messages sent by the embedded application.

Statistics		Errors counters	
Nr of Tx : 73	Nr of DATA : 63	TERRCNT : 128	
Nr of Rx : 26	Nr of RTR : 36	RERRCNT : 0	

Fig. 8. The values of the error counters displayed by the PC host application.

The experiments showed that the CAN application monitor is able to observe the traffic on CAN bus at rates almost as high as the existing on-market CAN monitor. However, its performance is still limited by the serial interface that represents a bottleneck for the data exchanged. The UART interface is programmed to work at 38400bps, with 8 bits of data, 1 bit of stop and no parity bit. An upgrade from serial to USB communication would allow the PC application to display correctly all the CAN exchanged information.

Certain FreeRTOS API services could be used to improve the functionalities of the CAN performance monitor as well. The transmission of periodical or at user’s command messages would be implemented by employing

the operating system’s task creation specific API services. The two tasks included for customized testing purposes (vSendECANRtr_Task and vSendECANData_Task) can be replaced with CAN message transmitting tasks created at user’s command. Each message period and content could be specified from the PC application, and could be dynamically reconfigured within the embedded application by taking into account the current status of the CAN bus.

VI. CONCLUSION

The paper presents a new CAN monitor designed to support the development of CAN based embedded applications. The proposed system is composed by two modules which communicate via a serial interface: a FreeRTOS-based embedded application that runs on a dsPIC33 platform and a PC host monitor. The embedded application sends the CAN extracted information to the computer whilst the PC application sends back the user’s commands.

In comparison with other available nowadays CAN monitors, the proposed solution shows extended information about CAN traffic regarding the type of communicated CAN messages and the detected communication errors, the detected overdue RTR message responses and the current CAN bandwidth. The PC host monitor allows flexible filtering options for the displayed messages and records the whole CAN activity in specific log files.

Future work will include the upgrade from serial to USB communication, as well the development of specific algorithms able to ensure a dynamic prioritization of the communication tasks based on the information provided by the CAN monitor.

REFERENCES

- [1] R. Zurawski, *Embedded Systems Handbook*, Ed. CRC Press, 2006.
- [2] Li Q., Yao C., *Real-Time Concepts for Embedded Systems*, CMP Books, USA, ISBN 1-57820-124-1, 2003.
- [3] G. Buttazzo, “Research Trends in Real-Time Computing for Embedded Systems”, *ACM SIGBED Review*, vol. 3, no. 3, July 2006.
- [4] R. Bosch, *CAN Specification Version 2.0*, September 1991.
- [5] R.I. Davis and A. Burns “Robust priority assignment for messages on Controller Area Network (CAN)”. *Real-Time Systems*, Volume 41, Issue 2, pages 152-180, February 2009.
- [6] R.I.Davis, A. Burns, R.J. Bril, and J.J. Lukkien. “Controller Area Network (CAN) Schedulability Analysis: Refuted, Revisited and Revised”. *Real-Time Systems*, Volume 35, Number 3, pp. 239-272, April 2007.
- [7] R.J. Bril, J.J. Lukkien, R.I. Davis, and A. Burns. “Message response time analysis for ideal controller area network (CAN) refuted”. *In proceedings, 5th International Workshop on Real-Time Networks (RTN’06)*. July 2006.
- [8] R. Barry , *FreeRTOS - User Manual*, 2007.
- [9] R. Goyette, *An Analysis and Description of the Inner Workings of the FreeRTOS Kernel*, 2007.

The Influence of Finite Word Length Representation of the Delta Predictive Controller Design for Automotive Powertrain Systems

C. Budaciu and C. Lazar

Abstract — The automotive systems contain fast processes that implies fast sampling rate. Moreover, the existing digital hardware contains processing registers that have finite precision and therefore, the representation can be restricted to a finite number of bits. Predictive control strategy in discrete δ domain, State Space δ Generalised Predictive Control, in particular, referred as SS δ GPC algorithm, can bring significant improvements in terms of choosing a small sampling period combined with finite word length representation. The aim of this paper is to emphasize the advantages of the SS δ GPC algorithm, compared to the classical GPC strategy for different word length representation.

I. INTRODUCTION

THE control law implementation using fixed point arithmetic provides lower memory requirements, power consumption and higher speed in comparison to floating point representation. Nowadays, due to the performance requirements and efficiency, the powertrain control systems must include powerful processors, sensors, actuators of the latest generation and at the same time these components should not be very expensive. Moreover, the automatic control strategy plays an important role in automotive powertrain systems. To combine all these requirements, the scientific community in the field of control system design suggested that model predictive control has a large potential for control of automotive powertrains [1]. The biggest challenge derives from the fact that implementation of the predictive control algorithm should be done using finite word length representation because of the limited word length of the control units. In contrast, almost all the automatic control strategies are designed with real numbers and usually implemented in floating point. Moreover, due to truncation errors a stable closed-loop system may become unstable when the infinite precision control law is implemented with a fixed point digital control processors [2]. Further, in terms of control performances, the stability of a digital system may be lost

due to the finite word length effects at practical implementation of digital control or filter algorithms [3]. Most traditional control algorithms are based on shift operator description, but these strategies are ill conditioned when applied to data taken at the sampling rates that are high relative to the dynamics of the underlying continuous time processes [4]. Many algorithms are better-conditioned numerically using *delta* (δ) operator implementation than that ones based on shift operator implementation. Since the publishing of [5], many researchers paid attention towards identification and adaptive control using δ operator [6]. Nowadays, the δ operator is widely applied in wideband communication and real time digital control [7]. Relationship between the q operator and δ operator has been analyzed in many research papers and it is shown that the δ parametrization has better finite word length (FWL) closed loop stability margin than the traditional discrete time operator. Unfortunately, the discrete domains are unconnected with the continuous domain, this is because the underlying continuous domain description cannot be obtained by setting the sample time $T_s = 0$ [8]. There is a close connection between continuous time result and δ representation. In fact δ domain description converges to the continuous time counterpart for sampling period tends to zero. Moreover, the δ operator has certain numerical advantages compared with q shift operator. The paper [9] compares the δ operator with the q operator state space realizations in terms of the effects of the finite word length errors on the actual transfer function. The results show that the parameterization in the δ -domain provides a superior sensitivity performance over those in the q -operator. The authors Cheng and Chiu in the paper [10] analyze and propose upper bounds on the sample rate, lower frequency of interest and the available word length for controller representation. Generalized predictive control [11], is the most popular controller among of all predictive control formulations. The shortcomings of these control techniques are particularly emphasized especially when processes are to be run very fast and involve high sampling frequency [12]. During the last few years some research paid attention to δ -domain GPC to emphasize the close connection between discrete time and continuous time theory. In this context, the scientific community in the field of control system design suggested that some strategies seem to improve both the design and implementation for embedded

Manuscript received June, 18, 2011. This work was supported by CNCIS-UEFISCSU, project number PN II-RU PD cod 331/2010.

C. Budaciu. is with Gh. Asachi Technical University of Iasi, Department of Automatic Control and Applied Informatics, (e-mail: chalaucu@ac.tuiasi.ro).

C. Lazar is with the Gh. Asachi Technical University of Iasi Department of Automatic Control and Applied Informatics, (e-mail: clazar@ac.tuiasi.ro).

devices due to numerical and conceptual advantages offered by the δ operator.

In this paper, the benefits of the δ operator are investigated from the predictive control strategy perspective. The aim of this paper is to emphasize the advantages of the State Space δ GPC algorithm, compared to the classical GPC strategy for different word length representation. The investigation is entitled because nowadays it is more desired the fixed point implementation for electronic control units in automotive industry for the reason of cost, simplicity, power consumption and memory space.

The paper is organized as follows: a brief background to the δ operator is given in Section 2, the framework for SS δ GPC is presented in Section 3. Section 4 deals with the automotive powertrain affine model on a state space approach system and the conversion to the δ -domain. In Section 5 the performance of the equivalent shift and δ -domain predictive control algorithms are compared in the scenario of fast-sampling and finite word length digitization and Section 6 concludes the paper.

II. DELTA OPERATOR

A. The δ operator definition

The δ -operator can be directly substituted into q -operator from the definition:

$$\delta = \frac{q-1}{T_s} = \frac{e^{sT_s} - 1}{T_s} \quad (1)$$

Although there is a linear transformation between the two discrete domains, the two operators have distinct conceptual roles:

$$q^j = (1 + \delta T_s)^j = \sum_{n=0}^j C_j^n (T_s \delta)^n \quad (2)$$

$$C_j^n = \frac{j!}{n!(j-n)!}$$

where T_s is the sampling period and q is the usual forward-shift operator. In the case of continuous time system, the stable poles should be in the left half plane of the complex plane, whereas in the discrete time this becomes the interior of the unit circle. In the case of δ -domain, the stability region is the interior of a circle centered at $(-1/T_s, j0)$ with the radius $1/T_s$ [10]. Moreover, the δ -domain converges to its continuous time counterpart as the sample time approaches zero value. This property gives the δ operator its superior performance at high sample rates as compared to the classical shift operator.

B. δ domain state space model

The δ discrete state space model can be written as:

$$\begin{aligned} \delta x(k) &= \mathbf{A}_\delta x(k) + \mathbf{B}_\delta u(k) \\ y(k) &= \mathbf{C}_\delta x(k) + \mathbf{D}_\delta u(k) \end{aligned} \quad (3)$$

The deterministic case of single input single output state space form in the classical representation in the q discrete domain is:

$$\begin{aligned} qx(k) &= \mathbf{A}_q x(k) + \mathbf{B}_q u(k) \\ y(k) &= \mathbf{C}_q x(k) + \mathbf{D}_q u(k) \end{aligned} \quad (4)$$

where the relationship between the two discrete models is given by [8]:

$$\begin{aligned} \mathbf{A}_\delta &= \frac{\mathbf{A}_q - I}{T} , \\ \mathbf{B}_\delta &= \frac{\mathbf{B}_q}{T} , \\ \mathbf{C}_\delta &= \mathbf{C}_q , \\ \mathbf{D}_\delta &= \mathbf{D}_q \end{aligned} \quad (5)$$

Although the matrices obtained in (5) are mathematically correct, the usage of it is not recommended due to the poor numerical properties in the q -domain representation. Therefore, Middleton and Goodwin have proposed a procedure for obtaining δ state space model directly from the continuous model [5]. Thus, they suggest the following relations for conversion from s -domain model into the discrete time δ -domain one:

$$\begin{aligned} \mathbf{A}_\delta &= \frac{e^{\mathbf{A}_c T_s} - I}{T_s} = \mathbf{\Omega} \mathbf{A}_c \\ \mathbf{B}_\delta &= \mathbf{\Omega} \mathbf{B}_c , \\ \mathbf{C}_\delta &= \mathbf{C}_c \\ \mathbf{D}_\delta &= \mathbf{D}_c \end{aligned} \quad (6)$$

where $\mathbf{A}_c, \mathbf{B}_c, \mathbf{C}_c, \mathbf{D}_c$ are continuous-time state space model matrices and:

$$\begin{aligned} \mathbf{\Omega} &= \frac{1}{T_s} \int_0^{T_s} e^{\mathbf{A}_c \tau} d\tau = \frac{1}{T_s} (e^{\mathbf{A}_c T_s} - I) \mathbf{A}_c^{-1} \\ &= I + \frac{\mathbf{A}_c T_s}{2!} + \frac{\mathbf{A}_c^2 T_s^2}{3!} + \dots \end{aligned} \quad (7)$$

The correspondence between the two domains is emphasized in the limit case:

$$\begin{aligned} \lim_{T_s \rightarrow 0} \mathbf{A}_\delta &= \mathbf{A}_c \\ \lim_{T_s \rightarrow 0} \mathbf{B}_\delta &= \mathbf{B}_c \end{aligned} \quad (8)$$

It is easy to notice that for the limit case $T_s \rightarrow 0$, there is no equivalence between the matrices obtained in the q discrete time domain and s -domain as one would expect:

$$\begin{aligned} \lim_{T_s \rightarrow 0} \mathbf{A}_q &= \mathbf{I} \\ \lim_{T_s \rightarrow 0} \mathbf{B}_\delta &= 0 \end{aligned} \quad (9)$$

An advantage of modeling linear systems in the δ -

domain consists in the fact that it provides an exact discrete-time representation of the system [8]. Moreover, the parameters of the identified model approach the continuous time values as the sampling time tends to zero.

III. STATE SPACE APPROACH OF THE GPC DESIGN IN δ -DOMAIN

Considering the deterministic case of single input single output δ -domain state space model [8], [12] with the known states unaffected by disturbance or noise:

$$\begin{aligned}\delta \mathbf{x}_k &= \mathbf{A}_\delta \mathbf{x}_k + \mathbf{B}_\delta \mathbf{u}_k, \\ \mathbf{y}_k &= \mathbf{C}_\delta \mathbf{x}_k,\end{aligned}\quad (10)$$

with $\mathbf{x}_k \in \mathbb{R}^n$, $\mathbf{u}_k \in \mathbb{R}^m$, $\mathbf{y}_k \in \mathbb{R}^p$ the state vector, the control vector and the output vector, respectively. Proceed from this model, the j -th order δ derivatives state are obtained as follows:

$$\delta^j x_k = \mathbf{A}_\delta^j x_k + \sum_{i=0}^{j-1} \mathbf{A}_\delta^{j-i-1} \mathbf{B}_\delta \delta^i u_k, \quad (11)$$

with $j = \overline{0, N_y}$, N_y being the prediction horizon. Using the model (3) the following δ derivative predictors are estimated in the δ -domain:

$$\delta^j y_k = \mathbf{C}_\delta \mathbf{A}_\delta^j x_k + \sum_{i=0}^{\min\{j, N_y\}-1} \mathbf{C}_\delta \mathbf{A}_\delta^{j-i-1} \mathbf{B}_\delta \delta^i u_k. \quad (12)$$

In a matrix notation the expression of δ derivatives predictors can be written as:

$$\hat{\mathbf{y}}_\delta = \mathbf{f} + \mathbf{G} \mathbf{u}_\delta, \quad (13)$$

where

$$\begin{aligned}\mathbf{u}_\delta &= [u_k \quad \delta u_k \quad \delta^2 u_k \dots \delta^{N_u-1} u_k]^T \\ \hat{\mathbf{y}}_\delta &= [y_k \quad \delta y_k \quad \delta^2 y_k \dots \delta^{N_y} y_k]^T.\end{aligned}$$

$\mathbf{G}$ is the expanded Toeplitz matrix containing the δ -based Markov parameters and it has the dimension $(N_y + 1) \times N_u$:

$$\mathbf{G} = \begin{bmatrix} g(0,0) & \dots & g(0, N_u-1) \\ \vdots & \ddots & \vdots \\ g(N_y, 0) & \dots & g(N_y, N_u-1) \end{bmatrix}, \quad (14)$$

where

$$g(j, i) = \begin{cases} \mathbf{C}_\delta \mathbf{A}_\delta^{j-i-1} \mathbf{B}_\delta, & 0 \leq i \leq \min\{j, N_u\} - 1 \\ 0, & \text{otherwise,} \end{cases} \quad (15)$$

and $\mathbf{f}$ the free response:

$$\mathbf{f} = [\mathbf{C}_\delta \mathbf{A}_\delta^1 \quad \dots \quad \mathbf{C}_\delta \mathbf{A}_\delta^{N_y}]^T \mathbf{x}_k \quad (16)$$

The optimal control sequence is obtained by minimizing an objective function, knowing the reference trajectory r_{k+i} :

$$J = \sum_{i=N_1}^{N_y} [\hat{\mathbf{y}}_{k+i} - \mathbf{r}_{k+i}]^2 + \lambda \sum_{i=1}^{N_u} [\Delta \mathbf{u}_{k+i-1}]^2, \quad (17)$$

where N_u is control horizon, N_l is minimum costing horizon and λ is control weighting factor. In order to obtain the optimal control sequence in the δ -domain, the set of

vectors that arise in criterion function are obtained from mapping the q -domain terms into the δ -domain through binomial expansion (2).

The control increments are preferred instead of absolute values to avoid steady state error. In the classical discrete time domain the control increments are subject to:

$$\Delta u_{k+j} = u_{k+j} - u_{k+j-1} = 0, \quad j > N_u. \quad (18)$$

In the δ -domain, the constraints on control increments are obtained as follows [12]:

$$\begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+N_u+1} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & -1 & 1 \end{bmatrix}}_{\mathbf{Q}} \mathbf{K}_u \begin{bmatrix} \delta^0 u_k \\ \delta u_k \\ \delta^2 u_k \\ \vdots \\ \delta^{N_u} u_k \end{bmatrix} + \begin{bmatrix} -u_{k-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (19)$$

where $\mathbf{K}_u$ is the matrix obtained from the conversion (2) analogous with $\mathbf{K}_y$:

$$\begin{bmatrix} q^0 y_k \\ q^1 y_k \\ \vdots \\ q^j y_k \\ \vdots \\ q^{N_y} y_k \end{bmatrix} = \underbrace{\begin{bmatrix} T_s^0 & 0 & 0 & \dots & 0 \\ T_s^0 & 1T_s^1 & 0 & \dots & 0 \\ T_s^0 & 2T_s^1 & T_s^2 & \dots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots \\ T_s^0 & \frac{j!}{(j-1)!} T_s^1 & \dots & \dots & \frac{j!}{(j-i)!} T_s^i \\ \left(\begin{smallmatrix} N_y-1 \\ 0 \end{smallmatrix} \right) T_s^0 & \left(\begin{smallmatrix} N_y-1 \\ 1 \end{smallmatrix} \right) T_s^1 & \left(\begin{smallmatrix} N_y-1 \\ 2 \end{smallmatrix} \right) T_s^2 & \dots & \left(\begin{smallmatrix} N_y-1 \\ N_y-1 \end{smallmatrix} \right) T_s^{N_y-1} \end{bmatrix}}_{\mathbf{K}_y} \begin{bmatrix} \delta^0 y_k \\ \delta^1 y_k \\ \vdots \\ \delta^j y_k \\ \vdots \\ \delta^{N_y} y_k \end{bmatrix} \quad (20)$$

The relation between the two discrete domains with respect to constraint (18) is easy to obtain using two components

$$\text{of: } \mathbf{u}_\delta = [\mathbf{u}_\delta^1 \quad \mathbf{u}_\delta^2]^T$$

$$\begin{aligned}\mathbf{u}_\delta^1 &= [\delta^0 u \quad \delta^1 u \quad \dots \delta^{N_u} u]^T \\ \mathbf{u}_\delta^2 &= [\delta^{N_u+1} u \quad \delta^{N_u+2} u \quad \dots \delta^{N_y-1} u]^T.\end{aligned} \quad (21)$$

The control increments can be expressed in the δ -domain as [12]:

$$\Delta \mathbf{u}_{k+i-1} = \mathbf{Q} \mathbf{K}_u \mathbf{u}_\delta^1, \quad i = \overline{1: N_u}. \quad (22)$$

Once arrived at the control horizon value, no adjustment can be done in control sequence, so that the command action is kept constant on the entire left interval. Therefore it is necessary to impose the condition:

$$u_{k+i} = u_{k+N_u}, \quad i = \overline{N_u+1, N_y-1} \quad (23)$$

or

$$\begin{bmatrix} u_{k+N_u+1} \\ u_{k+N_u+2} \\ \vdots \\ u_{k+N_y-1} \end{bmatrix} = \begin{bmatrix} u_{k+N_u} \\ u_{k+N_u} \\ \vdots \\ u_{k+N_u} \end{bmatrix} = \bar{\mathbf{I}} \cdot \mathbf{u}_\delta^1, \quad (24)$$

with $\bar{\Gamma} = \begin{bmatrix} \begin{pmatrix} N_u \\ 0 \end{pmatrix} T_s^0 & \begin{pmatrix} N_u \\ 1 \end{pmatrix} T_s^1 & \dots & \begin{pmatrix} N_u \\ N_u \end{pmatrix} T_s^{N_u} \end{bmatrix}$.

From (21) and (23) one may obtain:

$$\Gamma_u \mathbf{u}_\delta^1 + \Gamma_y \mathbf{u}_\delta^2 = \bar{\Gamma} \mathbf{u}_\delta^1 \quad (25)$$

where Γ_u , Γ_y are sub matrices of $\mathbf{K}_u$ with the number of columns equals the dimensions of $\mathbf{u}_\delta^1$ and $\mathbf{u}_\delta^2$, respectively so that:

$$\mathbf{u}_\delta^2 = \Gamma_y^{-1} (\bar{\Gamma} - \Gamma_u) \mathbf{u}_\delta^1. \quad (26)$$

The cost function has an important role in designing predictive control strategies and in the δ -domain can be expressed similarly [8], [12]. The predictor expression (12) is finally rewritten:

$$\hat{\mathbf{y}}_\delta = \mathbf{f} + [\mathbf{G}_u + \mathbf{G}_y \Gamma_y^{-1} (\bar{\Gamma} - \Gamma_u)] \mathbf{u}_\delta^1, \quad (27)$$

with

$$\mathbf{G}_u \in \mathbb{R}^{n(N_y+1) \times m(N_u+1)} \text{ and } \mathbf{G}_y \in \mathbb{R}^{n(N_y+1) \times m(N_y-N_u-1)} \quad (28)$$

Introducing the constraint (26) and the predictor (27) into the criterion function (16) and differentiating with respect to $\mathbf{u}_\delta^1$ produces the optimal strategy:

$$\frac{dJ_\delta}{d\mathbf{u}_\delta^1} = 0 \Rightarrow \mathbf{u}_\delta^1 = -(\mathbf{H}^T)^{-1} \mathbf{L}. \quad (29)$$

where:

$$\begin{aligned} \mathbf{f}_0 &= (\mathbf{f} - \mathbf{w})^T \mathbf{K}_y^T \mathbf{K}_y (\mathbf{f} - \mathbf{w}), \\ \tilde{\mathbf{G}} &= \mathbf{G}_u + \mathbf{G}_y \Gamma_y^{-1} (\bar{\Gamma} - \Gamma_u), \\ \mathbf{H} &= \tilde{\mathbf{G}}^T \mathbf{K}_y^T \mathbf{K}_y \tilde{\mathbf{G}} + \lambda \mathbf{K}_u^T \mathbf{Q}^T \mathbf{Q} \mathbf{K}_u, \\ \mathbf{L} &= \tilde{\mathbf{G}}^T \mathbf{K}_y^T \mathbf{K}_y (\mathbf{f} - \mathbf{w}). \end{aligned} \quad (30)$$

The reference vector is assumed to be of the form:

$$\mathbf{w}_\delta = [\delta^0 w \quad \delta^1 w \quad \delta^2 w \dots \delta^{N_y} w]^T. \quad (31)$$

The State Space δ GPC algorithm is remarkable by the fact that the strategy for determining the optimal control is fully developed in the δ -domain. The numerical calculations are performed in the δ discrete time representation. Therefore, the rounding errors that occur in the classic q -domain are reduced in this case, especially for small sampling periods in the context of finite number of bits representation as it will be illustrated in the following.

IV. THE DELTA DOMAIN POWERTRAIN STATE SPACE MODEL

The structure of a powertrain system illustrated in Fig. 1 consists of the following parts: engine, clutch, transmission, final drive, driveshafts and wheels [13]. Starting from the affine model and considering different working modes of the clutch a piecewise affine model was obtained. There are two working modes for the clutch: open and closed. The closed mode is defined by three

different phases corresponding to the stiffness of the springs that are being compressed [13]. The model used for this study corresponds to the second phase of the clutch closed mode, when the springs with the smaller stiffness are fully compressed and the springs with medium stiffness begin to compress. In this situation, the clutch stiffness has the value of $k_c = 1600 \text{ Nm/rad}$. In order to apply the State Space δ GPC algorithm, an accurate powertrain model with three rotational inertias is required to predict the vehicle's response to a torque input. The state space model (33)-(34) with the affine term is converted into the δ representation. This model will be used to design and simulate the predictive control strategy in the δ -domain. Moreover, the biggest challenge derives from the fact that implementation of the predictive control algorithm should be done using finite word length representation because of the limited word length of the microprocessors.

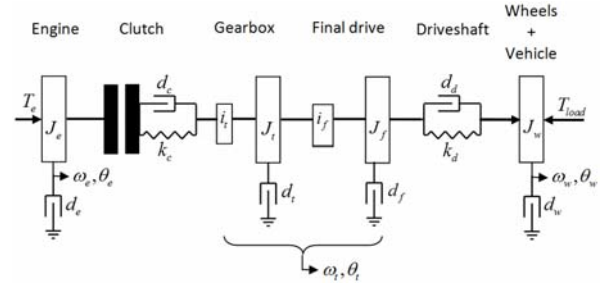


Fig. 1 Three inertias powertrain model

Considering the torsion angle between engine and transmission, the torsion angle between transmission and wheels, the angular speed of the engine, the angular speed of the transmission and the angular speed of the wheel as state variables, i.e:

$$\begin{aligned} x_1 &= \theta_e - \theta_t i_t \\ x_2 &= \theta_t / i_f - \theta_w \\ x_3 &= \omega_e \\ x_4 &= \omega_t \\ x_5 &= \omega_w \end{aligned}, \quad (32)$$

the affine state-space model is:

$$\dot{\mathbf{x}} = \mathbf{A}_c \mathbf{x} + \mathbf{B}_c \mathbf{u} + \mathbf{f}_c. \quad (33)$$

The system matrices $\mathbf{A}_c$, $\mathbf{B}_c$ and the affine term $\mathbf{f}_c$ are.:

$$\mathbf{A}_c = \begin{bmatrix} 0 & 0 & 1 & -i_t & 0 \\ 0 & 0 & 0 & \frac{1}{i_f} & -1 \\ -\frac{k_c}{J_1} & 0 & -\frac{d_c + d_e}{J_1} & \frac{d_c i_t}{J_1} & 0 \\ \frac{k_c i_t}{J_2} & -\frac{k_d}{i_f J_2} & \frac{d_c i_t}{J_2} & -\frac{d_{not}}{J_2} & \frac{d_d}{i_f J_2} \\ 0 & \frac{k_d}{J_3} & 0 & \frac{d_d}{i_f J_3} & -\frac{(d_w + d_d + c_{r2})}{J_3} \end{bmatrix} \quad (34)$$

$$\mathbf{B}_c = \begin{bmatrix} 0 & 0 & \frac{1}{J_1} & 0 & 0 \end{bmatrix}^T, \mathbf{f}_c = \begin{bmatrix} 0 & 0 & 0 & 0 & -\frac{T_{roll}}{J_3} \end{bmatrix} \quad (35)$$

$$\text{where } d_{not} = \left(d_c i_t^2 + d_2 + \frac{d_d}{i_f^2} \right).$$

The coefficients signification and all the values of the parameters are given in [13]. The input of the system is the engine torque $u = T_e$ and the controlled output is represented by the wheel speeds measured in km/h .

With the matrices calculated with relations (6), the state space δ -domain model (10) was used. The parameter values are derived from measurement data and the sampling time depends on engine speed, the sampling time of the ECU (engine control unit) and therefore of the powertrain system. The higher the sampling rate used, the better is the resolution of the result obtained. Sampling theorem states that the minimum sample frequency required to avoid aliasing needs to be at least twice the highest frequency component, which is also valid in δ -domain. The sampling rate up to 50 times is allowable when using a highly efficient computer. In the simulations, the sample time used for system discretization was set to $T_s = 0.02s$.

V. SIMULATION RESULTS

This section presents the performances of the GPC strategy designed in the δ -domain, investigated on the vehicle powertrain model considering finite word length representation. Several experimental studies encourage the model predictive control to work in practice especially for small sampling subject to a finite word length representation.

For assessing the performance of predictive control in discrete δ -domain applied on automotive transmission, some simulation experiments have been performed. The Fig. 2 illustrates the comparisons of engine torque and wheel speed using both SS δ GPC and classical GPC strategy. The initial states used in the simulations have the values: $x = [0 \quad 0 \quad 62.831 \quad 17.952 \quad 4.852]$ and the command action was initially set at $T_e = 9.99 Nm$. The control parameters were set to $N_u = 1, N_y = 9$ and $\lambda = 0.0001$ in each case. In the experimental studies SS δ GPC appears to be very much less sensitive to changes in control horizon and prediction horizon than the classical GPC.

The controller has to satisfy some physical constraints on input and states, so the engine torque is restricted by lower and upper bounds:

$$\begin{aligned} 0 &\leq u_k \leq T_e^{\max}, \\ -T_e^{\Delta} &\leq \Delta u_k \leq T_e^{\Delta}, \end{aligned} \quad (36)$$

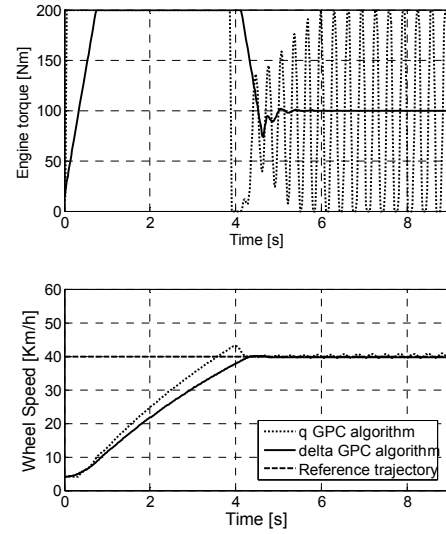


Fig. 2 Comparison of control signal and system output using both δ and q domain GPC strategy - best results for SS δ GPC

where $T_e^{\max}$ is the maximum torque that can be generated by the internal combustion engine, $\Delta u_k = u_k - u_{k-1}$, for all $k \in \mathbb{Z}_{\geq 1}$ and T_e^{Δ} is the maximum increase (or decrease) in torque at each sampling instant. In the simulations these constraints were set to $T_e^{\max} = 200 \text{ Nm}$ and $T_e^{\Delta} = 5$.

The engine torque represented in the first figure is referred as input signal and the output, the wheel speed which must follow the reference trajectory drawn with dotted line, in the second figure.

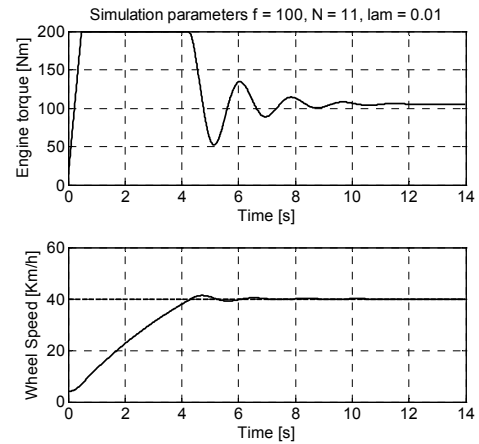


Fig. 3 Simulation results of SS δ GPC algorithm using 16 bits fixed point

As it is shown in the Fig. 2, the SS δ GPC performs much smoother without oscillations than the classical GPC algorithm. The wheel speed reaches the desired value in 4.4 seconds. Many simulation studies have been performed using different predictive control parameters. The results demonstrate that the SS δ GPC is demonstrably sensitive to the variation in control weighting factor and is less

influenced by the variation of prediction horizon. Fig. 3 shows the simulation results considering the sampling frequency of $f = 100\text{Hz}$, prediction horizon of $N_y = 11$ and $\lambda = 0.01$ on 16 bits fixed point quantization. An advantage of working in the δ -domain is that numerical properties are typically improved at fast sampling especially when using finite word length to perform numerical calculations. In order to illustrate the effects of using δ operator, the error in control action is obtained between a floating point implementation and finite word length of 9, 10, 16, 19 bits.

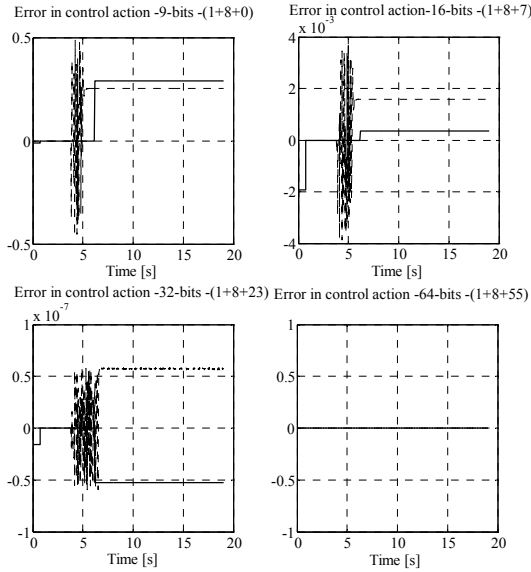


Fig. 4 The comparison of error in the control signal using both q and δ operator digitizing at word lengths: 9, 16, 32, 64 for $\lambda \rightarrow 0$

Matlab is a 64-bit floating point computation engine that provides results in decimal numbers. Therefore, fixed-point binary operations must be simulated in Matlab. The function determines the number of L bits for the magnitude wise largest coefficient and then assigns N-L bits to the fraction part. The resulting number is returned in B. Thus all coefficients have the same L+B+1 bit pattern. According to the coefficient quantization method [10], the range of number representation is from 0 to $(2^N - 1) / 2^B = 2^L - 2^{-B}$. Therefore, all the coefficients smaller than $2^{-15} = 3.0518 \cdot 10^{-5}$ will be truncated to 0. The error in control action is illustrated for both the q (dotted line) and δ GPC (solid line) case as is shown in the Fig. 4. The results demonstrate that the error is comparable between δ and q at all sample word length, but that δ is more accurate when the weighting factor tends to 0 while digitizing at word lengths: 9, 16, 32, 64. The δ operator encodes the free response by numerical differencing, the value of which is not likely to be numerically similar and hence are more robust against rounding effects.

VI. CONCLUSION

The main objective of this paper was to analyze by comparisons the SS δ GPC and the equivalent classical GPC algorithm under condition of fast sampling combined with different word length digitization. In order to illustrate the effects of using δ operator, the error in control action is obtained between a floating point implementation and finite word length of 9, 10, 16, 19 bits. Moreover, SS δ GPC exhibits at high sampling rates lower computational complexity and better numerical conditioning.

ACKNOWLEDGMENT

This work was supported by CNCSIS-UEFISCSU, project number PN II-RU PD COD 331/2010.

REFERENCES

- [1] H. J. Ferreau, P. Ortner, P. Langthaler, Luigi del Re, M. Diehl. "Predictive control of a real-world Diesel engine using an extended online active set strategy", *Annual Reviews in Control*, 2007.
- [2] T. Song and E. G. Collins, "Improved Closed Loop Stability for Fixed Point Controller Realizations using the Delta Operator", *International Journal of Robust and Nonlinear Control*, Vol. 11, 2001, pp. 41 - 57.
- [3] H.C. Reddy, K. Hung, G.S. Moschytz "Realization of Sampled Data Filters with Infinite Transmission Zeros Using Delta Operator", *European Conference on Circuit Theory and Design*, august 28-31, (2001). Espoo, Finland.
- [4] C. Cai, Z. Yun, Z. Duanjin, "State Feedback Design for singularly perturbed system using unified approach", *Journal of Control Theory and Application*, 2004, pp. 276 - 280.
- [5] R. Middleton, G. Goodwin "Improved finite word length characteristics in digital control using delta operators, *IEEE Transactions on Automatic Control*, vol. 31, 1986.
- [6] M. A. Chadwick, S. R. Anderson, V. Kadiramanathan, "An iterative Kalman smoother/least-squares algorithm for the identification of delta-ARX models" *International Journal of Systems Science*, Vol 41, Issue 7, 2010, pp. 839 - 851.
- [7] J.Wu, S.Chen, G.Li, R.H.Istepanian, J.Chu, "Shift and Delta Operator Realizations for Digital Controllers with Finite-Word-Length Considerations", *CiteSeer*, 2000, Pennsylvania State University.
- [8] C (Halaucă) Budaciu., Lazăr C "Delta Domain Predictive Control for Fast, Unstable and Non-minimum Phase Systems", *Control Engineering and Applied Informatics*. Vol 10, Nr. 4, 2008 pp. 26 - 31.
- [9] G Li and M. Gevers, "On the structure of digital controllers in sampled data systems with FWL consideration" In *Proceedings of the 35th IEEE Conference on Decision and Control* (Kobe, Japan), 1996, pp. 919 - 920.
- [10] H-M Cheng and G. T Chiu, Coupling between Sample Rate and Required Wordlength for finite Precision Controller Implementation with Delta Transform", *Proceedings of the 2007 American Control Conference*, New York, July 11-13, 2007.
- [11] D.W Clarke., C Mohthadi., P.S Tuffs, "Generalised predictive control-Part I. The basic algorithm, Part II. Extensions and Interpretations", *Automatica*, 23, 1987 pp. 137-160.
- [12] V. Kadiramanathan, C. (Halaucă) Budaciu and Anderson S. Predictive control of fast-sampled systems using the delta-operator, *International Journal of Systems Science*, 40:7, 2009, pp 745 - 756.
- [13] C. F Caruntu, A. E. Balau, M. Lazar, P.P.J.v.d. Bosh and S. Di Cairano, A predictive control solution for driveline oscillations damping, Accepted to Hybrid system: computation and Control (HSCC), Chicago, USA, 2011.

Interval observers used in state estimation for a class of biodegradable polymers

Elena Bunciu, Emil Petre, *Member, IEEE*, and Viorel Adrian Stanciu

Abstract — This paper deals with the design of interval observers for estimation of some state variables in an uncertain biotechnological process. More exactly, using some interval observers, two biomass concentrations in a process that produces biodegradable polymers are estimated. Since the modelling of these processes is uncertain only some bounds of the two biomass concentrations can be estimated by using two appropriately interval observers. These limits depend on the known limits of the influent substrate concentration involved in the process. These observers are used in an adaptive controlling problem of a substrate inside of the reactor. The effectiveness and performance of these interval observers as well as of the adaptive control strategy are illustrated by numerical simulations applied in the case of a Poly- β -hydroxybutyrate bioprocess.

I. INTRODUCTION

IN the last decades the control of biotechnological processes has been and remains to this day an important problem attracting wide attention. Applications of biotechnology can be found in many domains, but especially in the food industry, pharmaceutical production, wastewater treatment and depollution processes. In contrast with other kinds of processes that are perfectly described by physical laws - like mechanical or electrical systems - biotechnological processes are dealing with living organisms and therefore, their mathematical models are uncertain and are known to have a lower aptitude to accurately match experimental results [1].

The properties of living organisms evolve in a long time or at the smallest change in their environment. The organisms can become highly sensitive to such modifications. That is why the bioprocesses are characterized by highly nonlinear and uncertain dynamics and their mathematical modelling is sophisticated and complex [2].

A mathematical model for a biotechnological system can be constructed for various objectives like the need to predict the behaviour of this biotechnological system, or to estimate key parameters, or to develop and test a control strategy or to estimate variables through observers [2]. Online monitoring of these kinds of processes is a rather complicated task. This has happened because one of the main limitations for improving monitoring and optimisation of bioreactors is to

measure chemical and biological variables [3].

Over the last ten years, the difficulties encountered in the measurement of state variables imposed the development of a series of techniques from which could be mentioned: state estimation by means of convex sets, interval observers and bounded error estimators using interval analysis [4]. It must be noted that the choice of the design method depends on the kind of available models. So, when a correctly identified and validated model is available, a high gain observer may perform good estimations of the internal state, while if the parameters, inputs and measurements of the model have large uncertainties, interval estimations should give us the best approach [5].

Interval observers were introduced in a general context by using the concept of framer, which is defined as a unique pair of estimates able to provide uncertain state limits, without taking into consideration any stability constraint [5].

Interval observers work under the formulation of two observers: an upper observer, which produces an upper bound of the state vector, and a lower observer producing a lower bound, providing by this way a bounded interval in which the state vector is guaranteed to lie [5]. For the formulation of the interval observer, it is necessary to know bounds of the uncertainties in the model (i.e. uncertainties in model parameters, input variables, etc.) [1].

The theoretical foundation of an interval observer is the theory of cooperative systems. Such systems have the property to keep the order between 2 trajectories, and they can thus provide bounds of the behaviour of the state vector, depending on the bounds of the uncertainties in the model [1].

In this paper the design and the analysis of some interval observers for estimation of some state variables in a strongly nonlinear and uncertain biotechnological process is presented. More exactly, for a class of biodegradable polymers, namely in the case of a Poly- β -hydroxybutyrate bioprocess, by using some appropriately interval observers, two biomass concentrations are estimated. Since the modelling of this process is uncertain only some bounds of the two biomass concentrations can be estimated. These limits depend on the known limits of the influent substrate concentration implied in process. The proposed observers are used in an adaptive controlling problem of a substrate inside of the reactor. The effectiveness and performance of these interval observers are illustrated by numerical simulations applied in the case of the above mentioned bioprocess.

The paper is organized as follows. In Section 2 the mathematical model of a class of mixed processes is presented. The problem of estimation of some immeasurable states of a mixed process is presented and analysed in Section 3. An adaptive control strategy of analyzed mixed

Elena Bunciu is with the Automation Department within the Faculty of Automation, Computers and Electronic, Craiova, Romania; (e-mail: bunciu.elena@yahoo.com).

Emil Petre is with the Automation Department within the Faculty of Automation, Computers and Electronic, Craiova, Romania; (e-mail: epetre@automation.ucv.ro).

Viorel Adrian Stanciu is with the Mechatronics Department within the Faculty of Automation, Computers and Electronic, Craiova, Romania; (e-mail: stanciu@robotics.ucv.ro).

process is formulated and presented in Section 4. Concluding remarks in Section 5 finish the paper.

II. MATHEMATICAL MODELS

In this section we briefly present a major class of the biodegradable polymers category that is the class represented by the family of polyhydroxyalkanoates (PHAs) [5] which belong to the large class of mixed cultures. In this class an important member is PHB (Poly- β -hydroxybutyrate), which has some properties that make it potentially competitive with synthetic polymers, being a biodegradable material [6, 7].

Mixed or co-culture systems are fermentation systems, where microorganism assimilate one substrate and convert it into an intermediate metabolite which is converted from other microorganisms to a metabolite (final product) [10].

In the case of PHB production sugars such as glucose obtained from food processing waste were converted into lactate by *Lactobacillus Delbrueckii* and lactate is converted into poly- β -hydroxybutyrate by *Ralstonia Eutropha* strain E2 (previously *Alcaligenes* sp.) inside one bioreactor. Since this bioprocess is strongly nonlinear and time-varying the control design of such process is a very difficult problem [10].

In the last few years, several mathematical models have been developed for the lactic acid fermentation. In more recent studies [11] the mathematical model that describes the dynamics for cultivation of *L. Delbrueckii* in a mixed culture, obtained from the mass balance of the reaction components, has the form described by the following three nonlinear differential equations [10, 11]:

$$\dot{X}_1 = \mu_1(S, P, DO)X_1 - \frac{F}{V}X_1 \quad (1)$$

$$\dot{S} = -v_1(S, P, DO)X_1 + \frac{F}{V}(S_{in} - S) \quad (2)$$

$$\dot{P} = \sigma_1(S, P, DO)X_1 - v_2(S, P, D)X_2 - \frac{F}{V}P \quad (3)$$

where X_1 , S and P are respectively the concentrations of *L. Delbrueckii* bacteria (Biomass 1), of the consumption glucose (Substrate 1) and of the synthesis product, and μ_1 , v_1 and σ_1 are respectively the specific growth rate of *L. Delbrueckii* (biomass X_1), the specific glucose consumption rate and the specific lactate production rate. These specific reaction rates are strongly nonlinear and are described as follows [10, 11]:

$$\mu_1(S, P, DO) = \frac{\mu_{m1} \cdot S}{K_S + S} \left(1 - \frac{P}{P_m} \right) \quad (4)$$

$$v_1(S, P, DO) = \frac{\sigma_1(S, P, DO)}{Y_{P/S}} \quad (5)$$

$$\sigma_1(S, P, DO) = \alpha\mu_1(S, P, DO) + \beta(S, DO) \quad (6)$$

$$\beta(S, DO) = \frac{\beta_m \cdot DO \cdot S}{K_S + S} \quad (7)$$

In equation (3) the term $(-v_2(S, P, D)X_2)$ is presented, which is due to the substrate consumption by *Ralstonia Eutropha* in the mixed culture [10, 11], where the specific consumption rate is modelled as follows:

$$v_2(S, P, DO) = \frac{\mu_2(P, DO, N)}{Y_{X_2/P}} \quad (8)$$

For the modelling on *R. Eutropha* growth, only a limited number of papers have been published. Here we use the model presented in [10, 11] where the *R. Eutropha* growth process is described by the following three nonlinear differential equations:

$$\dot{X}_2 = \mu_2(P, DO, N)X_2 - \frac{F}{V}X_2 \quad (9)$$

$$\dot{N} = -v_3(S, P, DO)X_2 - \frac{F}{V}N \quad (10)$$

$$\dot{Q} = \sigma_2(N)X_2 - \frac{F}{V}Q \quad (11)$$

where X_2 , N , Q and DO are respectively the concentrations of *R. Eutropha* bacteria (Biomass 2), of the ammonia (Substrate 2), of PHB and of the dissolved oxygen, and σ_2 is specific PHB production rate μ_2 is specific growth rate of *R. Eutropha* and v_2 is specific substrate consumption rate. In the model (9)-(11), the expressions of these specific reaction rates are described by the following expressions [11]:

$$\mu_2(P, DO, N) = \frac{\mu_{m2} \cdot P}{K_P + P + P^2/K_i} \left(\frac{N}{K_N + N} \right) \quad (12)$$

$$v_3(S, P, DO) = \frac{\mu_2(P, DO, N)}{Y_{X_2/N}} \quad (13)$$

$$\sigma_2(N) = q_m \left(\frac{K_N}{K_N + N} \right) \quad (14)$$

In these equations the parameters μ_2 and σ_2 express experimental phenomena in which the cell growth is enhanced when ammonia concentration (N) increases and when N decreases, the poly- β -hydroxybutyrate production is enhanced [10].

The numerical values of the kinetic and process parameters are [12]:

$\alpha = 1.23$	$\beta_m = 1.80$	$\mu_{m1} = 0.375$
$\mu_{m2} = 0.734$	$K_i = 2.5$	$K_N = 0.146$
$K_P = 6$	$K_S = 35.8$	$P_m = 42.9$
$Y_{X_2/P} = 0.204$	$Y_{P/S} = 0.698$	$Y_{X_2/N} = 2.41$
$DO = 3 \text{ ppm}$		

In Table 1 we summarize the all kinetics and process parameters from equations (1)-(3) and (9)-(11) as well as their physical meaning and physical units.

TABLE I
SYMBOLS SIGNIFICANCE

Symbol	U.M.	Quantity
μ_1	h^{-1}	Specific growth rate of <i>L. Delbrueckii</i>
v_1	h^{-1}	Specific glucose consumption rate
σ_1	h^{-1}	Specific lactate production rate
μ_2	h^{-1}	Specific growth rate of <i>R. Eutropha</i>
v_2	h^{-1}	Specific substrate consumption rate
σ_2	h^{-1}	Specific PHB production rate
DO	ppm	Dissolved oxygen quantity
μ_{m1}, μ_{m2}	h^{-1}	Maximum value for μ_1 and μ_2
V	L	The volume of the bioreactor
F	l/h	Glucose feed rate
F/V	h^{-1}	Dilution rate
S_{in}	g/l	Influent substrate concentration
X_1	g/l	Concentration of <i>L. Delbrueckii</i> (Biomass 1)
X_2	g/l	Concentration of <i>R. Eutropha</i> (Biomass 2)
S	g/l	Substrate concentration
N	g/l	Ammonia concentration
P	g/l	Synthesis product concentration
Q	g/l	PHB concentration
$Y_{P/S}, Y_{X2/N}, Y_{X2/P}$	g/g	Yield coefficients

III. ESTIMATION PROCEDURE

From Section 2 it can be seen that the mixed bioprocess of PHB production is a very sophisticated and complex process. As a consequence the monitoring and control of such process is a difficult task. This process includes variables which can be on-line measurable for example glucose concentration as well as variables which can not be measurable such as biomass concentration and synthesis product concentration. Therefore for monitoring or control reasons or even from simulation reasons these variables must to be estimated from available measurements of other process variables.

In the case of the mixed bioprocess of PHB production the concentrations of *L. Delbrueckii* bacteria (Biomass X_1) and of *R. Eutropha* bacteria (Biomass X_2) and synthesis product concentration P are not measurable and therefore these must to be estimated. Then in the following we develop two state observers which will yield on-line estimates of X_1 , X_2 and P .

Note that in this process the influent substrate concentration S_{in} can be considered a perturbation whose exactly values are not necessarily to be known, but in the same time the values of S_{in} can vary between two known limits (an upper limit denoted by S_{in}^+ and a lower limit denoted by S_{in}^-). Knowing these two limits we attempt to estimate all the others immeasurable state variables in model (1)-(3) and (9)-(11) by using of course all available information.

To estimate the values of the biomass concentrations X_1 and X_2 as well as the value of the synthesis product P , first

we introduce the following three additional variables:

$$z_1 = \frac{\alpha}{Y_{P/S}} X_1 + S \quad (15)$$

$$z_2 = \frac{1}{Y_{X2/N}} X_2 + N \quad (16)$$

$$z_3 = \frac{1}{Y_{P/S}} P + S - \frac{Y_{X2/N}}{Y_{X2/P} Y_{P/S}} N \quad (17)$$

The dynamics of z_i , $i=1, 2, 3$ deduced from model (1)-(3) and (9)-(11), are expressed by the following linear stable equations

$$\dot{z}_1 = -\frac{F}{V} z_1 + \frac{F}{V} S_{in} \quad (18)$$

$$\dot{z}_2 = -\frac{F}{V} z_2 \quad (19)$$

$$\dot{z}_3 = -\frac{F}{V} z_3 + \frac{F}{V} S_{in} \quad (20)$$

that are independent of the process kinetics that could be completely unknown.

Because of S_{in} is known only by its upper and lower limits, then from (18) and (20) one can deduce that, in fact, z_1 and z_3 , can only be determined by their upper and lower limits values. For this reason, the real values of X_1 and P are located in an interval that is bounded by these parameters limits.

Then, from (15)-(20), the on-line estimation of X_1 , X_2 and P are given by the following relations:

$$\hat{X}_1^+ = \frac{Y_{P/S}}{\alpha} (\hat{z}_1^+ - S^+) \quad (21)$$

$$\hat{X}_1^- = \frac{Y_{P/S}}{\alpha} (\hat{z}_1^- - S^-) \quad (22)$$

$$\hat{X}_2 = Y_{X2/N} (\hat{z}_2 - N) \quad (23)$$

$$\hat{P}^+ = Y_{P/S} \left(\hat{z}_3^+ + \frac{Y_{X2/N}}{Y_{X2/P} Y_{P/S}} N - S^+ \right) \quad (24)$$

$$\hat{P}^- = Y_{P/S} \left(\hat{z}_3^- + \frac{Y_{X2/N}}{Y_{X2/P} Y_{P/S}} N - S^- \right) \quad (25)$$

where $\hat{z}_i$, $i=1, 2, 3$ are yielded by (18)-(20).

In equations (21)-(23) $\hat{X}_1$, $\hat{X}_2$ and $\hat{P}$ represent the estimations of the two biomasses X_1 and X_2 and of the synthesis product P , and the symbols "+" and "-" denoted respectively the upper and lower limits.

In Fig. 1 is presented a possible form for the real influent substrate. Because we don't know anything about the real shape of influent substrate evolution, we can choose any

form that is placed in a bounded domain ranges between the two limits: an inferior limit denoted by S_{in}^- and a superior limit denoted by S_{in}^+ . Because of only this limits of the influent substrate concentration must to be known, the shape of real influent substrate can be completely random.

The real influent substrate concentration in Fig. 1 is also corrupted with an additive white noise with zero average (5% from their nominal values).

The performance of designed interval observers (21)-(23) has been tested by performing extensive simulation experiments. The simulations were carried out by using the process model (1)-(3) and (9)-(11) under identical and realistic conditions. For the influent substrate concentration we chose the following two limits: $S_{in}^- = 300$ g/l and $S_{in}^+ = 340$ g/l. It can be seen that the signal of influent substrate concentration completed with the noise cannot exceed the interval required.

The time evolution of the estimates of the two unknown biomass concentrations X_1 and X_2 as well as the value of the synthesis product P provided by the observers (21), (22) and (23) is presented in Fig. 2, Fig. 3 and respectively in Fig. 4.

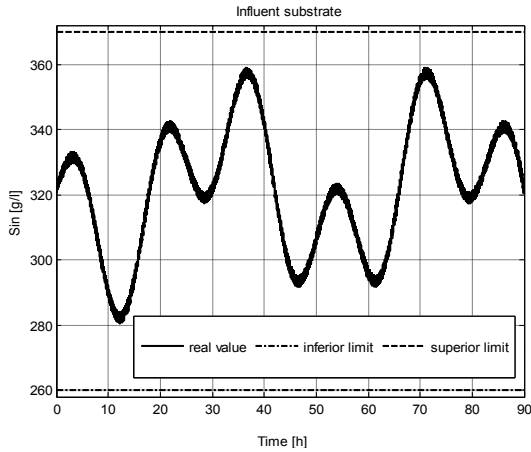


Fig. 1 Evolution of influent substrate concentration S_{in} .

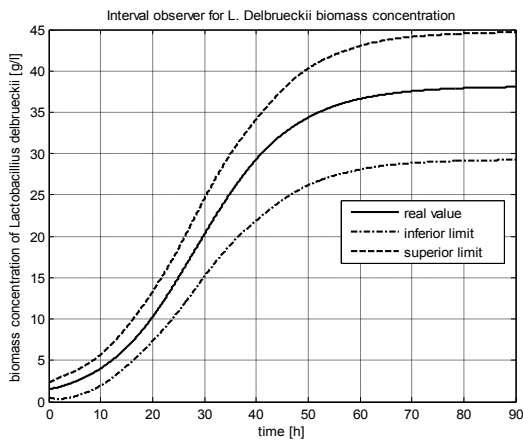


Fig. 2 Profile of estimates of unknown concentration of *L. Delbrueckii* bacteria (Biomass X_1) versus its real value.

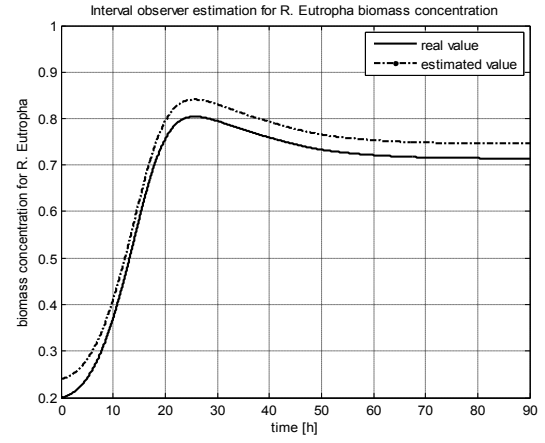


Fig. 3 Profile of estimates of unknown concentration of *R. Eutropha* bacteria (Biomass X_2) – interrupted line versus its real value – continuous line.

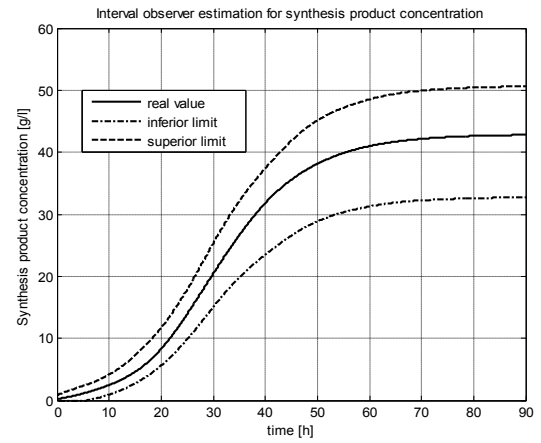


Fig. 4 Profile of estimates of unknown concentration of the synthesis product concentration P versus its real value.

In Fig. 2 and Fig. 4 the graphics marked with continuous line correspond to values of real concentration of S_{in} , as it should be known while the graphics marked with dotted and interrupt lines correspond respectively to known upper and lower limits of S_{in} .

Since the estimated value of the concentration of the biomass X_2 does not depend of S_{in} , in Fig. 3 only the time profile of estimates of unknown concentration of biomass X_2 versus its real value is presented.

Using different shapes of influent substrate concentration that are included between two known limits, it can be seen that the behaviour of the proposed interval observers are good, the values of the keestimated biomass X_1 and of the synthesis product P remaining between the limits determined from equations (21) and (23) or closely to its real value in the case of estimated values of biomass X_2 despite the influence of noisy measurements of concentration of S and N .

IV. CONTROL STRATEGIES

A. Control Objective

In this section we consider the problem of controlling the process in order to produce a large quantity of poly- β -hydroxybutyrate with a minimum cost, by using an appropriately control input under the following assumptions: (i) the reaction rates are known; (ii) all the model parameters are known; (iii) the concentration of the influent substrate S_{in} is completely unknown, but a upper and lower bounds of it are known either by measurements or by user's choice.

B. Exactly Linearizing Feedback Controller

Firstly, we consider the ideal case where maximum prior knowledge concerning the process (kinetics, yield coefficients and state variables) is available, and the relative degree of differential equations in the process model is equal to 1. Assume that for the controlled bioprocess we wish to have the following first order linear stable closed loop (process + controller) behaviour [14]:

$$\frac{d}{dt}(S^* - S) + \lambda_1(S^* - S) = 0 \quad (24)$$

where S^* is the desired values of S and $\lambda_1 > 0$ is a design parameter.

Since the equation (2) in the process model has the relative degree equal to 1, this can be considered as an input-output model [14] with relative degree equal to 1.

Then from (2) and (24) one obtains the following exactly linearizing feedback control law:

$$\frac{F}{V} = \frac{1}{S_{in} - S} (\dot{S}^* + \lambda_1(S^* - S) + \nu_1 X_1) \quad (25)$$

where $D = F/V$ represents the dilution rate and in our case in control law (25) is the input control variable.

The control law (25) applied to the process (2) will result in:

$$\dot{S} = \dot{S}^* + \lambda_1(S^* - S) \quad (26)$$

and leads to a linear error model in the form $\dot{e} = -\lambda_1 e$, where $e = S^* - S$ represents the tracking errors. It is clear that for $\lambda_1 > 0$, this error model has an exponential stable point at $e = 0$.

C. An Adaptive Control Strategy using Interval Observers

In the previous subsection, it was assumed that the functional forms of the nonlinearities as well as process parameters are known. Since such prior knowledge is not realistic, in this subsection we consider a more realistic case that is the case when the biomasses concentration in not measurable. In this case, the control law (25) becomes an adaptive control law of the form:

$$D = \frac{1}{S_{in} - S} (\dot{S}^* + \lambda_1(S^* - S) + \nu_1 \hat{X}_1) \quad (27)$$

where $\hat{X}_1$ is an estimate of X_1 calculated using (18) and (15). Because of for $\hat{X}_1$ only a lower limit and an upper limit can be estimated (see (21)) then we also will obtain two limits that define the control law (a superior and an inferior limit) as following:

$$\begin{aligned} D^+ &= \frac{1}{S_{in}^- - S} (\dot{S}^* + \lambda_1(S^* - S) + \nu_1 X_1^+) \\ D^- &= \frac{1}{S_{in}^+ - S} (\dot{S}^* + \lambda_1(S^* - S) + \nu_1 X_1^-) \end{aligned} \quad (28)$$

Finally, an adaptive linearizing controller is obtained by combination of (28), (18) and (21).

The performance of designed adaptive controller (28), by comparison to the exactly linearizing controller (25) (which yields the best response and can be used as benchmark), has been tested by performing extensive simulation experiments. For a proper comparison, the simulations were carried out by using the process model (1)-(3) and (9)-(11) under identical and realistic conditions.

After a lot of simulation experiments we find that a value around 15 g/l for residual substrate inside the reactor results in a maximal lactic production rate and a large quantity of poly- β -hydroxybutyrate. As a consequence in the presented results the operating point chosen is $S^* = 15 \text{ g/l}$.

The system's behavior was analyzed assuming that the influent glucose concentration is time varying, its real values are unknown, but the inferior and superior limits of its values are known (see Fig. 1).

Considering that in control laws (28) the value of the design parameter $\lambda_1 = 0.5$, the behaviour of adaptive closed-loop system by comparison to the behavior of the closed loop system with exactly linearizing controller is presented in Fig. 5 and Fig. 6. So, in Fig. 5 we represented the evolution of the controlled substrate concentration S for a reference $S^* = 15 \text{ g/l}$, and in Fig. 6 the evolution of input control. In this figure with dotted lines it was represented the inferior limit obtained by replacing the real value of the influent substrate with its superior limit; with interrupted line it was represented the inferior limit of the dilution rate determined by using the upper limit of the influent substrate concentration and the continuous line corresponds to the real value (represented in Fig. 1) and is obtained by using linearizing law (25).

Considering different forms of influent substrate concentration that are included between the two limits, we observed that all values of the estimated biomasses and of the synthesis product remain between the limits determined from equations (21)–(23).

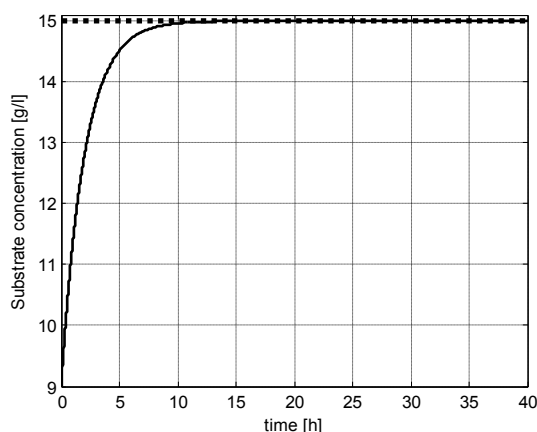


Fig. 5 Evolution of output S (the dotted line represents the reference and the continuous line is the substrate concentration)

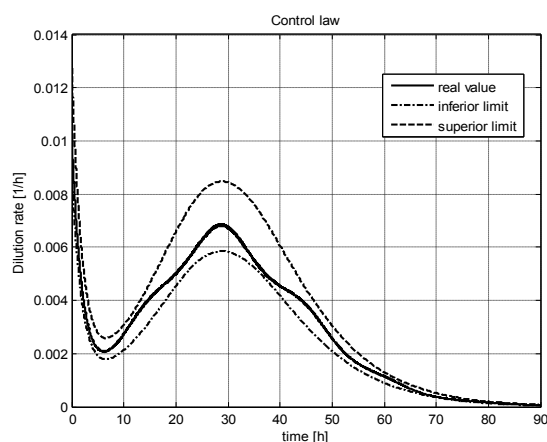


Fig. 6 Evolution of control input D .

V. CONCLUSION

In this paper the design and the analysis of some interval observers for the estimation of some state variables in a strongly nonlinear and uncertain biotechnological process was presented. More exactly, for a class of Poly- β -hydroxybutyrate bioprocess, by using some appropriately interval observers, two biomass concentrations are estimated. Since the modelling of this process is uncertain only some bounds of the two biomass concentrations can be estimated. These limits depend on the known limits of the influent substrate concentration implied in process. These observers were used in a control problem of the substrate inside of reactor. The effectiveness and performance of the presented interval observers as well as of the adaptive control algorithm were illustrated by numerical simulations applied

in the case of the above mentioned bioprocess.

The method can be applied in a large class of bioprocesses like wastewater treatment or anaerobic digestion processes.

ACKNOWLEDGMENT

This work was partially supported by the strategic grant POSDRU/88/1.5/S/50783, Project ID50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectorial Operational Programme Human Resources Development 2007–2013.

REFERENCES

- [1] M. Moisan, O. Bernard, "Interval estimators for non monotone systems. Application to bioprocess models," *6th IFAC World Congress*, Prague, July 3-8, 2005.
- [2] V.A. Gonzales, "Estimation et commande robuste non-linéaires des procedes biologiques de depollution des eaux uses. Application a la digestion anaerobie." *These de l'Université de Perpignan*, 2001.
- [3] O. Bernard, J. L. Gouze, "State estimation for bioprocesses. *Mathematical Control Theory*," *ICTP lecture notes*, 2002.
- [4] M. Moisan, O. Bernard, J.-L. Gouze, "A high/low gain bundle of observers: application to the input estimation of a bioreactor model," *Proc. of the 17th World Congress of International Federation of Automatic Control*, Seoul, Korea, 2008, pp 15547-15552.
- [5] T.V. Ojumu, I. Yu, B.O. Solomon, "Production of Poly-hydroxyalkanoates, a bacterial biodegradable polymer", *African Journal of Biotechnology*, vol. 3, 2004, pp 18-24.
- [6] Kato N., Konishi H, Shimao M., Sakazawa C., "Production of 3-hydroxybutyric acid trimer by *Bacillus megaterium* B-124," *J. Ferment. Bioeng.*, vol. 73, pp 246–247, 1992.
- [7] P.R. Patnaik, "Perspectives in the modeling and optimization of PHB production by pure and mixed cultures," *Crit. Rev. Biotechnol.*, vol. 25, 2005, pp 153–171.
- [8] G. Braunegg, G. Lefebvre, K.L. Genser, "Poly-hydroxyalkanoates, biopolymers from renewable resources: physiological and engineering aspects," *J. Biotechnol.*, vol. 65, 1998, pp. 127–161.
- [9] P. Patnaik, "Dispersion optimization to enhance PHB production in fed-batch cultures of *Ralstonia eutropha*," *Bioresource Technology*, 97, 2006, pp 1994-2001.
- [10] S. Popova, "On-line state and parameters estimation based on measurements of the glucose in mixed culture system," *Journal Biotechnology & Biotechnological Equipment*, vol. 3, no. 20, 2006, pp. 208-214.
- [11] M. Tohyama, T. Patarinska, Z. Qiang, K. Shimuzi, "Modeling of the mixed culture and periodic control for PHB production," *Journal Biochemical Engineering*, 10, 2002, pp.157-173.
- [12] J.H. Law, R.A. Slepecky, "Assay of poly- β -hydroxybutric acid," *Journal Bacterial.*, 1961, pp. 33-36.
- [13] E. Petre, D. Selișteanu, D. Șendrescu, "An Indirect Adaptive Control Strategy for a Lactic Fermentation Bioprocess," *Proceedings of Int. Conf. on Automation, Quality and Testing, Robotics AQTR 2010 (THETA 16)*, May 28-30, 2010, Cluj-Napoca, Romania, Tome I, pp. 175-180.
- [14] E. Petre, *Nonlinear Systems. Applications in Biotechnology*, second edition, Universitaria, Craiova, 2008.

An Image Based Visual Servoing Scheme for Wheeled Mobile Robots

Adrian Burlacu, Obada Mihai and Vladimir Tiganas

Abstract—In this paper a real-time image based visual servoing scheme for a nonholonomic mobile robot is presented. The proposed method takes into account the nonholonomic motion constraint of mobile robots. As a distinctive fact it does not require the estimation and decomposition of the homography or fundamental matrix. A switched image based controller is design, using point features, in order to solve the camera field-of-view (FOV) constraint. The proposed architecture was implemented using a real-time visual servoing system and multiple test were conducted. Experimental results are revealed and commented.

I. INTRODUCTION

Visual servoing takes account of using visual feedback to drive a robot from an initial configuration to the desired one, which is defined by a previously acquired image with a pinhole camera. Traditionally, research on visual servoing focuses on six-degree-of-freedom robot manipulators [1],[2],[3] where in motion constraints are generally not considered. Image-based visual servoing (IBVS) uses direct image measurements as feedback to control the motion of the robot. The robots positioning task is expressed as an image-based error function to be minimized using a suitable control law. Because IBVS does not explicitly solve for the Cartesian pose of the target object, its performance does not depend on the accuracy of a priori models. However, since the domain of the control law is in image space, there is no direct control over the Cartesian or joint-space trajectory of the robot end-effector.

Mobile robots are usually subject to nonholonomic constraints, and thus new challenges emerge for the visual servoing task because nonholonomic systems cannot be asymptotically stabilized by any time-invariant continuous state feedback control law [4]. Besides, the lack of depth information has been an inherent problem for monocular camera-based vision systems, for which the robot state cannot be exactly/completely reconstructed with unknown three-dimensional model of the target. Applying visual servoing techniques for guiding a wheeled mobile robots, involves well known control problems related to the nonholonomic constraint. Firstly, the linearization of these systems is uncontrollable, and, secondly, there do not exist smooth state feedback laws, for stabilizing these systems to an equilibrium. However, visual servoing techniques have been successfully used to control nonholonomic mobile robots. In [5] an experiment study is presented. The study is focused on vision-based control methods for the nonholonomic wheeled

mobile robots. The visual sensor was considered to be fixed and mounted on the ceiling of an indoor environment.

A number of significant papers present methods based on epipolar geometry for designing nonholonomic visual control laws. In [6] an algorithm for image-based navigation is described, the motion control being performed by recovering the relative pose of the robot with respect to the desired one. Another approach based on epipolar geometry is presented in [7]. The method's major contribution is the design of a control law that considers nonholonomic constraints of the robot while having robust feature detection and matching stages based on scale and rotation invariant features. An image-based visual servoing strategy for the autonomous navigation of a mobile holonomic robot from a current towards a desired pose, using catadioptric camera is proposed in [8]. The proposed visual servoing does not require any metrical information about the three-dimensional viewed scene and is mainly based on a geometrical property, named auto-epipolar condition, which occurs when two catadioptric views (current and desired) undergo a pure translation. A motion estimation technique for visual servoing control of a nonholonomic mobile robot is presented in [9]. A hybrid error vector was defined including both image signals and the estimated rotational angle and a smooth time-varying feedback controller is adopted to cope with the nonholonomic constraints.

The image based control algorithm proposed uses point feature type information to guide a mobile visual servoing system towards a desired configuration of a static object. First an image based control law that fulfills the FOV constraint is designed. Then, taking into consideration the nonholonomic constraints, and estimating the depths information of point features by using the kinematic model of the robot combined with planar homogenous transformation, a switched visual controller for three degrees of freedom is designed. A real time visual servoing system was designed and implemented, and the proposed image based controller was tested.

The structure of the paper is : Section II presents the image based control law that fulfills the FOV constraint; in Section III the kinematic model of the robot and the visual guidance architecture are revealed. Section IV is dedicated to experimental results while the conclusions of the paper are presented in Section V.

II. IMAGE BASED CONTROL

The main goal of any image based control architecture (IBVS) is to drive a robot system using information acquired by a visual sensor [2]. If the considered robot system is a manipulator robot that has a video camera mounted on

A. Burlacu, O. Mihai and V. Tiganas are with Faculty of Automatic Control and Computer Science, Automatic Control and Applied Informatics, Technical University of Iasi, D. Mangeron Street, No.27, 700050, Romania. aburlacu@tuiasi.ro

the gripper then a visual servoing structure with an eye-in-hand configuration is created. In order to achieve the main goal, a trajectory for the video camera must be designed. The trajectory is given by the integration of the reference camera velocity obtained from the IBVS architecture by minimizing the error between the current visual features configuration and the desiring one. Usually the control law is design for 6 d.o.f of the camera velocity. In the paper an approach for 3 d.o.f that takes into account the nonholonomic constraints of the mobile platform, is developed.

Image processing algorithms can generate different types of visual features. For objects' description, image moments, contour, lines or point features are the most used. In visual analysis applications point features are low level descriptors precisely locatable and persistent. These basic properties define the usefulness of point features for object description. The error between point features in the image plane is defined as follows:

$$e(t) = f(t) - f^*, \quad (1)$$

where $f(t)$ represents the point features position at time t and f^* the desired configuration. Equation (1) is the general representation of the input signal for the image based controller. The outputs of the image based controller is the reference velocity $\mathbf{v}_c^*$ of the camera. If the task of extracting point features and finding the correspondences in the desired configuration is completed then a relationship between time variation of $f(t)$ and camera velocity can be generated. Let $\mathbf{v}_c^*$ be a vector composed from linear velocities of the camera frame and the instantaneous angular velocities of the camera frame.

The link between the motion of point features and camera velocity is given by [1]:

$$\dot{f} = L\mathbf{v}_c^*, \quad (2)$$

where by $L \in \mathbb{R}^{2n \times m}$ is denoted the image Jacobian of n point features that compose the vector f that has m degrees of freedom.. From (1) and (2) it is obvious that:

$$\dot{e} = L\mathbf{v}_c^*. \quad (3)$$

In order to increase the robustness of the visual based control algorithm and fulfill the FOV constraint a different error was chosen:

$$e_D = De, \quad (4)$$

where $D = \text{Diag}\{d_1, \dots, d_n\}$ is a weighting matrix used to remove the outliers from the point features set. The weights d_i vary from 1 when the robust estimation gives full confidence and 0 when the feature is doubtlessly an outlier. For example an weighting algorithm for establishing the values for $\{d_1, \dots, d_n\}$ can be generated using the distance of each feature toward the center of the image. If the i -th point feature is positioned near the image center, the d_i will have a value close to 1, else if the point feature is near one the edges of the image the value of d_i will be close to 0. Taking into account that D can be considered to vary slowly and thus

$$\dot{e}_D = D\dot{e}. \quad (5)$$

A classical proportional controller is defined by [1]:

$$\mathbf{v}_c^* = -\lambda \widehat{L}^+ e_D, \quad (6)$$

but using (3), (4), (5) a more robust expression can be obtained:

$$\mathbf{v}_c^* = -\lambda (\widehat{DL})^+ De. \quad (7)$$

Let

$$L_f = DL, \quad (8)$$

be the interaction matrix of the detected point features and thus (7) can be expressed:

$$\mathbf{v}_c^* = -\lambda L_f^+ De. \quad (9)$$

In (9) $L_f^+ \in \mathbb{R}^{m \times 2n}$ is the Moore-Penrose pseudoinverse of L_f , that is $L_f^+ = (L_f^T L_f)^{-1} L_f^T$.

III. MOBILE ROBOT VISUAL CONTROL ARCHITECTURE

In this Section the IBVS method described previous will be adapted to control a system composed from a video camera mounted on a two wheel robot. First the robot kinematic model will be presented, while the control algorithm will be revealed in the second part of the section

A. Mobile Robot Model

In terms of locomotion for mobile robots, wheels represent one of the most used mechanism. Any wheeled vehicle is subject to kinematic constraints that reduce in general its local mobility, while leaving intact the possibility of reaching arbitrary configurations by appropriate maneuvers [10]. Nonholonomic constraints are nonintegrable linear constraints over the velocity space of a system. For instance, the rolling without slipping constraint of a differentially driven mobile robot is linear with respect to the velocity vector (vector of linear and angular velocities) of the differentially driven robot and is nonintegrable (it cannot be integrated into a constraint over the configuration variables). As a consequence, a differentially driven mobile robot can go anywhere but not following any trajectory.

Let's consider a two wheel robot with a pose defined by:

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (10)$$

Assuming that the motion is done without sliding, the following equations describe the kinematic behavior of the robot:

$$\begin{aligned} \dot{x} &= v_R \cos \theta \\ \dot{y} &= v_R \sin \theta \\ \dot{\theta} &= \omega_R \end{aligned} \quad (11)$$

where (v_R, ω_R) is the linear and angular velocity of the robot. Having an input (v_R, ω_R) the correspondent velocities of the wheels (right and left) can be computed from:

$$\begin{pmatrix} v_r \\ v_l \end{pmatrix} = \begin{pmatrix} 1 & r \\ 1 & -r \end{pmatrix} \cdot \begin{pmatrix} v_R \\ \omega_R \end{pmatrix} \quad (12)$$

where r is the half distance between the robot's wheels.

From (11) and (12) the following kinematic model, having the left and right velocities of the wheels as inputs, is obtained:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2r} & \frac{1}{2r} \\ \frac{1}{2r} & -\frac{1}{2r} \end{pmatrix} \cdot \begin{pmatrix} v_r \\ v_l \end{pmatrix} \quad (13)$$

with the nonholonomic constraint:

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0 \quad (14)$$

This equation is not integrable, so the feasible trajectory of robot is limited.

B. Visual Guidance Architecture

The visual servoing architecture proposed in the paper is illustrated in (Figure 1). The system is composed from a camera mounted in a two wheel platform. Images are acquired, post processed, in order to describe the scene by point features and thus having point features error in relation with a desired configuration. These errors are input in a switched image based controller.

In order to design the image based controller the intrinsic parameters of the camera

$$\mathbf{K} = \begin{bmatrix} 1 & 0 & 0 \\ o_u & \alpha_y & 0 \\ o_v & 0 & \alpha_z \end{bmatrix} \quad (15)$$

are considered to be known or obtained after a calibration stage. If a point $\mathbf{P} = (X, Y, Z)^T$ is considered then, using perspective projection model [11], one can obtain:

$$\begin{aligned} u &= -\alpha_y \frac{Y}{X} + o_u \\ v &= -\alpha_z \frac{Z}{X} + o_v \end{aligned} \quad (16)$$

For any point feature $f = (u, v)^T$, the interactive matrix must be derived taking into consideration the structure of the reference camera velocity vector $\mathbf{v}_c^* = (v_c, \omega_c)$. The proposed method of design for the image based controller will used two interaction matrixes. First will be computed

for $\mathbf{v}_c^* = \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix}$, while the second one will be associated with $\mathbf{v}_c^* = \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix}$.
Using:

$$\dot{\mathbf{P}} = -v_c - \omega_c \times \mathbf{P}, \quad (17)$$

after some simple mathematical steps it results:

$${}^1\mathbf{L} = \begin{bmatrix} \frac{u - o_u}{X} & -\frac{\alpha_y}{X} & -\alpha_y - \frac{(u - o_u)^2}{\alpha_y} \\ \frac{v - o_v}{X} & 0 & -\alpha_y - \frac{(o_u - u)(o_v - v)}{\alpha_y} \end{bmatrix} \quad (18)$$

for $\mathbf{v} = \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix}$, and

$${}^2\mathbf{L} = \begin{bmatrix} \frac{u - o_u}{X} & -\alpha_y - \frac{(u - o_u)^2}{\alpha_y} \\ \frac{v - o_v}{X} & -\alpha_y - \frac{(o_u - u)(o_v - v)}{\alpha_y} \end{bmatrix} \quad (19)$$

for $\mathbf{v} = \begin{bmatrix} v_x \\ \omega_z \end{bmatrix}$.

Taking into consideration the nonholonomic constraints (14), the reference camera velocity $\mathbf{v}_c^* = [v_x, v_y, \omega_z]^T$ could not be fulfilled by the mobile platform. Thus a decoupling of the servoing tasks is needed. The proposed image based controller will first minimize the error from the image plane that generates the v_y velocities. To address this problem the following steps are used:

- 1) $v_R = 0$; $\omega_R = -\text{sgn}(v_y) \cdot \frac{\pi}{2} - \theta$;
- 2) $v_R = |v_y|$; $\omega_R = 0$;
- 3) $v_R = 0$; $\omega_R = -\text{sgn}(v_y) \cdot \frac{\pi}{2} + \theta$;

The main objective of the third step of the algorithm is to keep the Field of View orientation unchanged by the current orientation of the mobile robot.

The above controller is kept active as long as the control law from (9), for $\mathbf{L} = {}^1\mathbf{L}$, generates velocities on the y axis larger then a threshold ξ set by the user. If $|v_y| < \xi$ then the controller switches to $\mathbf{L} = {}^2\mathbf{L}$ and thus generates through (9) a camera velocity vector that has two components $\mathbf{v} = \begin{bmatrix} v_x \\ \omega_z \end{bmatrix}$. This reference camera velocity is directly transferred to the mobile robot: $v_R = v_x, \omega_R = \omega_z$

IV. EXPERIMENTAL RESULTS

The visual control algorithm described in Section III was implemented, tested and validated using a real-time servoing system composed from a Khepera III mobile platform, a monocular camera Genius Eye 312 connected via USB with a PC that has Matlab software (Figure 2). In the structure from (Figure 2) the parameters T (computational time of the Matlab algorithm), T_c (communication time between PC and Robot), T_R (command execution time) are the components of the sample period:

$$T_s = T + T_c + T_R \quad (20)$$

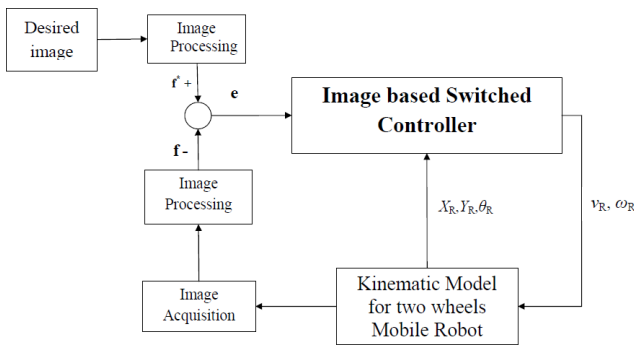


Fig. 1. Visual Guidance Architecture

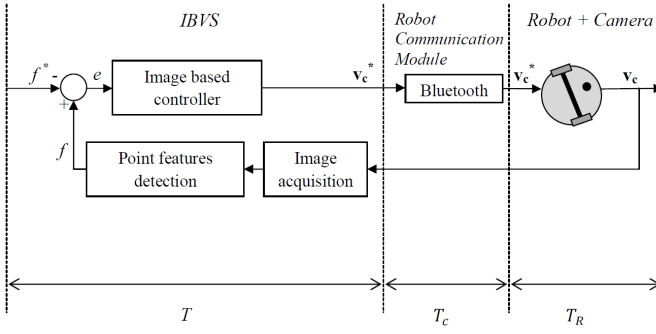


Fig. 2. Visual Servoing Structure

Khepera III robot has circular shape with a diameter of approximately 9 cm. It is differentially wheel-driven, and the user can directly specify the desired wheel speeds. The robot can be programmed using its internal memory, or a wireless (Bluetooth) communication with a computer, when the complexity of control algorithm increases. The capabilities of this type of robots allows implementation of motion planning algorithms without having to consider complicated robot dynamics. Because the developed visual feedback control strategy is too complex for robots processing and storage capabilities, the control of the robot is done via Bluetooth communication with the host computer. The robot will move on the table, and the mounted video camera will close the loop between the robot and the computer (Figure 3).

Before implementing the visual control algorithm information about the intrinsic parameters of the video camera are needed. Using the Camera Calibration Toolbox for Matlab, developed by Computational Vision Group at Caltech University [12], the following intrinsic parameters matrix was obtained:

$$\mathbf{K} = \begin{bmatrix} 1 & 0 & 0 \\ 183 & 468 & 0 \\ 103 & 0 & 468 \end{bmatrix} \quad (21)$$

One of the most time consuming problems when dealing with real-time implementation for visual servoing systems is the detection and tracking of the point features ensemble.

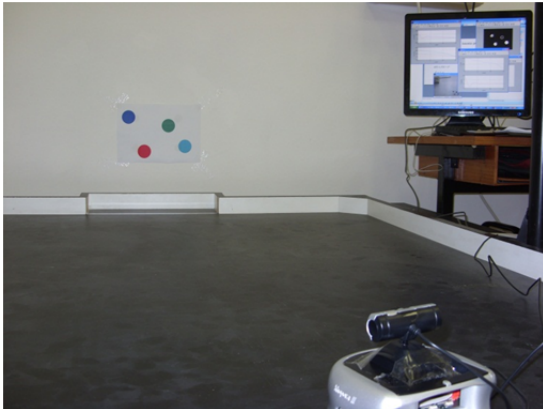


Fig. 3. Real time visual servoing system

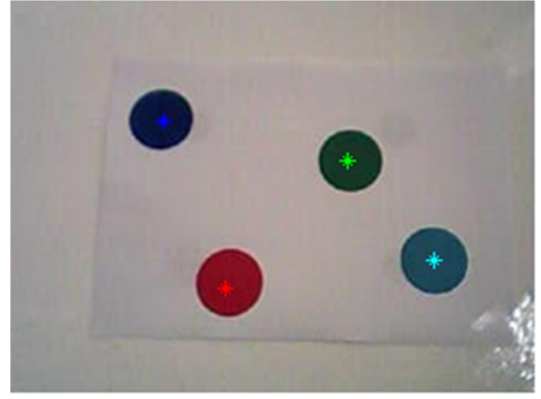


Fig. 4. Desired Image

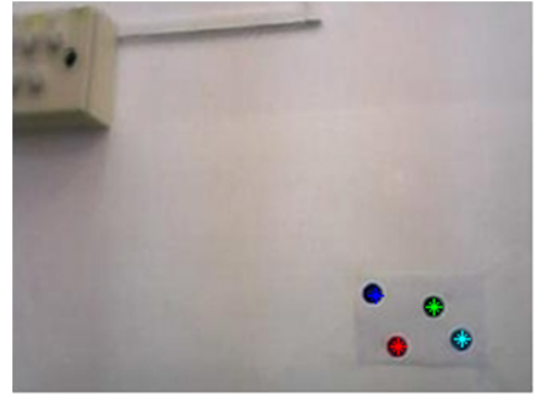


Fig. 5. Start Image

In order to have a low computational time the following solution was considered: the acquired images will be colored and process directly in the RGB space. For these four disks, each filled with a different color, are considered (Figure 4). After image acquisition, a post processing stage is conducted and color segmentation is obtained. A method based on the classical Euclidean distance was considered, while the mean of the RGB components of each disk was extracted using a statistical algorithm. After the segmentation stage, each separate region is described by its centroid and thus the tracking problem between consecutive frames is solved.

Multiple experiments were considered and in the followings the results of two distinct cases are revealed. The desired image is presented in (Figure 4) and the starting image is illustrated in (Figure 5). First the control parameters were set on $T = 1[s]$; $^1\lambda = 0.01$; $^2\lambda = 0.2$. In this case the component v_y , of the reference camera velocity, had an absolute value lower then $0.01[m/s]$. The behavior of the camera velocities are illustrated in (Figure 6), while the convergence of the system can be observed from the error behavior (Figure 7). The second set of results is the output of tuning the control parameters on $T = 0.5[s]$; $^1\lambda = 0.03$; $^2\lambda = 0.6$. Having the initial pose of the robot the visual control algorithm manages the fulfill the servoing task (Figure 8), (Figure 9).

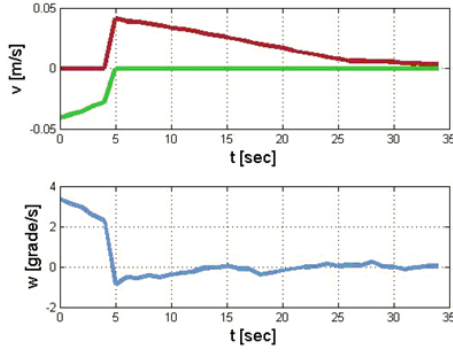


Fig. 6. Camera velocity experiment 1

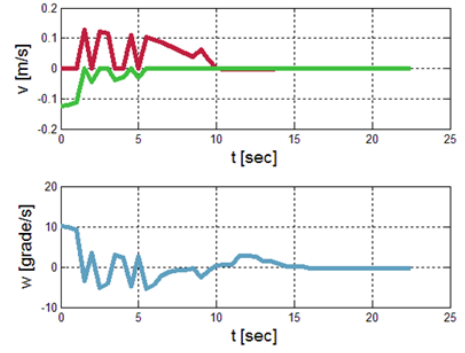


Fig. 8. Camera velocity experiment 2

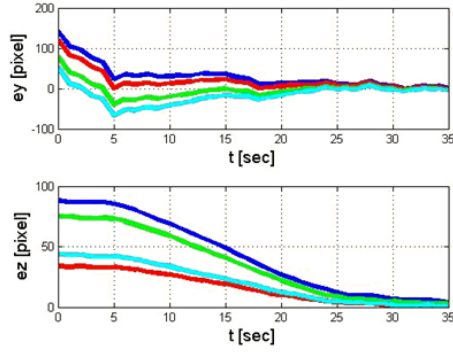


Fig. 7. Point features error experiment 1

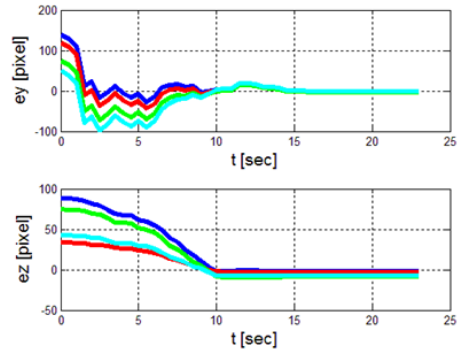


Fig. 9. Point features error experiment 2

V. CONCLUSIONS

In the present paper a real-time visual servoing strategy for a nonholonomic mobile robot was developed. An image based control architecture for two wheels mobile visual servoing system was presented. The proposed method takes into account the nonholonomic planar motion constraint of mobile robots and does not require the estimation and decomposition of the homography or fundamental matrix. A switched image based controller was designed, using point features, in order to solve the camera field-of-view (FOV) constraint. The proposed architecture was implemented using a real-time visual servoing system and multiple test were conducted. Experimental results revealed accuracy and convergence for all the considered tests.

Future work will imply the development of a point features predictor for mobile visual servoing systems in order to address applications with non-static objects.

VI. ACKNOWLEDGMENTS

This paper was supported by the project PERFORM-ERA "Postdoctoral Performance for Integration in the European Research Area" (ID-57649), financed by the European Social Fund and the Romanian Government.

REFERENCES

- [1] F. Chaumette and S. Hutchinson, "Visual servo control, part i: Basic approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.
- [2] S. Hutchinson, G. Hager, and P. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, 1996.
- [3] F. Chaumette and S. Hutchinson, "Visual servo control, part ii: Advanced approaches," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 109–118, 2007.
- [4] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics' Modelling, Planning and Control*. London: Springer, 2009.
- [5] A. D. Luca, G. Oriolo, L. Poone, and P. Giordano, "Experiments in visual feedback control of a wheeled mobile robot," in *Proc. of IEEE Conference on Robotics and Automation*, Washington, 2002, pp. 2073–2078.
- [6] R. Basri, E. Rivlin, and I. Shimshoni, "Image-based robot navigation under the perspective model," in *Proc. of IEEE Conference on Robotics and Automation*, vol. 4, 1999, pp. 2578–2583.
- [7] G. Lopez-Nicolas, C. Sagues, J. Guerrero, D. Kragic, and P. Jensfelt, "Nonholonomic epipolar visual servoing," in *Proc. of IEEE Conference on Robotics and Automation*, 2006, pp. 2378–2384.
- [8] G. Mariottini and D. Prattichizzo, "Image-based visual servoing with central catadioptric cameras," *The International Journal of Robotics Research*, vol. 27, no. 41, pp. 41–56, 2008.
- [9] X. Zhang, Y. Fong, and X. Liu, "Visual servoing of nonholonomic mobile robots based on a new motion estimation technique," in *Joint 48th IEEE Conf. on Decision and Control and 28th Chinese Control conference*, 2009, pp. 8428–8433.
- [10] B. Siciliano and O. Khatib, Eds., *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer, 2008.
- [11] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry, *An Invitation to 3-D Vision*. Springer, 2003.
- [12] J. Bouquet, "Camera calibration toolbox for matlab," Pasadena, USA, jul 2010, dept. Elect. Eng., California Institute of Technology. [Online]. Available: <http://www.vision.caltech.edu/bouquetj/index.html>

Real-time Simulation of a Vehicle Drivetrain Controlled through CAN using a Robust MPC Strategy

Constantin Florin Caruntu*, Doina Onu** and Corneliu Lazar*

Abstract—This paper proposes a real-time implementation of a networked one step ahead robust predictive controller designed to damp driveline oscillations, which is crucial in improving driveability and passenger comfort, while compensating the time-varying delays that appear due to sending the control commands and the measurements from the sensors through Controller Area Network (CAN). Firstly, the designed real-time structure integrated with CAN test-bench is described and then the model of the drivetrain is derived. Secondly, considering that the CAN-induced time-varying delays are bounded, a method to model their effects as disturbances is proposed. Then, a robust one step ahead predictive controller based on flexible control Lyapunov functions is designed, which explicitly takes into account the bounds of the disturbances caused by time-varying delays and guarantees also the input-to-state stability of the system in a non-conservative way. The proposed control scheme is tested using the designed test-bench and the experiments based on realistic scenarios illustrate its performances.

I. INTRODUCTION

Recently, the need for increased driveability and passenger comfort lead to the design of proper models for vehicle drivetrains and to the development of different control strategies to minimize the effects of drivetrain oscillations. When a vehicle is subjected to acceleration, the elasticity of the various components in the driveline may cause torsional vibrations or disturbances. The oscillations give rise, apart from material stress, to noticeable reduced driveability and comfort.

Various research has been conducted on understanding and damping the driveline oscillations of conventional vehicles and numerous control strategies were reported in the literature: robust pole placement [1], $\mathcal{H}_\infty$ optimization [2], linear quadratic gaussian control design with loop transfer recovery (LQG/LTR) [3] and even model predictive control (MPC) [4].

All of the above control solutions assume that the sensors, controllers and actuators are directly connected, which is not realistic. Rather, in modern vehicles, the control signals from the controllers and the measurements from the sensors are exchanged using a communication network, e.g., Controller Area Network (CAN) or Flexray, among control system components. This brings up a new challenge on how to deal

with the effects of the network-induced delays and packet losses in the control loop. The delays may be unknown and time-varying and may degrade the performances of control systems designed without considering them and can even destabilize the closed-loop system.

The testing of control algorithms in their real environment represents an important phase in the development of control units for drivetrains. The major feature of a real-time simulation is that the simulation can be carried out as quickly as the real system would actually run, thereby allowing to combine the simulation and the real CAN network. The absence of the real device in the simulation particularly simplifies the development, since it allows systematically and effectively validation of the closed-loop control system, even if the controller is not hardware implemented.

The use of real-time simulation can replace significantly the construction of expensive prototypes to test drivetrain systems. Due to the utilization of the model in the closed loop, the simulation is cheaper, easier and less time consuming than experiments on the real device. Moreover, the risks to damage the real device are eliminated.

The computing power required by real-time simulation highly depends on the characteristics of the simulated model: if it contains very demanding calculations you have to provide a lot of computing power because the timing cannot be satisfied otherwise. A simulation performed in Simulink takes as much computer time as needed to compute the control command and then to calculate the behavior of the system. If the model is complex, much more time is needed to carry out the necessary calculations. Because a strict time requirement for Simulink simulation does not need to be fulfilled, the complexity of the model does not need to be reduced.

As such, the problem considered in this paper is the development of a real-time simulation test-bench designed to validate a one step ahead networked predictive control strategy proposed to minimize the oscillations of a vehicle drivetrain, while compensating the time-varying delays introduced by CAN. The proposed solution consists of four steps: firstly, a piecewise affine (PWA) three inertias drivetrain model, which takes into consideration the clutch flexibility together with the driveshaft flexibility is derived. Secondly, the error caused by the time-varying CAN-induced delays is considered as a disturbance and a novel method of finding the bounds of the disturbance is proposed. Thirdly, a robust one step ahead MPC scheme is designed using

*Department of Automatic Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Romania
{caruntuc, clazar}@tuiasi.ro

**Continental Automotive Romania
Doina.Onu@continental-corporation.com

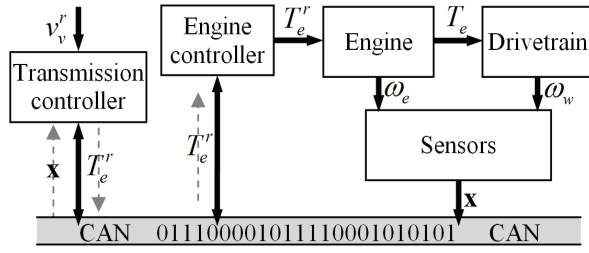


Fig. 1. Drivetrain control architecture.

the concept of flexible control Lyapunov functions (CLF) [5] that explicitly accounts for rejection of disturbances introduced by the time-varying delay in the communication network and guarantees also the input-to-state stability of the system in a non-conservative way. Fourthly, the one step ahead networked predictive controller and the plant model are implemented on a real-time simulation test-bench and the designed experiments validate the proposed approach and indicate that the proposed scheme can outperform other types of controllers, e.g., PI.

Notation and basic definitions. $\mathbb{R}$, $\mathbb{R}_+$, $\mathbb{Z}$ and $\mathbb{Z}_+$ are the real, non-negative real, integer and non-negative integer numbers. We use the notation $\mathbb{Z}_{\geq c_1}$ and $\mathbb{Z}_{(c_1, c_2]}$ to denote the sets $\{k \in \mathbb{Z}_+ \mid k \geq c_1\}$ and $\{k \in \mathbb{Z}_+ \mid c_1 < k \leq c_2\}$, respectively, for some $c_1, c_2 \in \mathbb{Z}_+$. Let $\|x\|_\infty := \max_{i \in \mathbb{Z}_{[1, n]}} |x_i|$, where $|\cdot|$ denotes the absolute value. A function $\varphi : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ belongs to class $\mathcal{K}$ if it is continuous, strictly increasing and $\varphi(0) = 0$. A function $\varphi : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is said to belong to class $\mathcal{K}_\infty$ if it is of class $\mathcal{K}$ and $\lim_{s \rightarrow \infty} \varphi(s) = \infty$.

II. DRIVETRAIN CONTROL ARCHITECTURE

The structure of a vehicle drivetrain consists, in general, of the following parts: engine, clutch, transmission, final drive, driveshafts and wheels.

A. Network-controlled architecture

As it can be seen in Fig. 1 the components of the closed-loop vehicle drivetrain control system, e.g., sensors, controller, actuator, communicate through a communication network, i.e., the engine torque (control signal), and also the wheel speed are sent through CAN. Many of the messages transmitted on CAN, have real-time constraints associated with them.

The complete network-controlled architecture considered in this paper, which is graphically depicted in Fig. 1, consists of the following operations:

- the sensors measure the outputs of the system and send the samples to the controller through CAN;
- the transmission controller receives the measurements from the sensors and the desired velocity reference $v_v^r = r_w \omega_w^r$ and computes the required torque, while handling the physical constraints and the delays, where r_w is the wheel radius and ω_w^r is the desired wheel speed;
- the control signal, i.e., the torque computed by the controller, is sent to the engine controller through CAN;

- the engine controller actuates the spark timing and airflow as requested for driveline control.

In Fig. 1 the dashed lines represent the direction of the messages sent to and from the controllers.

B. Real-time simulation setup

The test-bench setup for real-time simulation of the closed-loop drivetrain control system, which is designed to validate the control algorithm, consists of: dSPACE MicroAutoBox, Host PC, CAN network, power supply, a Simulink model of the drivetrain and controller and a Graphical User Interface (GUI) as it can be seen in Fig. 2.

MicroAutoBox consists of the base board DS1401 and it performs the drivetrain plant-model and controller calculations. It has an Ethernet interface for direct connection to the host PC, for tasks such as loading models and reading or adjusting parameters in GUI. The HostPC also runs the simulator's console software (GUI), designed in ControlDesk, to communicate with the real-time Simulator (MicroAutoBox). The application allows to download and control the real time process such as download, start, stop, variable observation and data collection.

Real-Time Interface (RTI) is the link between dSPACE MicroAutoBox hardware and the MATLAB/Simulink models. Real-Time Workshop (RTW) generates the model code which is compiled and downloaded in the real-time hardware. The RTI CAN MultiMessage Blockset is an extension for Real-Time Interface and it is used for configuring the CAN network. CAN configurations are read from a database container (DBC) file.

Note that the Simulink model of the drivetrain and controller exchange data via a CAN bus. Also, the Host PC has access to the CAN bus through the developed ControlDesk application.

C. Drivetrain model

A PWA three inertias model, which takes into consideration the clutch flexibility together with the driveshaft flexibility, was derived from the laws of motion [6], in which the first inertia corresponds to the engine, the second one includes the inertia of the gearbox and the inertia of the final drive, and the last inertia corresponds to the wheel and vehicle mass, as it is represented in Fig. 3. The clutch torsional flexibility is a result of an arrangement of smaller springs in series with springs with much larger stiffness. The reason for this arrangement is vibration insulation.

In this paper four working modes are introduced, i.e., open, closing, closed and locked, which yields a more accurate model of the clutch. In the *open* mode, there is no mechanical connection between the engine and the rest of the driveline, so no torque is transmitted towards the wheels. In the *closing* mode, the smaller springs in the clutch are compressed, which means that the engine torque is gradually transmitted to the driveline. In the *closed* mode, the stiffer springs in the clutch are compressed and the gradual transmission of torque to the driveline continues. The *locked* mode corresponds to the phase when the springs in

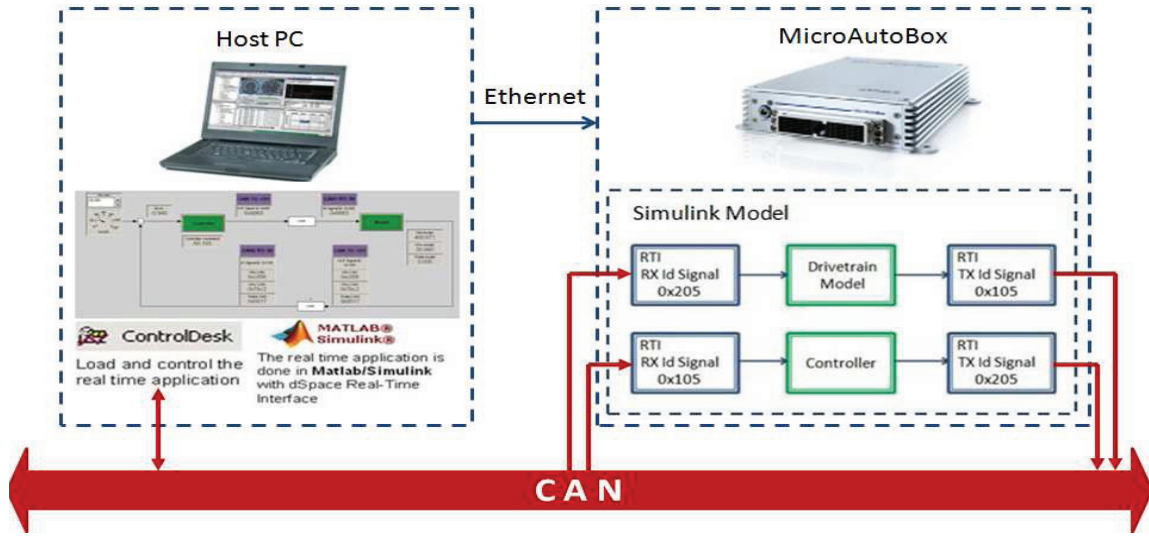


Fig. 2. Real-time test-bench.

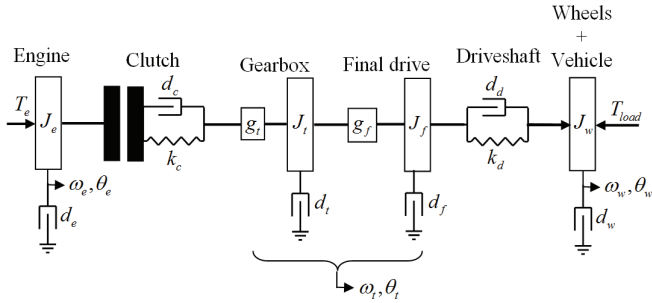


Fig. 3. Three inertias drivetrain model including the clutch.

the clutch cannot be compressed any further, i.e., the clutch hits a mechanical stop, and the maximum amount of torque is transmitted from the engine to the wheels through the driveline.

Consider as state variables: the torsional angle between engine and transmission, the torsional angle between transmission and wheel, the angular speed of the engine, the angular speed of the transmission and the angular speed of the wheel, and as control input of the system, the engine torque, i.e.,

$$\begin{aligned} x_1 &= \theta_e - \theta_t, \quad x_2 = \frac{\theta_t}{g_f} - \theta_w, \\ x_3 &= \omega_e, \quad x_4 = \omega_t, \quad x_5 = \omega_w, \quad u = T_e, \end{aligned} \quad (1)$$

where θ_e , θ_t and θ_w are angles of the engine, transmission and wheel, respectively, ω_e , ω_t and ω_w are the angular speeds of the engine, transmission and wheel, respectively, and T_e is the torque produced by the engine.

The following PWA state-space model is obtained:

$$\dot{x}(t) = A_{ci}x(t) + b_c u(t) + f_c \quad \text{if } x(t) \in \Omega_i, \quad (2)$$

where $x := (x_1, \dots, x_5)^T \in \mathbb{R}^5$ and $i \in \mathcal{I} := \mathbb{Z}_{[1,4]}$. Here i denotes the active mode at time $t \in \mathbb{R}_+$, $A_{ci} \in \mathbb{R}^{5 \times 5}$, $b_c \in \mathbb{R}^{5 \times 1}$ are the system matrices and $f_c \in \mathbb{R}^{5 \times 1}$ is the

affine term. Notice that although there are 3 angles, only two states are introduced as only the angle difference is relevant. The collection of sets $\{\Omega_i \mid i \in \mathcal{I}\}$ defines a partition of the state-space $\mathbb{X} \subseteq \mathbb{R}^5$ as follows:

$$\begin{cases} \Omega_1 := \{x \in \mathbb{R}^5 \mid x_3 \leq \omega_e^{\text{closing}}\}, & \text{- open} \\ \Omega_2 := \{x \in \mathbb{R}^5 \mid x_3 > \omega_e^{\text{closing}} \text{ \& } |x_1| \leq \theta_1\}, & \text{- closing} \\ \Omega_3 := \{x \in \mathbb{R}^5 \mid x_3 > \omega_e^{\text{closing}} \text{ \& } \theta_1 < |x_1| \leq \theta_2\}, & \text{- closed} \\ \Omega_4 := \{x \in \mathbb{R}^5 \mid x_3 > \omega_e^{\text{closing}} \text{ \& } \theta_2 < |x_1|\}, & \text{- locked} \end{cases} \quad (3)$$

where $\omega_e^{\text{closing}}$ is the engine closing speed and θ_1 , θ_2 are threshold values for the torsional angle between the engine and the transmission, which determine the working mode of the clutch. Note that when a transition from the open mode to the closing mode occurs, the following reset condition must be imposed:

$$\begin{aligned} \forall t_1 \in \mathbb{R}_+, \forall t_2 \in \mathbb{R}_{>t_1}, \text{ if } x(\tau) \in \Omega_1, \forall \tau \in \mathbb{R}_{[t_1, t_2)} \\ \text{and } x(t_2) \in \Omega_2, \text{ set } x_1(t_2) := 0. \end{aligned} \quad (4)$$

As the engine angle θ_e tends to infinity in the open mode, so the state x_1 tends to infinity, synchronization of the engine angle and the transmission angle must be attained at the moment the clutch switches from the open mode to the closing mode. Notice that the reset condition does not apply to initial conditions, as the model cannot be initialized in the closing mode. The matrices A_{ci} , b_c and f_c can be obtained by deriving the whole dynamical model of the drivetrain using the generalized Newton's second law of motion as it is done in [7].

The engine torque (i.e., the control input) is restricted by lower and upper bounds and by a torque rate constraint as follows:

$$\begin{aligned} 0 &\leq u(t) \leq T_e^{\text{max}}, \quad \forall t \in \mathbb{R}_+, \\ T_e^{\text{m}} &\leq \dot{u}(t) \leq T_e^{\text{M}}, \quad \forall t \in \mathbb{R}_+, \end{aligned} \quad (5)$$

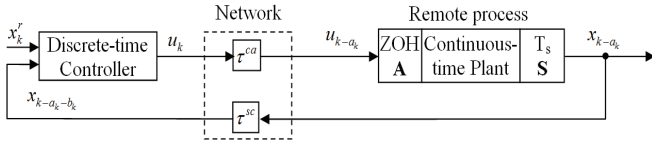


Fig. 4. Control system with network-induced time delay.

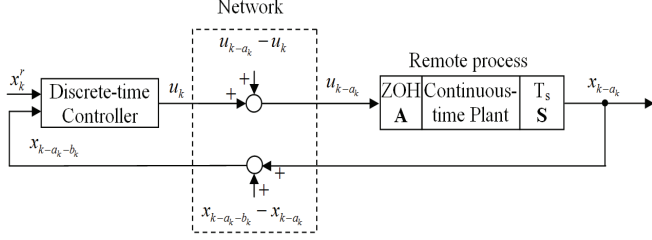


Fig. 5. Control system with network-induced time delay as disturbance.

where $T_e^{\max}$ is the maximum torque that can be generated by the internal combustion engine and T_e^m , T_e^M are torque rate bounds. Furthermore, the engine and wheel speeds are bounded, i.e.,

$$\omega_e^{\min} \leq x_3(t) \leq \omega_e^{\max}, \quad \forall t \in \mathbb{R}_+, \quad (6)$$

$$\omega_w^{\min} \leq x_5(t) \leq \omega_w^{\max}, \quad \forall t \in \mathbb{R}_+, \quad (7)$$

where $\omega_e^{\min}$ and $\omega_e^{\max}$ are the idle speed and the engine limit speed, respectively, and $\omega_w^{\min}$ and $\omega_w^{\max}$ are the minimum and the maximum speed of the wheels.

III. NETWORK DELAY MODELED AS DISTURBANCE

Consider the standard NCS illustrated in Fig. 4, which is composed of five parts: a communication network, where it is assumed that delay occurs completely randomly, a physical plant, one sensor node (S), one controller node and one actuator node (A). The delays introduced by the communication network, which can be smaller and larger than a sampling period, are represented as τ^{ca} for the delay in the forward channel and as τ^{sc} for the delay in the feedback channel.

The physical plant is given by a linear state-space model

$$\dot{x}(t) = A_c x(t) + B_c u(t - \tau^{ca}) \quad (8)$$

where $x(t)$ is the system state, $u(t)$ is the control signal and A_c and B_c are the continuous-time system matrices.

It is easy to observe that Fig. 4 is equivalent to Fig. 5, in which the difference between the actual control signal and the delayed one is regarded as a hypothetical disturbance [8]. Furthermore, the difference between the actual output and the delayed one is regarded as another hypothetical disturbance. Note that, this method is different than the one proposed in [9], where some assumptions were made in order to consider that the time-varying delays that appear in the feedback channel can be considered as time delays in the forward channel.

Notation: Let a_k and b_k denote the delay in the forward channel and in the feedback channel, respectively, at time

instant k , expressed as a number of sampling periods: $a_k = \lceil \tau_k^{ca} / T_s \rceil$, $b_k = \lceil \tau_k^{sc} / T_s \rceil$, where T_s is the sampling period of the system. Moreover, let $\bar{a}$ and $\bar{b}$ denote the maximum delays in the forward channel and in the feedback channel, respectively, expressed as a number of sampling periods: $\bar{a} = \lceil \tau^{ca \max} / T_s \rceil$, $\bar{b} = \lceil \tau^{sc \max} / T_s \rceil$.

A. Forward channel delays

The discrete-time forward channel disturbance representation is given by

$$u_k^d = u_{k-a_k} - u_k \quad (9)$$

and the goal is to find a bounded set $\mathbb{W}_u$, in which to include all possible disturbances that can appear due to the time-varying delays introduced in the forward channel, knowing that the input of the remote process is bounded.

Now consider that the control signal is bounded by lower and upper bounds

$$u^{\min} \leq u_k \leq u^{\max}, \quad (10)$$

where $u^{\min}$ and $u^{\max}$ are the minimum and the maximum control signal values that can be given as input to the remote process. Then, the disturbance can be bounded as [9]

$$u^{\min} - u^{\max} \leq u_k^d \leq u^{\max} - u^{\min}. \quad (11)$$

Furthermore, if the discrete control signal is also restricted by an incremental bound

$$-u^\Delta \leq \Delta u_k \leq u^\Delta, \quad (12)$$

where $\Delta u_k := u_k - u_{k-1}$, for all $k \in \mathbb{Z}_{\geq 1}$, with u_0 some pre-determined value and u^Δ is the maximum increase/decrease of the control signal at each sampling time instant $k \in \mathbb{Z}_{\geq 1}$, the disturbance can be rebounded as [9]

$$-\bar{a}u^\Delta \leq u_k^d \leq \bar{a}u^\Delta. \quad (13)$$

B. Feedback channel delays

The hypothetical disturbance exerted on the physical plant delays the feedback signal, the discrete-time feedback channel disturbance representation being given by

$$x_k^d = x_{k-a_k-b_k} - x_{k-a_k}. \quad (14)$$

The goal is to find a bounded set $\mathbb{W}_x$, in which to include all possible disturbances that can appear due to the time-varying delays introduced in the feedback channel, knowing that the input of the remote process is bounded.

Consider the physical plant (8) with input disturbance

$$\dot{x}(t) = A_c x(t) + B_c (u(t) + u^d(t)) \quad (15)$$

Discretizing (15) yields

$$\begin{aligned} x_{k+1} &= A_d x_k + B_d (u_k + u_k^d) = \\ &= A_d^2 x_{k-1} + A_d B_d (u_{k-1} + u_{k-1}^d) + B_d (u_k + u_k^d) = \\ &= A_d^j x_{k-j+1} + A_d^{j-1} B_d (u_{k-j+1} + u_{k-j+1}^d) + \dots \\ &\quad \dots + A_d B_d (u_{k-1} + u_{k-1}^d) + B_d (u_k + u_k^d) = \\ &= A_d^j x_{k-j+1} + \sum_{i=0}^{j-1} A_d^i B_d (u_{k-i} + u_{k-i}^d), \end{aligned} \quad (16)$$

where $A_d = e^{A_c T_s}$, $B_d = \int_0^{T_s} e^{A_c(T_s-\theta)} d\theta B_c$.

Then, (14) becomes

$$\begin{aligned} x_k^d &= x_{k-a_k-b_k} - A_d^{b_k} x_{k-a_k-b_k} - \\ &\quad - \sum_{i=0}^{b_k-1} A_d^i B_d (u_{k-a_k-i-1} + u_{k-a_k-i-1}^d) = \\ &= (I_n - A_d^{b_k}) x_{k-a_k-b_k} - \\ &\quad - \sum_{i=0}^{b_k-1} A_d^i B_d (u_{k-a_k-i-1} + u_{k-a_k-i-1}^d) \end{aligned} \quad (17)$$

$x_{k-a_k-b_k}$ is known at time instant k , all u_{k-i} , $i \in \mathbb{Z}_{\geq 1}$ are known, all u_{k-i}^d , $i \in \mathbb{Z}_{\geq 1}$ are bounded, a_k and b_k are bounded by $\bar{a}$ and $\bar{b}$, respectively, so x_k^d can be dynamically bounded at each sampling instant $k \in \mathbb{Z}_+$.

Now, (16) becomes

$$x_{k+1} = A_d x_k + B_d u_k + B_d u_k^d + x_k^d. \quad (18)$$

Even though the time-varying delay gives a time-varying disturbance, the sets $\mathbb{W}_u$, which is defined by $B_d u_k^d$, and $\mathbb{W}_x$, which is defined by x_k^d , remain fixed, so this modeling technique is suitable for the use of the results presented in [10], in which the disturbances are explicitly taken into account during the design phase of the predictive controller, which will be accomplished in the next section.

IV. ONE STEP AHEAD ROBUST PREDICTIVE CONTROLLER DESIGN

The main idea of the proposed RLMPC is to formulate appropriate constraints in the predictive controller optimization problem based on an existing Lyapunov-based controller, in such a way that the MPC controller inherits the robustness and stability properties of the Lyapunov-based controller, in a non-conservative way. Standard MPC strategies, which typically require a sufficiently long prediction horizon to assure stability and performance, would lead to solutions that are too complex and, at the same time, too conservative to be implemented in real-time control applications, for physical plants with fast dynamics, which require short sampling periods.

Recently, an extension and relaxation of the conventional Lyapunov condition was proposed in [5], yielding so-called flexible Lyapunov functions which offer a stability guarantee (under a recursive feasibility assumption) even for a short prediction horizon. In this paper, a robust one step ahead MPC scheme that employs a flexible Lyapunov function [5] to attain stability and performance, in which the disturbances are explicitly taken into account during the design phase by using input-to-state stability (ISS) concepts (see, e.g. [11]), is proposed. Furthermore, it is shown that for a particular choice of the Lyapunov function candidate the proposed design leads to a low-complexity linear program (LP) that can be solved efficiently within the required time range, while strictly enforcing the constraints and considering the delays induced by the communication network, via the usage of the model derived in Section III.

Now, consider the perturbed discrete-time constrained nonlinear system of the form

$$\begin{aligned} x_{k+1} &= \bar{\phi}(x_k, u_k, w_k) := \phi(x_k, u_k) + w_k \\ &:= f(x_k) + g(x_k)u_k + w_k, \quad k \in \mathbb{Z}_+, \end{aligned} \quad (19)$$

where $x_k \in \mathbb{X} \subseteq \mathbb{R}^n$ is the state, $u_k \in \mathbb{U} \subseteq \mathbb{R}^m$ is the control input and $w_k \in \mathbb{W} \subseteq \mathbb{R}^n$ is an unknown disturbance at the discrete-time instant k . $\bar{\phi} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}^n$, $\phi : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$, $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ are arbitrary nonlinear, possibly discontinuous, functions with $\bar{\phi}(0,0,0) = 0$, $\phi(0,0) = 0$, $f(0) = 0$ and $g(0) = 0$. Naturally, it is assumed that the set of feasible states $\mathbb{X}$, the set of feasible inputs $\mathbb{U}$ and the disturbance set $\mathbb{W}$ are bounded polyhedra with non-empty interiors containing the origin. Next, let $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ and let $\sigma \in \mathcal{K}$.

Definition IV.1 A function $V : \mathbb{R}^n \rightarrow \mathbb{R}_+$ that satisfies

$$\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|), \quad \forall x \in \mathbb{R}^n \quad (20)$$

and for which there exists a possibly set-valued control law $\pi : \mathbb{R}^n \rightrightarrows \mathbb{U}$ such that

$$\begin{aligned} V(\bar{\phi}(x, u, w)) - V(x) &\leq -\alpha_3(\|x\|) + \sigma(\|w\|), \\ \forall x \in \mathbb{X}, \forall u \in \pi(x), \forall w \in \mathbb{W}, \end{aligned} \quad (21)$$

is called an *input-to-state stability control Lyapunov function (ISS-CLF)* in $\mathbb{X}$ for system (19) and disturbances in $\mathbb{W}$.

ISS theory (see [11]) can be used to derive an input-to-state stabilizing predictive control scheme with improved disturbance rejection, as done in [10], where this property is referred to as *optimized ISS*.

As such, let $\mathbb{W}$ be a convex hull of the vertices w^e , $e = 1, \dots, E$, and let λ_k^e , $k \in \mathbb{Z}_+$, be optimization variables associated with each vertex w^e . Let $J(\lambda^1, \dots, \lambda^E, \lambda) : \mathbb{R}_+^E \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$ be a strictly convex, radially unbounded function (i.e. $J(\cdot)$ tends to infinity when its arguments tend to infinity) and let $J(\lambda^1, \dots, \lambda^E, \lambda) \rightarrow 0 \Rightarrow \lambda^e \rightarrow 0$ for all $e = 1, \dots, E$ and $\lambda \rightarrow 0$, and $J(0, \dots, 0, 0) = 0$.

Choose off-line a CLF $V(\cdot)$ for system (19) without disturbances and let $\alpha_3 \in \mathcal{K}_\infty$ and $x \in \mathbb{X}$ be given. At each control sampling instant $k \in \mathbb{Z}_+$ the one step ahead ISS MPC controller solves the following problem.

Problem IV.2 At time $k \in \mathbb{Z}_+$ measure the state x_k and minimize the cost $J(\lambda_k^1, \dots, \lambda_k^E, \lambda_k)$ over $u_k, \lambda_k^1, \dots, \lambda_k^E$ and λ_k , subject to the constraints

$$u_k \in \mathbb{U}, \quad \phi(x_k, u_k) \in \mathbb{X}, \quad \lambda_k^e \geq 0, \quad \lambda_k \geq 0, \quad (22a)$$

$$V(\bar{\phi}(x_k, u_k, 0)) - V(x_k) + \alpha_3(\|x_k\|) \leq \lambda_k, \quad (22b)$$

$$V(\bar{\phi}(x_k, u_k, w^e)) - V(x_k) + \alpha_3(\|x_k\|) - \lambda_k^e \leq 0, \quad (22c)$$

for all $e = 1, \dots, E$. $\square$

Let $\pi(x_k) := \{u_k \in \mathbb{R}^m \mid \exists \lambda_k, \lambda_k^e, e \in \mathbb{Z}_{[1,E]} \text{ s.t. (22) holds}\}$ and let $\bar{\phi}_{cl}(x_k, \pi(x_k), w_k) := \{\bar{\phi}(x_k, u_k, w_k) \mid u_k \in \pi(x_k)\}$ denote the difference inclusion corresponding to system (19) in closed-loop with

the set of feasible solutions obtained by solving Problem IV.2 at each sampling instant $k \in \mathbb{Z}_+$.

Next, the main robust stability result in terms of ISS is stated. This result is an adaptation of the main result in [10], to fit the relaxation (22b) of Problem IV.2, i.e., $\lambda_k = 0$ for all $k \in \mathbb{Z}_+$ corresponds to the problem considered in [10].

Theorem IV.3 *Let $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$, a continuous and convex CLF $V(\cdot)$ and a cost $J(\cdot)$ be given. Suppose that Problem IV.2 is feasible for all states x in $\mathbb{X}$ and assume that $\lim_{k \rightarrow \infty} \lambda_k^* = 0$. Then, the trajectories generated by the difference inclusion*

$$x_{k+1} \in \bar{\phi}_{cl}(x_k, \pi(x_k), w_k), \quad k \in \mathbb{Z}_+, \quad (23)$$

with initial state $x_0 \in \mathbb{X}$ converge in finite time to a robustly positively invariant subset of $\mathbb{X}$, in which the difference inclusion is ISS for disturbances in $\mathbb{W}$.

The proof of Theorem IV.3 follows from standard arguments employed in proving input-to-state stability and Lyapunov stability and is therefore omitted here. The interested reader is referred to [10] and [5] for more details. Advantageous properties of the proposed robust controller are that ISS is guaranteed for any (feasible) solution of the optimization problem, state and input constraints can be explicitly accounted for, and feedback to disturbances is provided actively, on-line. The key of the stability proof is the limiting condition $\lim_{k \rightarrow \infty} \lambda_k^* = 0$. In what follows a non-conservative solution for guaranteeing this condition is provided, such that λ_k can be small or even 0 for some time after which it is allowed to increase again, as long as this does not violate the upper bound.

Lemma IV.4 *Let $\Delta \in \mathbb{R}_+$ be a fixed constant to be chosen a priori and let $\rho \in \mathbb{R}_{[0,1]}$ and $M \in \mathbb{Z}_{>0}$. If*

$$0 \leq \lambda_k \leq \rho^{\frac{1}{M}} (\lambda_{k-1}^* + \rho^{\frac{k-1}{M}} \Delta), \quad \forall k \in \mathbb{Z}_{\geq 1}, \quad (24)$$

then $\lim_{k \rightarrow \infty} \lambda_k = 0$.

Lemma IV.4 is proven in [7] and is omitted here for brevity. By augmenting Problem IV.2 with constraint (24) the property $\lim_{k \rightarrow \infty} \lambda_k^* = 0$ is thus guaranteed, which is sufficient for asymptotic stability.

The fact that the developed one step ahead robust MPC scheme for the constrained system (19) can be implemented by solving a single LP during each control cycle using an infinity-norm based candidate CLF is shown in [12] and is omitted here.

V. EXPERIMENTAL RESULTS

This section presents the validations of the proposed networked one step ahead robust predictive control strategy investigated on the vehicle drivetrain model using the designed real-time simulation test-bench.

The continuous-time PWA model (2)-(4) was implemented in Matlab/Simulink and the proposed one step ahead predictive controller proposed in this paper was applied to damp

driveline oscillations. The sampling period of the system was chosen as $T_s = 5\text{ms}$. The values of the parameters used in simulations, which relate to a medium size passenger car, can be found in [7].

The upper bound of the delays that are induced by CAN was calculated using the methodology described in [13], resulting that $\tau^{max} = 3.4T_s = 0.017\text{s} < 4T_s$, which is used to apply the methodology described in Section III to model the effects of the variable time delays as disturbances. Then, the bounds of the disturbances are explicitly taken into account by the robust one step ahead predictive control strategy described in Section IV. The delays are time-varying and uniformly distributed in the interval $[0, \tau^{max}]$.

The control objective is to reach a desired speed reference in a short time, but, at the same time, to increase the passenger comfort by reducing the oscillations that appear in the driveline. The axle wrap is calculated as the difference between the engine speed (divided by the total transmission ratio) and the wheel speed, and it is used as a measure of the driveline oscillations.

The technique presented in [14] was used for the off-line computation of the infinity norm based local CLF $V(x) = \|Px\|_\infty$, where

$$P = \begin{pmatrix} 38.5 & -54.85 & -1.44 & 1 & -4.89 \\ -26.9 & 18.13 & 0.91 & -4.41 & 5.18 \\ 22.31 & 287.69 & 0.72 & -0.53 & -47.14 \\ -62.38 & -17.33 & 9.06 & -9.02 & 26.48 \\ 25.92 & 263.26 & -0.11 & 0.37 & 55.71 \end{pmatrix}. \quad (25)$$

For the robust one step ahead MPC scheme developed in this paper, recursive feasibility implies asymptotic stability. However, recursive feasibility is not a priori guaranteed and hinges mainly on the constraint (24) on the future evolution of λ_k^* . For all simulation scenarios case studies, the values $\Omega = 15000$ and $M = 5$ proved to be large enough to guarantee recursive feasibility for the desired operating scenarios.

The worst case time needed for computation of the control input for the proposed predictive controller was less than 2ms, which meets the timing constraints.

The results of a tip-in, tip-out maneuver simulation, in which the reference vehicle velocity goes from 0 km/h to 30 km/h to 10 km/h and back to 30 km/h, are presented in Fig. 6. The vehicle velocity reference and the response of the system are illustrated in Fig. 6, top right, where it can be seen that the system reaches the reference in a short time, having no overshoot when it approaches the reference velocity. The amplitude of the axle wrap is represented in Fig. 6, top right. In Fig. 6, bottom left, the engine torque (control command) is illustrated and it can be seen that the lower and upper constraints are satisfied. The clutch mode is represented in Fig. 6, bottom right.

VI. CONCLUSIONS

This paper considered the design of a real-time simulation test-bench in order to validate and to test the performances of

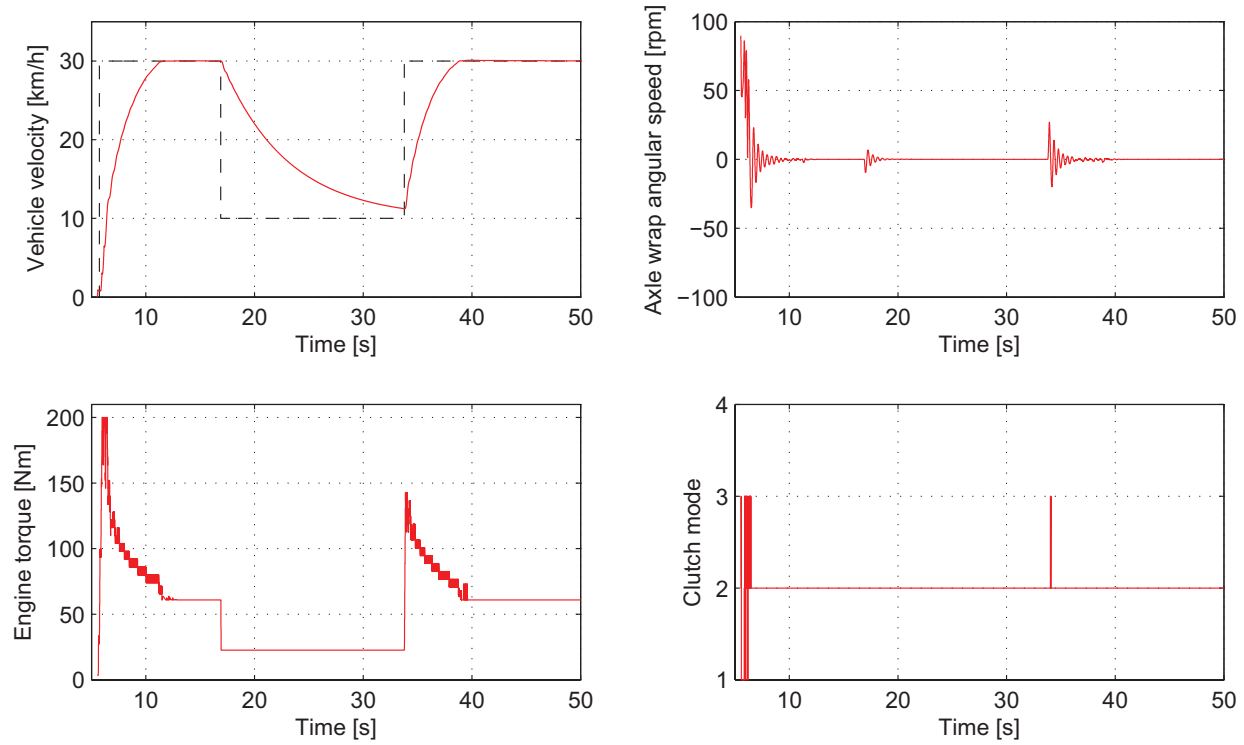


Fig. 6. Tip-in, tip-out maneuver simulation.

the robust control algorithm developed with the aim of damping driveline oscillations, while decreasing the influence of the communication time-varying delays on the closed-loop control performances over the CAN network. The designed robust one step ahead MPC strategy can handle the performance/physical constraints and can explicitly take into account the disturbances caused by the time-varying delay. Also, a flexible control Lyapunov function was employed to obtain a non-conservative ISS stability guarantee for the developed one step ahead MPC scheme. The proposed control strategy was tested on a real-time simulation test-bench including CAN communications and the results obtained illustrate that the proposed controller has good performances and it meets the required timing constraints.

ACKNOWLEDGMENT

The work was supported by National Center for Programs Management from Romania under the research grant SICONA - 12100/2008.

REFERENCES

- [1] P. Stewart, J. Zavala, and P. Flemming, "Automotive drive by wire controller design by multi-objectives techniques," *Control Engineering Practice*, vol. 13, pp. 257–264, 2005.
- [2] D. Lefebvre, P. Chevrel, and S. Richard, "An hinfinity-based control design methodology dedicated to the active control of vehicle longitudinal oscillations," *IEEE Transactions on Control System Technology*, vol. 11, 2003.
- [3] M. Berriri, P. Chevrel, and D. Lefebvre, "Active damping of automotive powertrain oscillation by a partial torque compensator," *Control Engineering Practice*, vol. 16, pp. 874–883, 2008.
- [4] P. Rostalski, T. Besselmann, M. Maric, F. Van Belsen, and M. Morari, "A hybrid approach to modelling, control and state estimation of mechanical systems with backlash," *International Journal of Control*, vol. 80, pp. 1729–1740, 2007.
- [5] M. Lazar, "Flexible control Lyapunov functions," in *28th American Control Conference*, St. Louis, MO, USA, 2009, pp. 102–107.
- [6] U. Kiencke and L. Nielsen, *Automotive control systems: for engine, driveline, and vehicle*. Springer Verlag, 2005, vol. 290.
- [7] C. F. Caruntu, A. E. Balau, M. Lazar, P. P. J. van den Bosch, and S. Di Cairano, "A predictive control solution for driveline oscillations damping," in *Hybrid Systems: Computation and Control*, Chicago, USA, 2011, pp. 181–190.
- [8] K. Natori, T. Tsuji, K. Ohnishi, A. Hase, and K. Jezernik, "Robust bilateral control with internet communication," in *30th Annual Conference of the IEEE Industrial Electronics Society*, Busan, Korea, 2004.
- [9] C. F. Caruntu and C. Lazar, "Stabilizing MPC for network-controlled systems with an application to DC motors," in *IEEE International Conference on Mechatronics*, Istanbul, Turkey, 2011.
- [10] M. Lazar and W. P. M. H. Heemels, "Optimized input-to-state stabilization of discrete-time nonlinear systems with bounded inputs," in *27th American Control Conference*, Seattle, USA, 2008.
- [11] Z.-P. Jiang and Y. Wang, "Input-to-state stability for discrete-time nonlinear systems," *Automatica*, vol. 37, pp. 857–869, 2001.
- [12] C. F. Caruntu and C. Lazar, "Robust MPC for TrueTime simulation of a vehicle drivetrain controlled through CAN," in *accepted for IEEE International Conference on Emerging Technologies and Factory Automation*, Toulouse, France, 2011.
- [13] U. Klehmet, T. Herpel, K.-S. Hielscher, and R. German, "Delay bounds for CAN communication in automotive applications," in *14th GI/ITG Conference Measurement, Modelling and Evaluation of Computer and Communication Systems*, Dortmund, Germany, 2008.
- [14] M. Lazar, "Model predictive control of hybrid systems: Stability and robustness," Ph.D. dissertation, Eindhoven University of Technology, The Netherlands, 2006.

Influence of supervisory control on the dependability of a hybrid system

Gabriel Antonio Pérez Castañeda, Nicolae Brinzei, and Jean-François Aubry

Abstract— A dynamic hybrid system is described by a set of continuous variables and a set of discrete events interacting together. The reality also requires taking into account component failures and the stochastic behavior of the system. Some events or variables take then a stochastic character. The component failures are in fact unobservable events and partially controllable. We are going to show how the control theory might be a mean to help to the conception of safe systems assuming that changes in the system (introduction of safety barriers ...) would act as a supervisor in the context of simulation. The behavior of the system so “controlled” may be established without having to rebuild a complete model of the modified system. This assumes that the propagation of dangerous initiating events is inhibited and then the probability of dangerous states of the system reduced to zero.

I. INTRODUCTION

The dynamic reliability is the discipline that takes into account the interactions between the dynamic and deterministic behavior of a system and the stochastic behavior (failures and repairs) of its components. Several approaches have been developed around the concept of dynamic reliability [1, 2]. Analytical solutions were applied to solve some of the problems inherent to this concept [2]-[5]. However, the mathematical complexity of the analytical assessment of the reliability of complex dynamic hybrid systems leads often to resort to simulation. Therefore a Stochastic Hybrid Automaton (SHA) has been formally defined [6, 7] and implemented in the simulation Scilab-Scicos environment. The interest of implementing the SHA is, at first, to take into account the different continuous functional modes of the system defined in different discrete states of the automaton and the transitions from one to another on deterministic or stochastic events occurrences. Secondly, the automaton handles without difficulty the interactions between the continuous physical variables of the system and the random variables associated to the failures of the system’s components. The SHA integrates all the

possible behaviors of the system. A simulation (a history) is the evolution of the SHA from the initial state, at the initial time, to a deadlock state or up to a predefined mission time. After a number of simulations large enough to “visit” all of the various discrete states of the model many times enough, we can access to the dependability assessment by statistics (Monte Carlo method).

In this context, the assessment studies of the predictive dependability of systems must take into account the following aspects [8]-[10]:

- the dynamic interactions between the physical parameters (usually represented by continuous variables) and the nominal or dysfunctional behavior of the components (usually represented by the occurrence of events);
- the deterministic or stochastic character of the events and of the physical variables;
- the evolution over time (reconfiguration) of the reliability structure (indicating how the reliability of the system depends on the reliability of its parts);
- the multiple modes of aging of the components according to the system discrete state;
- the non-binary behavior of the components; the consideration of the time instant and of the occurrence order of the events associated to the changes of the discrete state. These events can correspond to component failures or to continuous variables thresholds.

The mathematical complexity of the analytical dependability assessment [8] is only possible under certain assumptions or when the system is reasonably complex [11, 12]. In order to overcome this mathematical complexity, the Monte Carlo simulation was explored [6].

In this article we present the interest of control by supervision on the system’s dependability in a dynamic reliability context. From the observation of an event sequence we want to know if an unobservable event, such as component failure, has occurred. The model consists of three Scicos blocks: a stochastic hybrid automaton representing the plant, an observer automaton and a supervisor. The supervisor invalidates the undesirable events occurring in the plant. To identify the occurrence of observable events we used the observer automaton which is composed by an observable event identifier block and a finite-state automaton block. Finally, we show that the presence of the supervisor improves the dependability indicators, and especially the probability of dangerous state.

Gabriel Antonio Pérez Castañeda is teacher at the Instituto Tecnológico de Tehuacán, Metal-Mechanical Department and Laboratory, Tehuacán, Puebla, Mexico (email: perezc76@ensem.inpl-nancy.fr, gapercas@yahoo.com).

Nicolae Brinzei is associate professor at the Nancy-University, Institut National Polytechnique de Lorraine, Research Centre for Automatic Control, CNRS UMR 7039, Vandoeuvre-lès-Nancy, France, (corresponding author, phone: +33.3.83.59.56.36; fax: +33.3.83.59.56.44; email: Nicolae.Brinzei@ensem.inpl-nancy.fr).

Jean-François Aubry is Emeritus Professor at the Nancy-University, Institut National Polytechnique de Lorraine, Research Centre for Automatic Control, CNRS UMR 7039, Vandoeuvre-lès-Nancy, France (email: Jean-Francois.Aubry@ensem.inpl-nancy.fr).

II. STOCHASTIC HYBRID AUTOMATON

Hybrid automata have been defined formally by Alur [13] in the aim of the specification and verification of hybrid dynamic systems and real-time systems [14]. These automata model the dynamic hybrid systems as a timed finite state automaton provided with real variables evolving continuously in the time.

In order to take into account the failures of components (random phenomenon) and the quantitative assessment of the system dependability parameters, an extension of the model proposed by Alur has been introduced and implemented by Castaneda et al.: the stochastic hybrid automaton (SHA) [7]. The stochastic hybrid automaton takes into account the different continuous operation modes of the system and the transition from one to another on the occurrence of deterministic or stochastic events. The first ones are produced by threshold crossings of continuous variables, the seconds ones are produced by component failures characterized by their probability distribution laws. The continuous dynamics of the system are defined by ordinary differential equations.

The stochastic hybrid automaton has been defined as an 11-tuple:

$$(\mathcal{X}, \mathcal{E}, \mathcal{A}, \mathcal{X}, \mathcal{A}, \mathcal{H}, \mathcal{F}, \mathcal{P}, x_0, \dot{x}_0, P_0) \quad (1)$$

where:

- $\mathcal{X}$ is a finite set of discrete states;
- $\mathcal{E}$ is a finite set of determinist or stochastic events;
- $\mathcal{A}$ is a finite set of arcs in the form of (x_e, G, R, x') where x and x' are respectively the begin and the goal discrete states of the arc, e is the event associated of the arc, G is the guard condition and R is the reset function;
- $\mathcal{X}$ is a finite set of real variables evolving in the time;
- $\mathcal{A} : \mathcal{X} \times \mathcal{X} \rightarrow (\mathbb{R}^{n^+} \rightarrow \mathbb{R})$ is a function of “activities” associating to each $\mathcal{X} \times \mathcal{X}$ element a function defined on $\mathbb{R}^{n^+}$ and with values in $\mathbb{R}$;
- $\mathcal{H}$ is a finite set of clocks;
- $\mathcal{F} : \mathcal{H} \rightarrow (\mathbb{R} \rightarrow [0,1])$ is an application associating a probability distribution function to each clock;
- $\mathcal{P} = [p_i^j]$ is a matrix of transition probability distributions where p_i^j is a probability distribution of state $p(x^i | x^j, e_q)$. For instance, let e_q be the same event defining the transitions of the discrete state x^j to the discrete states $x^1, x^2, \dots, x^j$ (so there are j transitions in conflict, the finite state automaton underlying would be not deterministic); let p_1^j be the probability of going from the state x^j to the state x^1 , the probability p_2^j of going from the state x^j to the state x^2 and the probability p_j^j of going from the state x^j to the state x^j , with $p_1^j + p_2^j + \dots + p_j^j = 1$.
- $x_0, \dot{x}_0$ and P_0 correspond respectively to the initial

discrete state, the initial value of the continuous state vector in the discrete initial state and the initial distribution of transition probabilities.

The elements $\mathcal{X}, \mathcal{E}$ and $\mathcal{A}$ of the stochastic hybrid automaton correspond to the finite state automaton defining its discrete part. In contrast, $\mathcal{X}, \mathcal{A}, \mathcal{R}$ and $\mathcal{G}$ define its continuous part. $\mathcal{H}$ corresponds to its timed aspect and finally $\mathcal{F}$ and $\mathcal{p}$ express its stochastic aspect. The performance and repair duration of components are materialized by the clock set $\mathcal{H}$. The probability distribution $\mathcal{p}$ can model different behaviors of the system under the occurrence of a same event, and so the non-determinism is solved by a probabilistic choice.

The stochastic hybrid automaton has been implemented in Scicos-Scilab. Scilab [15], a free open-source package for scientific computation. It is an interpreted language with dynamically typed objects. Scilab includes hundreds of general purpose and specialized functions for numerical computation, organized in libraries called toolboxes that cover such areas as simulation, optimization, systems and control, and signal processing. In addition, Scicos [16] is a Scilab toolbox for modeling and simulation of dynamical systems. It is particularly useful for modeling systems where continuous-time and discrete-time components are interconnected. Scicos provides a modular way to construct complex dynamical systems using a block diagram editor. Scicos handles, in particular, the interaction between continuous-time dynamics and system events associated with the timing of a discrete-time clock [17].

The SHA has been implemented in a Scicos block [18-19] (DyRelA tool). The block is also named SHA. It is defined by two functions. The first one, referred to as the *interfacing function*, must be in the Scilab language. That function is used to draw, connect the block, and to modify its parameters. In addition, the interfacing function initializes block states and specifies certain block properties. It also specifies the name of the second function to be used by the simulator. The second function, which is called the *computational function*, is normally written in C, but can also be a Scilab function. This function defines the behavior of the block during the simulation and it is called in various ways by the simulator.

The SHA block has three output ports, see Fig. 1. The output port marked $[x_b, x_{t-1}]$ provides the current discrete state of the system as well as the precedent state. The output port marked $[X, \dot{X}]$ provides the values of continuous state variables as well as their derivatives. The output port (at the bottom) marked “e” produces an event each time a discrete state transition is made.

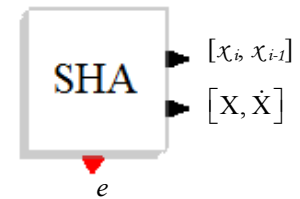


Fig. 1. Scicos block of the Stochastic Hybrid Automaton

The computational function defines all of the three aspects of the SHA: the discrete part, the continuous part and the timed part. The discrete part is defined in terms of discrete states, transitions between these states and events controlling these transitions. The continuous part is defined by means of the continuous dynamics, the guard conditions and the initializations conditions for each of the discrete states. A random generator dedicated to the production of the clock structure $\mathcal{H}$ is also defined in this function and materializes the timed aspect of the SHA. The random generator executes random draws corresponding to the random transitions between the involved states. Whenever a state transition occurs the output e of the automaton bloc generates an event and, at this time, the random generator draws the random values for the clock $\mathcal{H}$.

III. SUPERVISORY CONTROL

The theory of the supervision of discrete event systems (DES) was initiated in 1987 by Ramadge and Wonham in [20]. In this approach the time is not taken into account and the study takes place at a qualitative level (control trajectory validation). The system to be controlled is considered as a DES evolving only on occurrence of events. The operation of such a system can be described by a sequence set of these events. A supervisor (who is itself a DES), observes the events following one another in a discrete event system and is able to authorize or prohibit the occurrence of some future events expected by the system to globally change its behavior.

Cassandras and Lafortune [21] presented a theory of supervision where the DES is modeled at a logical level by an automaton $\mathcal{G}$ with an unsatisfactory behavior that should therefore be amended. To alter the behavior of $\mathcal{G}$, we have introduced a supervisor $\mathcal{S}$. The behavior of $\mathcal{G}$ is changed by the equivalent of a “feedback loop” (Fig. 2) to obtain a new behavior conforms to a set of specifications.

A. General principle of supervisory control

A DES coupled to a supervisor may be perceived as receiving at any time the list of events whose occurrence is allowed. This system is called supervised discrete event system and in this system the occurrence of an event can take place only if authorized by the supervisor. Basically, the observation of the system by the supervisor is asynchronous.

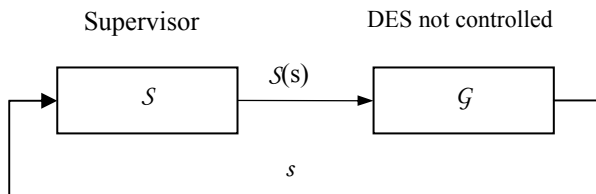


Fig. 2. The supervisor and the plant as the DES

The occurrence of an observable event lead the supervisor to a new state which immediately gives the list of controllable events whose occurrence is validated. Some events generated by the system cannot be disabled. Such event will be called uncontrollable event. However, we will call controllable event any event that may be disabled at any time. An uncontrollable event is an unpredictable event such as a failure. It may also be due to a physical constraint, as a limitation of sensors, saturation or overflow operations. Moreover, the phenomena of sensor limitation and local events not transmitted in distributed systems may be examples of unobservable events.

B. Proposed Approach

As the supervisor can modify the access to the dangerous states of a system, the supervisory control is a part of the dependability means. However, like all others parts, its effectiveness must be evaluated before its implementation. In the dynamic context, reliability evaluation is only accessible by simulation as said in part I. For that, we had to build the appropriate model. We implemented the necessary tools in the Scilab-Scicos environment. The hybrid dynamic system and its control by supervision are described by the diagram in Fig. 3. The SHA block representing the system to be controlled (the plant) was improved by the adjunction of an input, to receive, from the supervisor, the list of the events to be disabled. So, the behavior of the discrete part of the system is satisfactorily changed in order to avoid for example that the SHA enter a dangerous state.

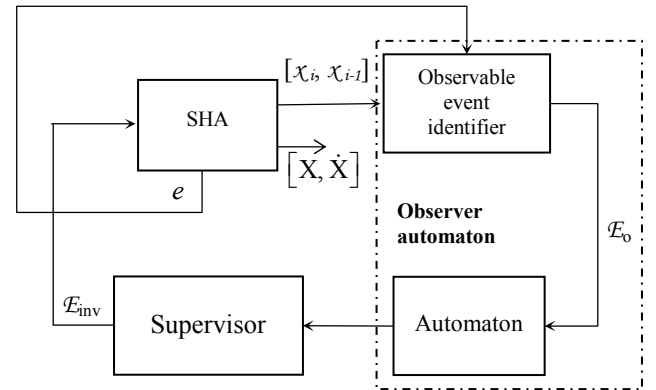


Fig. 3. Hybrid dynamical system and its control by supervision
 E_{inv} – events to invalid; E_o – observable events

The operation of the control system by supervision is quite simple. The initial system is modeled by the stochastic hybrid automaton. The SHA block manages the sequence of events related as well to the continuous part (crossing threshold) as to the discrete one (failure, repair ...). The “observer automaton” block consists of two blocks, the observable event identifier and the automaton model of the system observed. The first one identifies the events of the SHA from the knowledge of the previous and current states of each occurrence of event of the SHA. In addition, it has to ignore the events predefined as unobservable. The second

block noted “automaton” is the observable model of the system that evolves on the occurrence of observable events. This block provides on its output the number of current observed state. From this information the supervisor determines the controllable events to be invalidated which are transmitted to the SHA block in order to change the system behavior.

IV. TEMPERATURE CONTROL SYSTEM OF A OVEN

To illustrate the action of the control system by supervision a dynamic system is chosen into consideration: the temperature control system of an oven with is composed by two controllers: a PI type and an On-off type.

A. System Description

The system shown in Fig. 4 contains two control loops. The first loop contains a Proportional and Integral (PI) controller which maintains the temperature at a desired level T_{ref} . When PI fails, the second control loop is launched: the on-off (ON/F) controller. The ON/F controller maintains the temperature T within a given range: $T_{infON/F} \leq T \leq T_{supON/F}$. Precisely, when the temperature goes below the lowest value $T_{infON/F}$ (respectively above the highest value $T_{supON/F}$), the heater is switched on a maximal power (respectively switched off), when the temperature is within the range no control action is carried out.

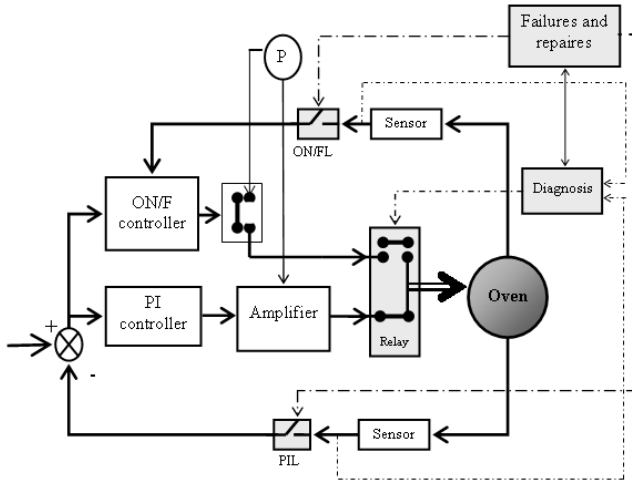


Fig. 4. Structural diagram of the temperature control system of an oven

B. System behavior

The system works as follows: at the beginning the oven temperature is controlled by the PI controller. After a random time the controller fails (with a failure rate λ_{PI}) and the temperature increases rapidly. The diagnosis detects that the temperature has reached a dangerous value ($T \geq T_{max}$) deducing then that the oven temperature is out of control. It gives the order to switch the relay on the ON/F loop, which control the oven temperature and the repair process of the PI controller (with repair rate μ_{PI}) starts. However, the possibility of the ON/F failure exists, after another random time ($\lambda_{ON/F}$). The repair process of the ON/F controller starts

(characterized by a repair rate $\mu_{ON/F}$). It was considered that the oven is faultless.

The threshold crossing detection system may be itself submitted to failures (λ_{Smin} and λ_{Smax} are the failure rates of the sensors detecting the minimum threshold and, respectively, the maximum threshold). When these sensors failed, the system reaches a dangerous state.

The states-transitions diagram and the description of the discrete states, of the transitions and events and of the temperature evolution in each discrete state is given in the Fig. 5 and table I.

V. SIMULATION AND DEPENDABILITY ASSESSMENT

A. Simulation of the SHA model of a hybrid systems

Firstly, the SHA model of the temperature control system, without supervisor, (Fig. 5 and table I) are implemented in Scilab/Scicos software and the results of one simulation trajectory are presented in Fig. 6. The top curve shows the evolution of the system current state over the time (according to the automaton of the figure 5) and the curve at the bottom shows the evolution of the oven temperature over the time.

When the system enter in the state 10 the control system is out of service, and the oven temperature is uncontrollable. The system reaches a dangerous state.

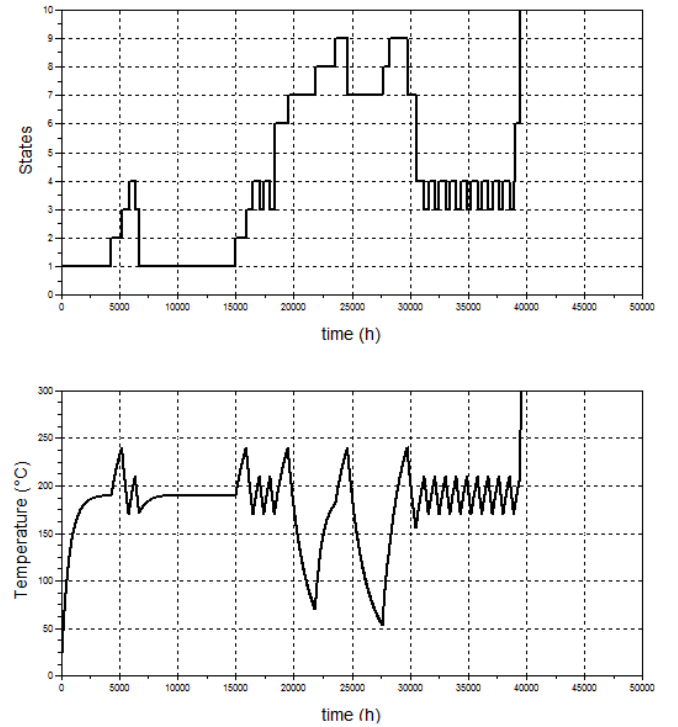


Fig. 6. Hybrid system simulation (discrete states evolution and continuous variable evolution) with the SHA model

The failure and repair rates of components are the following:

$$\begin{aligned} \lambda_{PI} &= 13 \cdot 10^{-5} \text{ h}^{-1}, & \mu_{PI} &= 21 \cdot 10^{-3} \text{ h}^{-1} & \lambda_{ON/F} &= 8 \cdot 10^{-5} \text{ h}^{-1}, \\ \mu_{ON/F} &= 14 \cdot 10^{-3} \text{ h}^{-1}, & \lambda_{Smin} &= \lambda_{Smax} &= 2 \cdot 10^{-4} \text{ h}^{-1}. \end{aligned}$$

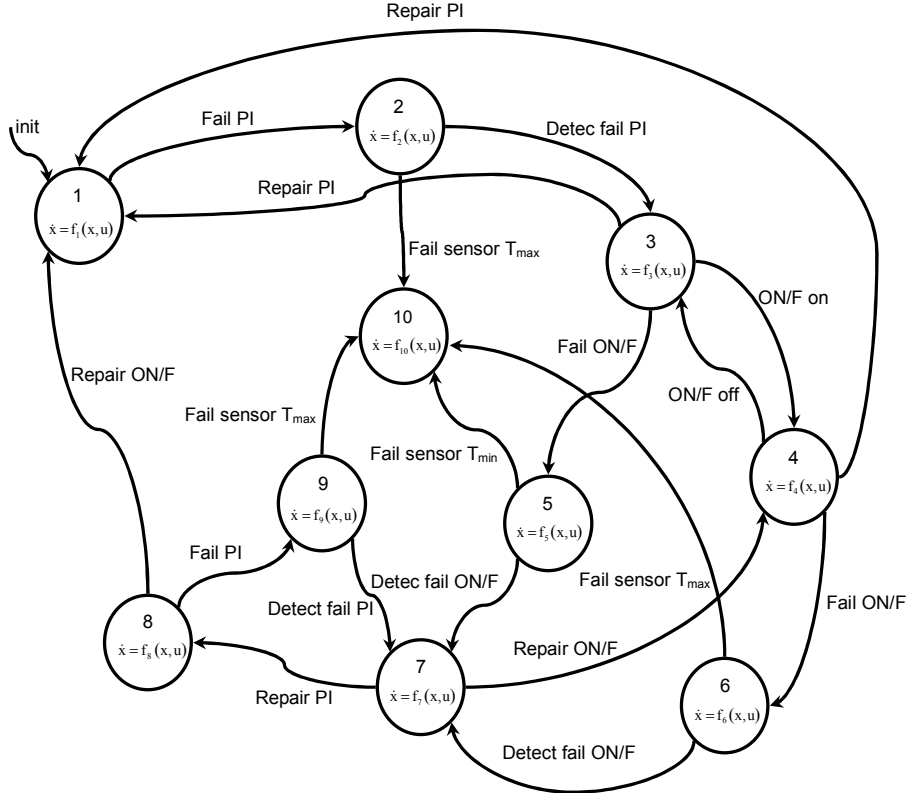


Fig. 5. States and transitions for the temperature control system

TABLE I. DESCRIPTION OF SYSTEM'S STATES AND TRANSITIONS

Current state χ	Current state description	Future state χ'	Events	Temperature evolution
1	PI: OK, active, ON/F: OK passive	2	λ_{PI}	$\dot{x} + 0.0015x - 0.0015u_{ref} = 0$
2	PI: FL, active, ON/F: OK passive	3	$T \geq T_{max}$	$1500\dot{x} + x - u_{map} = 0$
3	PI: FL, repair, ON/F: OK (off), active	1	μ_{PI}	$1500\dot{x} + x - u_{mip} = 0$
		4	$T \leq T_{inf\ ON/F}$	
		5	$\lambda_{ON/F}$	
4	PI: FL, repair, ON/F: OK (on), active	1	μ_{PI}	$1500\dot{x} + x - u_{map} = 0$
		3	$T \geq T_{sup\ ON/F}$	
		6	$\lambda_{ON/F}$	
5	PI: FL, repair, ON/F: FL (off), active	7	$T \leq T_{min}$	$1500\dot{x} + x - u_{mip} = 0$
		10	λ_{Smin}	
6	PI: FL, repair, ON/F: FL (on), active	7	$T \geq T_{max}$	$1500\dot{x} + x - u_{map} = 0$
		10	λ_{Smax}	
7	PI: FL, repair, ON/F: FL, repair	4	$\mu_{ON/F}$	$1500\dot{x} + e^{1/1500}x - u_s = 0$
		8	μ_{PI}	
8	PI: OK, active, ON/F: FL, repair	1	$\mu_{ON/F}$	$\dot{x} + 0.0015x - 0.0015u_{ref} = 0$
		9	λ_{PI}	
9	PI: FL, active, ON/F: FL, repair	7	$T \geq T_{max}$	$1500\dot{x} + x - u_{map} = 0$
		10	λ_{Smax}	
10	Sensor failure, furnace incontrollable	-	-	$1500\dot{x} + x - u_{map} = 0$

“OK” the component is in the functioning state, “FL” the component is in the failure state, $u_{ref}=90^\circ\text{C}$ (reference temperature), $u_{map}=400^\circ\text{C}$ (temperature at maximal power), $u_{mip}=25^\circ\text{C}$ (temperature at minimal power), $u_s=25^\circ\text{C}$ (room temperature)

B. Dependability assessment of the controlled hybrid systems

In order to avoid that the hybrid system reaches the dangerous state 10, a supervisor was designed in according to the approach proposed in III.B and in Fig. 3. The set of all events occurring in this system is:

$$\mathcal{E} = \{\lambda_{PI}, \lambda_{ON/F}, \lambda_{Smin}, \lambda_{Smax}, \mu_{PI}, \mu_{ON/F}, T_{min}, T_{max}, T_{infON/F}, T_{supON/F}\}$$

The sub-set of the observable events is composed by the following thresholds crossing and repair events:

$$\mathcal{E}_o = \{\mu_{PI}, \mu_{ON/F}, T_{min}, T_{max}, T_{infON/F}, T_{supON/F}\}$$

The sub-set of the unobservable events is composed by the following failure events:

$$\mathcal{E}_{uo} = \{\lambda_{PI}, \lambda_{ON/F}, \lambda_{Smin}, \lambda_{Smax}\}$$

According to [21], the observer automaton was designed to identify the state of the hybrid system (SHA) with respect to observable events. The observed state is the input of the supervisor (Fig. 3). The supervisor was designed by a classical Kumar algorithm [21] and its task is to prohibit the hybrid system (SHA) entering the dangerous state 10. This supervisor was implemented in Scilab/Scicos software and, by means of Monte-Carlo simulation (10,000 histories on a mission time of 10^6 hours), the cumulative probability of the access to a dangerous state was evaluated with and without the presence of the supervisor (Fig. 7). These results show that without a supervisor the hybrid system reaches the dangerous state with a non-zero probability (and certainly in the case of a long mission time). In the presence of a supervisor, the probability of the dangerous state access is always equal to zero because the supervisor prohibits the access of the dangerous state.

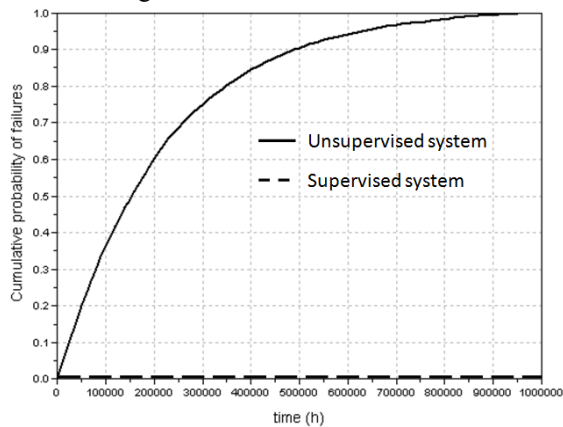


Fig. 7. Cumulative probability of the dangerous state access

VI. CONCLUSIONS

In this paper we used the stochastic hybrid automaton as a convenient tool to allow modeling and evaluation of the dependability of a hybrid system. This is an actually problem in dependability research domain, known as dynamic reliability. In order to improve the dependability indicators and more particularly the access probability to a dangerous state, the theory of supervisory control has been used. Thus the introduction of a supervisor in a feed-back

loop on the stochastic hybrid automaton model of a hybrid system completely prohibits the access of a dangerous state and, in this manner, contributes to increase the dependability of a system and to design safer systems.

REFERENCES

- [1] N. Siu, "Risk assessment for dynamic systems: An overview", *Reliability Engineering and System Safety*, vol. 43, pp. 43-73, 1994.
- [2] P. E. Labeau, C. Smidts and S. Swaminathan, "Dynamic reliability: towards an integrated platform for probabilistic risk assessment", *Reliability Engineering and System Safety*, vol. 68, pp. 219-254, 2000.
- [3] J. Devooght, C. Smidts, "Probabilistic reactor dynamics – I: The theory of continuous event trees", *Nuclear science and engineering*, vol. 111, pp. 229-240, 1992.
- [4] C. Smidts, C. Devooght, "Probabilistic reactor dynamics – II: A Monte Carlo study of a fast reactor transient", *Nuclear science and engineering*, vol. 111, pp. 241-256, 1992.
- [5] C. Coccozza-Thivent, M. Desgrouas, S. Mercier, "Algorithme de calcul de disponibilité asymptotique en fiabilité dynamique", *15ème Congrès en Maîtrise de risques et sûreté de fonctionnement, Lambda Mu 15*, Lille, France, 2006.
- [6] G. A. Pérez Castañeda, *Evaluation par simulation de la sûreté de fonctionnement de systèmes en contexte dynamique hybride*, PhD thesis of the Institut National Polytechnique de Lorraine, Nancy, France, 2009.
- [7] G. A. Pérez Castañeda, J. F. Aubry, N. Brinzei, "Stochastic hybrid automata model for dynamic reliability assessment", *Proceedings of the Institution of Mechanical Engineers Part O Journal of Risk and Reliability*, vol. 225, no. 1, pp. 28-41, 2011.
- [8] C. Kermish, P. E., Labeau, *Approche dynamique de la fiabilité des systèmes. Tâche n°1 : établissement de l'état de l'art en fiabilité dynamique*, Projet 6/2000 de l'ISdF, Université Libre de Bruxelles, 2000.
- [9] P. E. Labeau, C. Smidts, S. Swaminathan, "Dynamic reliability: towards an integrated platform for probabilistic risk assessment", *Reliability Engineering and System Safety*, vol. 68, pp. 219-254, 2000.
- [10] F. Dufour, Y. Dutuit, "Dynamic Reliability: a new model", *European Safety and Reliability Conference, ESREL 2002*, Lyon, France, 2002.
- [11] M. Belhadj, T. Aldemir, "Some computational improvements in process system reliability and safety analysis using dynamic methodologies", *Reliability Engineering System Safety*, vol. 52, pp. 339-347, 1996.
- [12] C. Coccozza-Thivent, R. Eymard, "Algorithmes de fiabilité dynamique", *15ème Congrès en Maîtrise de risques et sûreté de fonctionnement, Lambda Mu 15*, Lille, France, 2006.
- [13] R. Alur, C. Courcoubetis, T. A. Henzinger, P. H. Ho, "Hybrid automata: an algorithmic approach to the specification and verification of hybrid systems", *Lecture Notes in Computer Science*, vol. 736, pp. 209-229, 1992.
- [14] R. Alur, M. Hill, D. Dill, "Automata theoretic verification of real-time systems", *Formal Methods for Real-Time Computing, Trends in Software Series*, John Wiley & Sons Publishers, pp. 55-82, 1995.
- [15] Scilab, Scientific Software, available at <http://www.scilab.org>
- [16] Scicos, available at: <http://www.scicos.org>
- [17] S. L. Campbell, J.-P. Chancelier, R. Nikoukhah, *Modeling and Simulation in Scilab/Scicos*, Springer Verlag, 2006.
- [18] G. A. Pérez Castañeda, J. F. Aubry, N. Brinzei, "DyRela (Dynamic Reliability and Assessment)", *1st Workshop on Dynamic Aspects in DEpendability Models for Fault-Tolerant Systems, DYADEM-FTS 2010 in conjunction with European Dependable Computing Conference EDCC 8*, Valence (Spain), 2010.
- [19] G. A. Pérez Castañeda, J. F., Aubry, N. Brinzei, "Amélioration de l'outil DyRela pour une évaluation de la Sûreté de fonctionnement des systèmes dynamiques", *9ème édition du Congrès International Pluridisciplinaire Qualita 2011*, Angers, France, March 22-25 2011.
- [20] P. J. Ramadge, W. M. Wonham, "Supervisory control of a class of discrete event processes", *SIAM Journal on Control and Optimisation*, vol. 25, no. 25, pp. 206-230, 1987.
- [21] C. G. Cassandras, S. Lafortune, *Introduction to Discrete Event Systems*, Springer Verlag, 2010.

Model Following with Output Feedback for Fault Tolerant Control

Elena M. Cimpoeșu, Bogdan D. Ciubotaru, and Dumitru Popescu

Abstract—The increasing control demands for safety-critical applications have led to the need for a new type of control system, capable of guaranteeing that the desired system performances are met, as well as ensuring the system operational safety, both under normal operating conditions and in the presence of system faults. Such systems are known in the literature as Fault Tolerant Control Systems (FTCS) regarding the procedure of redesigning a controller after a fault occurs so that the control objectives from the nominal regime are still achieved - this is referred to as fault accommodation. This paper investigates and applies two controller reconfiguration strategies based on the principles of Model Following (MF) control, namely the Implicit MF (IMF) and the Explicit MF (EMF) approaches, while developing the new control configurations where output feedback is used. The advantages and disadvantages of each technique are briefly discussed and an application example is provided showing how the I/E-MF can be used in aircraft Fault Tolerant Control (FTC).

I. INTRODUCTION

During the past years, increasing progress has been made with regard to the automatic control of complex processes, including safety-critical applications, like chemical and petrochemical installations, nuclear power plants, aircraft and spacecraft. Also, in the last decades, significant effort has been made with respect to the design of control strategies and the development of algorithms for control systems that are able to ensure closed-loop stability and to meet the performance requirements, both under nominal operating conditions and in the presence of potential system faults, thus accommodating component failures automatically. These types of control systems are known as Fault Tolerant Control Systems (FTCS) [1], [2] (see also [3]); they are further classified as Passive (PFTCS) and Active (AFTCS), respectively.

Active FTCS react to the system component failures by reconfiguring control actions that may consist of selecting a pre-computed control law to compensate for the impact of a specific fault or of synthesizing a new control law on-line see [4]. This way, AFTCS ensure preserving stability and providing acceptable performance of the entire system, although a slight performance degradation is accepted in some cases. In their turn, AFTCS can be classified according to several criteria like the mathematical tools, the design approaches, the reconfiguration mechanisms or the controlled process, see again [2].

E.M. Cimpoeșu, B.D. Ciubotaru, and D. Popescu, are with APCC, Automatic Process Control and Computers Laboratory, Polytechnic University of Bucharest, 060042 Bucharest, Romania bogdandc@indinf.pub.ro

To this scope, extended research has been carried out regarding all Fault Tolerant Control (FTC) methods in the past decades, resulting in a series of fault accommodation strategies and algorithms. However, some of the most reliable FTC methods, while still acceptable from the point of view of their computational complexity, seem to be the ones based on the Model Following (MF) principle.

This paper represents a continuation to the research activity that led to the results presented in [5] and is dedicated to the interrogation of the Model Following technique and to the development and application of its subsequent Implicit and Explicit (I/E-MF) approaches using the output feedback. The paper is organized as follows: in Section II, MF is synthesized and the I/E-MF output feedback approaches are developed and analyzed, while in Section III they are applied to an impaired aircraft longitudinal model. Finally, some concluding remarks are presented in Section IV.

II. IMPLICIT AND EXPLICIT MODEL FOLLOWING WITH OUTPUT FEEDBACK

A. Design Idea

The basic idea of the Linear Model Following (LMF) approach as described in [6] consists of a state-space design methodology by which a control system is designed in order to make the output or the state vector trajectories of the real plant follow the ones of a reference system as closely as possible, even in the presence of system faults, see also [7] and [8].

In this approach, the design objectives are incorporated into the reference model, thus avoiding a great difficulty in control system design, namely the need to express the design specifications directly in terms of controller parameters. Also, both feedback and feedforward controllers are used in order to achieve the objectives of the LMF problem.

B. Ideal and Nominal Operation

Assuming that the reference model and the plant are of the same dimension, let the reference model be described by the following state-space representation

$$M_m : \begin{cases} \dot{x}_m(t) = A_m x_m(t) + B_m u_m(t), \\ y_m(t) = C_m x_m(t), \end{cases} \quad (1)$$

where $x_m \in \mathbb{R}^n$ is the state vector, $u_m \in \mathbb{R}^m$ is the control input and $y_m \in \mathbb{R}^p$ represents the monitored output of the reference model. Also, $A_m \in \mathbb{R}^{n \times n}$, $B_m \in \mathbb{R}^{n \times m}$ and $C_m \in \mathbb{R}^{p \times n}$ are the matrices describing the reference system behavior.

Of course, in system reconfiguration it is desired that the accommodated system behavior is as close as possible to the nominal system behavior.

Therefore, let us consider the Linear Time Invariant (LTI) representation of the open-loop system from the nominal case

$$M_n : \begin{cases} \dot{x}_n(t) = A_n x_n(t) + B_n u_n(t), \\ y_n(t) = C_n x_n(t), \end{cases} \quad (2)$$

where $x_n \in \mathbb{R}^n$ is the state vector of the nominal system, $u_n \in \mathbb{R}^m$ is the nominal control input, $y_n \in \mathbb{R}^p$ represents the measured system outputs, while $A_n \in \mathbb{R}^{n \times n}$, $B_n \in \mathbb{R}^{n \times m}$ and $C_n \in \mathbb{R}^{p \times n}$ are the matrices that describe the system behavior.

Next, let us consider the output feedback gain $K_n \in \mathbb{R}^{m \times p}$ that guarantees closed-loop stability and the desired performances for the nominal system.

The closed-loop system behavior from the normal functioning regime will then be described by the following state-space representation

$$\hat{M}_n : \begin{cases} \dot{x}_n(t) = A_n x_n(t) - B_n K_n y_n(t) = \hat{A}_n x_n(t), \\ y_n(t) = C_n x_n(t), \end{cases} \quad (3)$$

where $\hat{A}_n = A_n - B_n K_n C_n$ is the closed-loop system matrix.

Throughout this paper, the system in (3) will represent the reference model for system reconfiguration using the LMF approach.

C. Faulty Operation

Now, consider that a multiplicative fault occurs that affects a system component, be it a sensor, an actuator or the actual plant. Assuming there are no changes in the dimension of the faulty system and also that an ideal FDD (Fault Detection and Diagnosis) module exists, providing up-to-date information about the system status at all times, with no delay, then the post-fault model of the plant as estimated by the FDD module is described by the following state-space representation

$$M_f : \begin{cases} \dot{x}_f(t) = A_f x_f(t) + B_f u_f(t), \\ y_f(t) = C_f x_f(t), \end{cases} \quad (4)$$

where $x_f \in \mathbb{R}^n$ is the state vector of the impaired system, $u_f \in \mathbb{R}^m$ is the control input for the faulty functioning regime, $y_f \in \mathbb{R}^p$ represents the measured system outputs, and matrices $A_f \in \mathbb{R}^{n \times n}$, $B_f \in \mathbb{R}^{n \times m}$ and $C_f \in \mathbb{R}^{p \times n}$ describe the new system behavior.

Further, let the distance between the reference model and the impaired plant be described by the difference between the measured outputs of the two systems

$$e(t) = y_m(t) - y_f(t). \quad (5)$$

The following subsection analyzes the concept of Perfect Model Following with regard to the output error e introduced in (5).

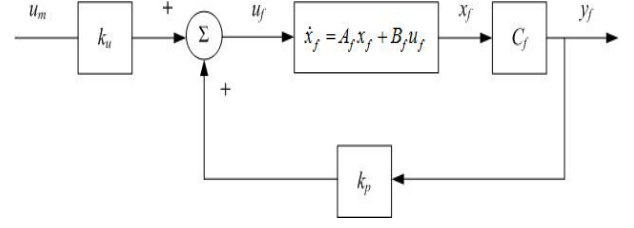


Fig. 1. IMF Control Configuration

D. Perfect Model Following

Generally speaking, the goal of the Perfect Model Following (PMF) is to ensure that the state or output variables of the faulty plant have the same trajectory as those of the reference model; in this case, for any control $u_m(t)$ piecewise continuous and $e(0) = 0$, one should have $e(t) = 0, \forall, t > 0$.

The sufficient conditions for achieving PMF have been identified in [9] as being

$$\begin{cases} 0 = (I - B_f B_f^\dagger)(A_m - A_f), \\ 0 = (I - B_f B_f^\dagger)B_m, \end{cases} \quad (6)$$

where $B_f^\dagger$ represents the pseudo-inverse of matrix B_f .

As emphasized in [7], [8], the Erzberger conditions are rather restrictive and show that, for an arbitrary plant, it may be very difficult to find an appropriate reference model in order to achieve PMF.

The following subsections develop two different model following control configurations that highlight the advantages and disadvantages of the two different approaches, namely the Implicit and the Explicit techniques.

The development performed further on follows the line indicated in [7], [8], but it is new as it focuses on the output feedback approach, instead of the state feedback, since this concerns the minimization of the computational effort and the effects of the possible approximation errors that are due to the estimation of the state variables of the impaired model, for instance when these are not directly measurable. This is especially useful when a configuration that requires the reference model to be simulated on-line is chosen.

E. Implicit Model Following (IMF)

The control system configuration of the Implicit Model Following (IMF) does not explicitly contain the reference model, although it is used to obtain the control parameters - a standard control configuration of the IMF is shown in Fig. 1.

From (1), (4) and (5), the following equality can be written

$$\begin{aligned} \dot{e}(t) &= C_m A_m C_m^\dagger e(t) \\ &+ (C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) y_f(t) \\ &+ C_m B_m u_m(t) - C_f B_f u_f^{IMF}(t), \end{aligned} \quad (7)$$

where

$$u_f^{IMF}(t) = k_p y_f(t) + k_u u_m(t) \quad (8)$$

is the control input for the faulty plant, k_p is the feedback control gain and k_u is the feedforward control gain; also $C_f^\dagger$ and $C_m^\dagger$ represent the pseudo-inverses of matrices C_f and C_m , respectively.

In order for the PMF to be achieved, the control parameters should be chosen so that

$$\dot{e}(t) = C_m A_m C_m^\dagger e(t), \quad (9)$$

or equivalently

$$\begin{aligned} C_f B_f u_f^{IMF}(t) \\ = (C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) y_f(t) + C_m B_m u_m(t). \end{aligned} \quad (10)$$

The solution to (10), if it exists, takes the following form

$$\begin{aligned} u_f^{IMF}(t) &= B_f^\dagger C_f^\dagger (C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) y_f(t) \\ &+ B_f^\dagger C_f^\dagger C_m B_m u_m(t), \end{aligned} \quad (11)$$

which means that the control parameters computed via IMF are

$$k_p = B_f^\dagger C_f^\dagger (C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) \quad (12)$$

and

$$k_u = B_f^\dagger C_f^\dagger C_m B_m; \quad (13)$$

the sufficient conditions for PMF are the ones stated in (6).

Even if PMF is not possible, it should be noted that the solution in (11) still minimizes the quality criterion consisting of the 2-norm of the difference between $\dot{e}$ and $C_m A_m C_m^\dagger e$, also equivalent to

$$\begin{aligned} \|(C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) y_f + C_m B_m u_m - C_f B_f u_f^{IMF}\|_2 \\ = \|\dot{e} - C_m A_m C_m^\dagger e\|_2 = J_f^{IMF}; \end{aligned} \quad (14)$$

however, when PMF is achieved, the value of this criterion is zero.

The main advantage of the IMF approach lies in the fact that the design of the control law does not explicitly involve the reference model; in other words, the reference model is not run on-line, which minimizes the control system complexity.

However, there are also disadvantages to this method, namely the fact that, when PMF is not achievable, the trajectory of y_f may differ significantly from that of y_m . Also, the accommodated closed-loop system stability is not guaranteed.

This is counteracted in the Explicit Model Following technique, where the stabilization procedure is first used, followed by a correction afterwards.

F. Explicit Model Following (EMF)

As opposed to the IMF approach, in Explicit Model Following (EMF) the controller design is based on a real model, which means that the reference model is implemented as part of the actual controller - a typical configuration is shown in Fig. 2.

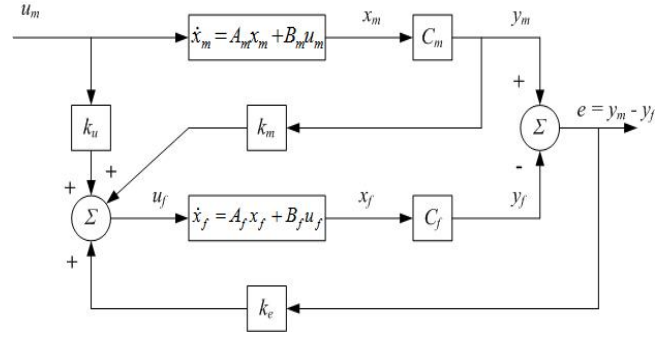


Fig. 2. EMF Control Configuration

By manipulating (1), (4) and (5), $\dot{e}(t)$ can be written as

$$\begin{aligned} \dot{e}(t) &= C_f A_f C_f^\dagger e(t) \\ &+ (C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) y_m(t) \\ &+ C_m B_m u_m(t) - C_f B_f u_f^{EMF}(t). \end{aligned} \quad (15)$$

As it can be seen from Fig. 2, in EMF there are three control matrices, namely k_u , k_m and k_e , and the control input of the faulty plant can be written as follows

$$\begin{aligned} u_f^{EMF}(t) &= u_f^{stab}(t) + u_f^{corr}(t) \\ &= (k_e e(t)) + (k_m y_m(t) + k_u u_m(t)), \end{aligned} \quad (16)$$

where

$$u_f^{stab}(t) = k_e e(t) \quad (17)$$

is the stabilizing gain, while

$$u_f^{corr}(t) = k_m y_m(t) + k_u u_m(t) \quad (18)$$

should be determined so that the criterion given by

$$\begin{aligned} \|(C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) y_m + C_m B_m u_m - C_f B_f u_f^{corr}\|_2 \\ = \|\dot{e} - C_f (A_f C_f^\dagger - B_f k_e) e\|_2 = J_f^{EMF} \end{aligned} \quad (19)$$

is minimized.

Obviously, PMF is achieved when

$$\dot{e}(t) = C_f (A_f C_f^\dagger - B_f k_e) e(t), \quad (20)$$

or equivalently

$$(C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) y_m(t) + C_m B_m u_m(t) = C_f B_f u_f^{corr}(t), \quad (21)$$

in which case the value of the above mentioned criterion is also zero; the sufficient conditions for PMF remain the ones stated in (6).

From (16) and (21), it can easily be deduced that the control parameters that minimize the quality criterion are

$$k_m = B_f^\dagger C_f^\dagger (C_m A_m C_m^\dagger - C_f A_f C_f^\dagger) \quad (22)$$

and

$$k_u = B_f^\dagger C_f^\dagger C_m B_m. \quad (23)$$

If the occurrence of a component fault leads the closed-loop system unstable, it is essential that the plant be first stabilized and then the fault be accommodated.

There are several methods for computing k_e so that $(A_f + B_f k_e C_f)$ has all its eigenvalues in the left half-plane, one of them being the LQR (Linear Quadratic Regulator) control design approach.

While more computationally complex than the Implicit approach, the EMF has the advantage of being able to guarantee closed-loop stability for the impaired plant whenever it is stabilizable.

Since a degradation in the performance of the closed-loop impaired system is considered to be acceptable in most cases, while the loss of stability cannot be tolerated, the EMF method is preferred in practical applications to the IMF method.

In the sequel, two fault scenarios are analyzed and the Implicit or Explicit MF techniques are used to accommodate the respective faults.

III. APPLICATION EXAMPLE OF I/E-MF CONTROL ON B747 AIRCRAFT LONGITUDINAL MODEL

A. Flight Condition

Consider the open-loop longitudinal model of a Class III large transport aircraft Boeing 747 in Class I cruise flight. The aircraft flight condition is assumed to be a straight and constant level flight at a fixed altitude of 40000 ft and 0.8 Mach. The corresponding steady-state speed is 774 fps and the aircraft mass is 19792 slug. The CoG (center of gravity) position is considered to be (0.25, 0, 0) of m.a.c. (mean aerodynamic chord).

B. Ideal Model

The aircraft longitudinal dynamics is described by the state vector $x = [u \ w \ q \ \theta]^T$, where u (fps) and w (fps) represent the inertial velocities in the x - and z - directions of the body-axis reference frame F_B , while q (rad/s) and θ (rad) represent respectively the pitch rate and the pitch angle. The control input δ_e (rad) is the elevator deflection.

The linearized open-loop model of the system is given by the set of matrices below, see [10],

$$A_n = \begin{bmatrix} -0.0069 & 0.0139 & 0 & -32.2000 \\ -0.0904 & -0.3147 & 773.9766 & 0 \\ 0.0001 & -0.0010 & -0.4284 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad (24)$$

$$B_n = \begin{bmatrix} -0.0002 & -18.0610 & -1.1577 & 0 \end{bmatrix}^T \text{ and} \quad (25)$$

$$C_n = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}. \quad (26)$$

Now, let

$$K_n = 0.6056$$

be the nominal output feedback control gain that ensures the closed-loop system stability; it was computed using LQR for the penalty matrices $Q_n = 10^{-4}$ and $R_n = 10^{-4}$.

K_n provides the closed-loop system $\hat{M}_n$, with $\hat{A}_n = A_n + B_n K_n C_n$, where

$$\hat{A}_n = \begin{bmatrix} -0.0069 & 0.0139 & -0.0001 & -32.2000 \\ -0.0904 & -0.3147 & 763.0383 & 0 \\ 0.0001 & -0.0010 & -1.1296 & 0 \\ 0 & 0 & 1.0000 & 0 \end{bmatrix},$$

which has the stable spectrum

$$\Lambda(\hat{A}_n) = \{-0.7227 \pm 0.7842i, -0.0028 \pm 0.0606i\}.$$

It is precisely the closed-loop nominal system $\hat{M}_n$ that will be considered the reference model throughout this section of the paper.

C. Implicit Model Following for the Stable Case

First, let us consider a fault that does not compromise the closed-loop system stability. Let the impaired system be described by matrices

$$A_f = \begin{bmatrix} -0.0065 & 0.0132 & 0 & -30.5900 \\ -0.0859 & -0.2990 & 735.2777 & 0 \\ 0.0001 & -0.0010 & -0.4070 & 0 \\ 0 & 0 & 0.9500 & 0 \end{bmatrix}, \quad (27)$$

$$B_f = \begin{bmatrix} -0.0001 & -13.5457 & -0.8683 & 0 \end{bmatrix}^T \text{ and} \quad (28)$$

$$C_f = \begin{bmatrix} 0 & 0 & 0.9 & 0 \end{bmatrix}, \quad (29)$$

that were obtained by considering a 25% loss-of-effectiveness in the elevator deflection, which can be modeled in the post-fault system by multiplying the state-control transfer matrix with $(1 - \tau_f) = 0.75$; this leads to $B_f = (1 - \tau_f)B_n$. Also, another fault was considered to affect the system matrix so that $A_f = (1 - \gamma_f)A_n$, where $\gamma_f = 0.05$, standing for a 5% structured identification error of the FDD module. Finally, a biased sensor type of fault was considered as well, that affects the system behavior so that $C_f = 0.9C_n$.

Before the controller reconfiguration, the closed-loop system matrix is given by $A_f + B_f K_n C_f$, which has all its eigenvalues in the left half-plane

$$\Lambda(A_f + B_f K_n C_f) = \{-0.5901 \pm 0.7892i, -0.0028 \pm 0.0592i\},$$

meaning that the closed-loop system is still stable.

In this case, the IMF method can be used in order to compute the new control law, thus obtaining the following control matrices

$$k_p = 0.0044$$

and

$$k_u = 0.0061.$$

The closed-loop system after reconfiguration has the stable spectrum

$$\Lambda(A_f + B_f k_p C_f) = \{-0.3549 \pm 0.8428i, -0.00031 \pm 0.0638i\}.$$

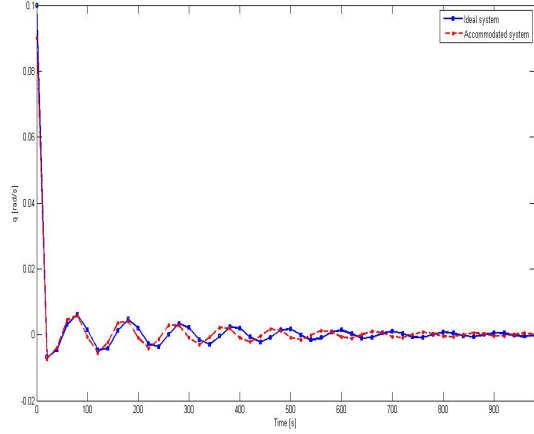


Fig. 3. Undriven Response of the Ideal/Accommodated Systems with IMF

The undriven response of the accommodated system compared with the one of the ideal (nominal) system is illustrated in Fig. 3.

Although it can be seen that the PMF is not achieved, the figure shows that the output variable of the accommodated system with the control matrices computed via IMF follow the reference trajectory closely.

The measure of the closeness between the two systems is given by the value of the norm criterion in (14) computed for the considered simulation time, which value is

$$J_f^{IMF} = 0.1363.$$

To summarize, the IMF technique can be effectively used for the accommodation of system faults that do not alter the closed-loop system stability, in which case it also guarantees the fact that the closed-loop system after the fault accommodation remains stable.

However, when a fault occurs that leads the system into instability, the IMF can no longer be used. Next, an example of such a fault scenario is analyzed in order to illustrate how the EMF can be used for system reconfiguration.

D. Explicit Model Following for the Unstable Case

Now, consider a new type of fault that affects the controlled system so that the closed-loop stability is lost and the state-space matrices that describe the impaired system behavior are

$$A_f = A_n, B_f = \begin{bmatrix} -0.0001 \\ -13.5457 \\ 8.6830 \\ 0.0000 \end{bmatrix}, C_f = \begin{bmatrix} 0 \\ 0 \\ 0.9 \\ 0 \end{bmatrix}^T,$$

consistent with a stuck actuator that reflects in the changes in matrix B_n and biased sensors which reflect in the changes in matrix C_n , respectively.

In this particular case, the occurrence of the fault leads to the loss of closed-loop stability, which makes it impossible for the IMF technique to provide a valid solution to the fault accommodation problem.

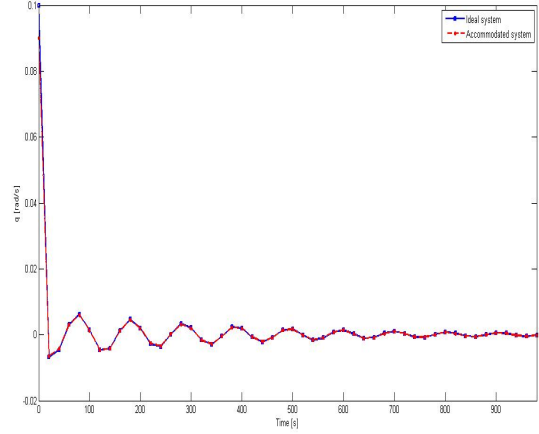


Fig. 4. Undriven Response of the Ideal/Accommodated Systems with EMF

The closed-loop system eigenvalues before controller reconfiguration are

$$\Lambda(A_f + B_f K_n C_f) = \{-0.1053 \pm 0.0652i, 0.0662, 4.1272\}.$$

However, the EMF technique can be used for controller reconfiguration even when the post-fault system model is no longer stable.

For the fault considered, let the stabilizing is

$$k_e = -0.9416,$$

computed using LQR for the same penalty matrices as in the nominal case; k_e provides the stable closed-loop system spectrum

$$\begin{aligned} \Lambda(A_f + B_f k_e C_f) \\ = \{-7.6672, -0.4235, -0.0037 \pm 0.0357i\}. \end{aligned}$$

Also, the correction gains computed using the EMF technique are

$$k_m = -0.0261$$

and

$$k_u = -0.0431.$$

Fig. 4 shows the undriven response of the accommodated system compared with that of the nominal system.

The measure of the closeness between the reference model and the accommodated system is given by the value of the norm criterion in (19) computed for the considered simulation time, which is

$$J_f^{EMF} = 0.1007.$$

To summarize, the EMF technique with output feedback can be effectively used for system reconfiguration, even when the occurrence of the fault affects the closed-loop stability.

IV. CONCLUSION

The paper analyzed the Implicit and the Explicit approaches of the Linear Model Following technique.

Guidelines are provided for the post-fault controller re-configuration so that fault accommodation is automatically achieved. Also, the various advantages and disadvantages of each method are briefly discussed. A control configuration was proposed for each technique so that the state feedback is replaced by output feedback control.

An application example from aircraft control was considered in order to demonstrate how the I/E-MF methods for output feedback could be used in fault accommodation. The simulation results show that, when a component fault does affect the aircraft behavior, the model-following methods can be effectively used in order to recompute the controller so that the accommodated system state variables follow the nominal system ones as closely as possible.

Also, according to the theoretical and to the simulation results, the controller designed using the EMF method ensures the closed-loop stability for the impaired system whenever this is stabilizable, which makes the method more attractive for practical use.

ACKNOWLEDGEMENTS

The work of Bogdan D. Ciubotaru has been co-funded by the Sectorial Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labor, Family and Social Protection through the Financial Agreement POSDRU/89/1.5/S/62557.

REFERENCES

- [1] J. Jiang, "Fault tolerant control systems - An introductory overview," *Acta Automatica Sinica*, vol. 31, no. 1, pp. 161–174, 2005.
- [2] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual Reviews in Control*, vol. 32, no. 2, pp. 229–252, 2008.
- [3] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault Tolerant Control*, 2nd ed. Springer, 2006.
- [4] J. Lunze and J. Richter, "Reconfigurable fault-tolerant control: A tutorial introduction," *European Journal of Control*, vol. 5, pp. 359–386, 2008.
- [5] E. Cimpoeșu, B. Ciubotaru, and D. Popescu, "Application example of model-following in aircraft fault tolerant control," in *Proceedings of the 18th International Conference on Control Systems and Computer Science*, Bucharest, Romania, 2011, pp. 41–46.
- [6] J. Tyler Jr., "The characteristics of model-following systems as synthesized by optimal control," *IEEE Transactions on Automatic Control*, vol. 9, no. 4, pp. 485–498, 1964.
- [7] Z. Gao and P. Antsaklis, "Reconfigurable system design via perfect model-following," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, New Orleans, Louisiana, USA, 1991, pp. 239–246.
- [8] —, "Reconfigurable system design via perfect model-following," *International Journal of Control*, vol. 56, no. 4, pp. 783–798, 1992.
- [9] H. Erzberger, "Analysis and design of model following control systems in state-space techniques," in *Proceedings of the Joint Automatic Control Conference*, Ann Arbor, Michigan, USA, 1967, pp. 572–581.
- [10] B. Etkin and L. Reid, *Dynamics of Flight: Stability and Control*, 3rd ed. John Wiley & Sons, 1996.

Increasing Vector Processor Pipeline Efficiency with a Thread-Interleaved Controller

Valeriu Codreanu, Lucian Petrică, Radu Hobincu

Abstract—Vector processors are a fast and energy-efficient way of executing code with large amounts of data parallelism. Programmability has however been a difficult topic and has limited vector processors to a few niche applications. Recent advances in compiler auto-vectorization promise to make vector processors relevant for general-purpose computing. However, compiler-generated code is inefficient and makes poor use of vector resources, which are often the most area and power-consuming devices within a processing system. We propose adapting interleaved multi-threading, a proven technique for increasing pipeline efficiency of scalar processors, to improve utilization of vector resources, thereby providing gains in speed as well as potential reduction in energy consumption.

I. INTRODUCTION

Vector processors have seen extensive research as a means of high performance parallel computing. A typical vector processor architecture consists of a scalar processor feeding instructions to a SIMD (Single Instruction Multiple Data) engine. All branches, as well as all scalar code, are executed on the scalar processor, while vector operations are passed to the SIMD array for execution [1]. This approach has the advantage of simplicity and area efficiency since fetch and branch logic is centralized and does not increase with SIMD engine size. As a result, vector processors scale well to hundreds or thousands of cores and are more energy-efficient for certain workloads than MIMD (Multiple Instructions, Multiple Data) architectures [2]. With the recent advances in auto-vectorizing compilers and optimization tools [3], software can more easily be written for, or converted to vector implementations, with potentially significant increase in performance. However, the inherent advantages of vector processors are often wasted by inefficient compilers or simply scalar code intermixed with vector code. Such control bubbles are inevitable in a vector processor and reduce pipeline utilization in the SIMD engine, lengthening computation and wasting energy.

Data locality is another issue that limits vector processing efficiency. In order for the required operands to be loaded

into vector registers in the correct form required by the algorithm, time consuming IO operations are required which, in most cases, stall the pipeline and reduce the overall efficiency. Similar delays appear when inter-cell communication is required, for example when copying data left or right in the same vector register [4].

We propose using a thread-interleaved scalar processor to alleviate the issues of low SIMD engine utilization. Thread-interleaved processors execute instructions from different threads sequentially, thereby eliminating some of the data and control dependencies between, consecutive instructions. High-latency blocking operations like memory loads can also be masked in this mode of execution by instructions from other threads, providing higher overall pipeline efficiency, as has been shown in prior art [5][6][7].

This paper is concerned with vector processor pipeline efficiency and the gains attainable by coupling a thread-interleaved scalar processor to a SIMD engine. The outline of the paper is as follows. Section II presents our proposed architecture and detailed explanation of the execution mechanism. Section III presents the test workloads and experimental setup. Section IV presents and discusses the experimental results. Section V presents some concluding remarks while in section VI we suggest a couple of ideas for future work.

II. PROPOSED ARCHITECTURE

The proposed architecture is presented in Fig. 1. The scalar processor employs a 32-bit instruction set and supports four hardware threads. It is a custom implementation of the BEAM execution model presented in [6] as a part of the authors' PhD research. The SIMD engine, ConnexArray™, is currently being developed as a commercial product by AllSeach Inc. Both the vector and the scalar processor share most of the same instruction set, which simplifies the design and implementation of a compiler for this architecture. Instructions are fetched into separate buffers for each thread while instructions are selected and issued in a round-robin fashion. Threads which encounter blocking operations are marked as inactive until they can be unblocked, at which time they are reinserted in the round-robin queue. Issued instructions which have a vector register as source or destination are pushed onto the SIMD engine instruction queue. The SIMD engine consists of 128 processing elements (PE). Each PE can operate on 16-bit values and consists of a 16-entry register file, an ALU and a large SRAM local store of 1024 entries. A vector Load/Store Unit serves the SIMD engine globally and can perform DMA between main memory and the local store.

Manuscript received June 18, 2011. This research was supported by POSDRU projects 7713 and 61178 and was carried out with the guidance and support of professor S.D. Cotofana of the Computer Science department of TU Delft and professor Gheorghe Țefan of the DCAE department of "Politehnica" University of Bucharest.

V. Codreanu is with the "Politehnica" University of Bucharest, Iuliu Maniu 1-3, Bucharest, Romania (corresponding author to provide phone: +4 0723 295 987; e-mail: vali.codreanu@arh.pub.ro).

L. Petrică is with the "Politehnica" University of Bucharest, Iuliu Maniu 1-3, Bucharest, Romania (e-mail: lucian.petrica@arh.pub.ro).

R. Hobincu is with the "Politehnica" University of Bucharest, Iuliu Maniu 1-3, Bucharest, Romania (e-mail: radu.hobincu@arh.pub.ro).

Vector to scalar and scalar to vector conversions are supported through the use of the reduction and distribution networks respectively. A 16-bit value can be converted to a vector by replicating it 128 times, and a vector can be reduced to a scalar by performing logical OR, maximum of the selected values or summation on all or some of the vector elements. All vector instructions are executed in one clock cycle except multiply and divide (which are interpreted), reduction and load/store. Scalar instructions are also executed in one clock cycle except multiply and divide, as well as loads. Encountering any of these instructions will remove the issuing thread from the running queue until the instruction has completed.

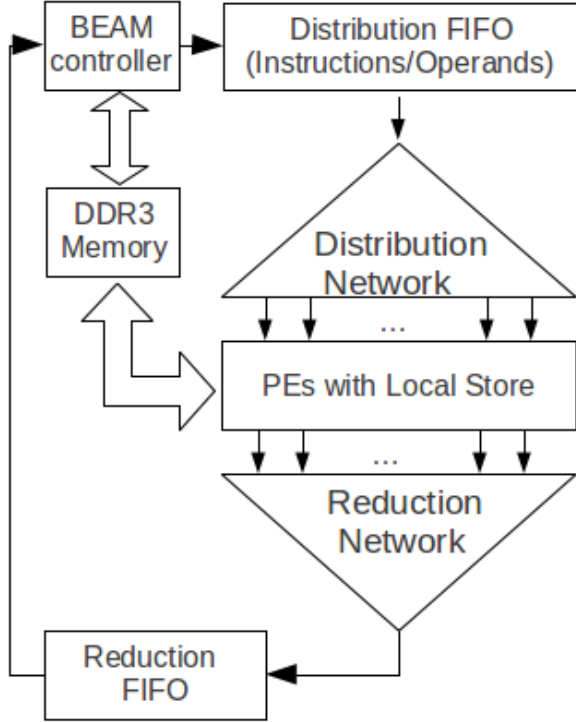


Fig. 1. Vector processor structure

For the purpose of converting the initial system to an interleaved multi-threaded scheme, a clear resource partitioning scheme must be implemented to isolate threads from each-other and provide program coherency as well as system-level security. The scalar processor has a 64 entry register file and provides each thread with 16 registers. Register windows are enforced by hardware and from a programming point of view threads are invisible to each-other. Our goal was to implement a similar partitioning scheme for vector resources while adding as little hardware as possible to SIMD processing elements. This can be achieved by restricting access from each controller thread to a subset of vector resources. In this way, each controller thread has access to 16 scalar registers, 4 vector registers and 256 entries in the local store. Fig. 2 presents the memory resources available to each thread, and data paths between resources.

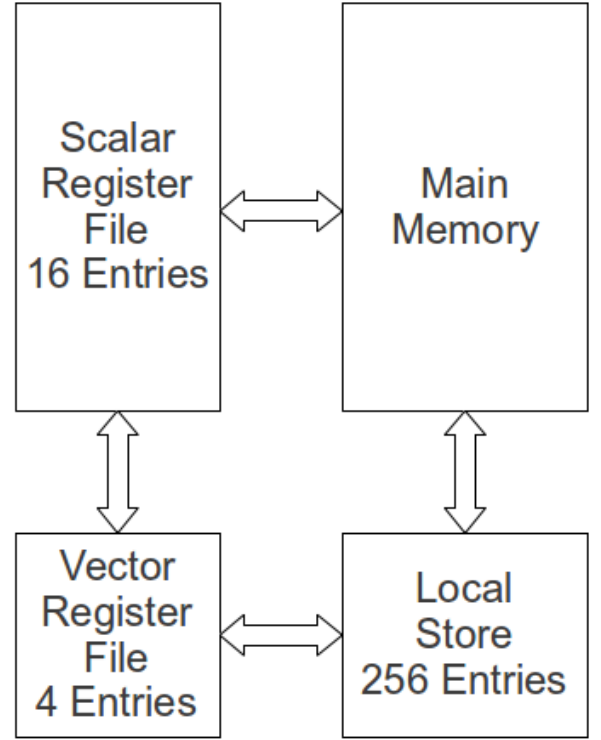


Fig. 2. Vector processor memory resources

While this partitioning allows four separate threads to issue instructions to the SIMD engine, the total vector resources available to each thread are reduced by a factor of four, which may affect per-thread performance and power consumption by increasing variable swapping between SIMD registers, the local store and main memory. These factors must be included in the overall efficiency estimates. Section III evaluates each test workload with regard to resource utilization.

III. TEST SETUP AND BENCHMARKS

We have chosen several application kernels from different application classes to test the proposed architecture under different types of workloads. All programs were written in C and compiled with a custom port of the GNU Compiler Collection (GCC). The cycle-accurate simulation was generated by an open-source tool called Verilator from the original Verilog source code.

The autocorrelation test is a signal processing kernel used in many applications including LPC (Linear Predictive Coding) voice encoding. The test consists of calculating the autocorrelation vector of 384 input sequences of 128 samples each. Samples are 16-bit fixed-point values. To simulate a LPC processing workload, only 8 values of the autocorrelation vector are computed and stored as potential input to a Levinson-Durbin algorithm implementation [8]. Profiling shows autocorrelation to be a MAC (Multiply and Accumulate) intensive kernel, where multiplication and reduction operations make up most of the vector instructions. Resource usage and IO requirements for this kernel are minimal: it only requires one entry in the local

store for input data and three registers for intermediate results. All output is done through the reduction network.

AES is the most widely-used encryption algorithm and considered to be the most secure. We have split the encryption of an AES block across 16 PEs of the SIMD engine. In addition, if CTR (Counter Mode) chaining mode is used, all the PEs can be active simultaneously to encrypt or decrypt 8 blocks in parallel. The test consists of encrypting 4 kilobytes of data in CTR mode with a 128-bit key. AES consists mostly of bitwise logical operations and as such most of the instructions are single-clock.

The FFT test consists of a batch of 64 128-point fast Fourier transforms.

The last test is the application of a Gaussian blur filter to a 300K pixel image.

For the purpose of testing the proposed architecture we have ported GCC to the new instruction set and implemented a custom assembler. Compiled and assembled code can be executed on a cycle-accurate simulator or an instruction-accurate emulator. For each test we will measure controller CPI (Clocks Per Instruction) as well as SIMD CPI and utilization (percentage of the total execution time in which the SIMD instruction queue is not empty). All tests have been set up as a pool of tasks which can be executed sequentially on a single thread or split between several threads. Instruction counts are generated by the emulator and cycle counts, as well as pipeline utilization statistics, by the simulator. The simulated memory interface has a 40-cycle latency and 3.2 GB/s of bandwidth in long sustained bursts.

IV. TEST RESULTS

Autocorrelation test results are presented in Fig. 3. They show a significant increase in SIMD engine utilization until up to three threads. Utilization levels off at 94 percent for 3 and 4 threads, as do the SIMD CPI and the controller CPI. This is because most of the test is made up of multiplication and reduction instructions. Multiplication takes 8 cycles and reduction takes 10 cycles, and are both blocking operations for the SIMD engine, while reduction is also blocking for the controller. This high density of blocking instructions sets a theoretical limit to the achievable CPI on both the controller and SIMD engine. It must be noted that for most of the tests with 3 or 4 threads the SIMD engine is fully used and total utilization is only 94% because of thread initialization overhead. Another important observation is that the 4-thread test takes more cycles to complete because the machine is already saturated performance-wise at 3 threads, and adding another thread introduces additional initialization overhead.

AES test results are presented in Fig. 4. The algorithm consists mainly of vector bitwise logical operations and does not contain any vector multiplication or reduction. About 30% of the total instruction count is scalar, and 10% are loads and stores. Because of this instruction distribution, the single-threaded AES test yields only 29% SIMD utilization. Utilization increases to 56% at 3 threads and levels off as the controller memory interface becomes saturated.

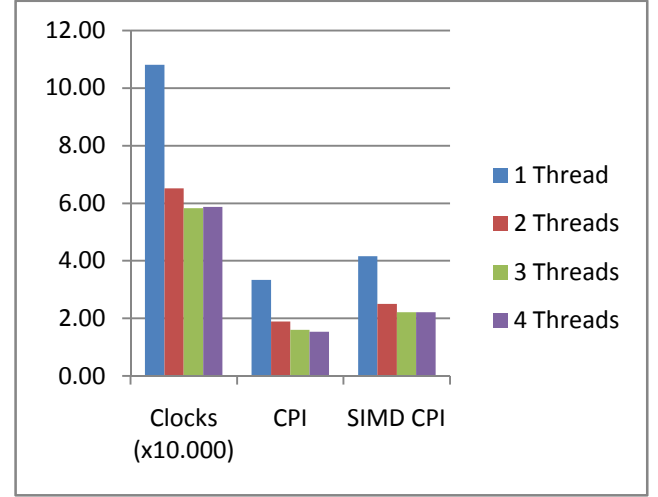


Fig. 3. Autocorrelation test results for different execution threads

The FFT implementation for the Connex system is described in [9] and the test results are presented in Fig. 5. It is similar in characteristics to AES and scales in the same way. It is able to maintain a modest 5% increase in SIMD utilization from 3 to 4 threads because it contains vector multiplications which use up more vector pipeline cycles.

Blur filter test results are presented in Fig. 6. They show no further performance increase beyond 2 threads. This is because the algorithm has a high ratio of IO to computation. Most of the time is spent waiting for vector DMA transfers between the local store and main memory, and utilization suffers accordingly.

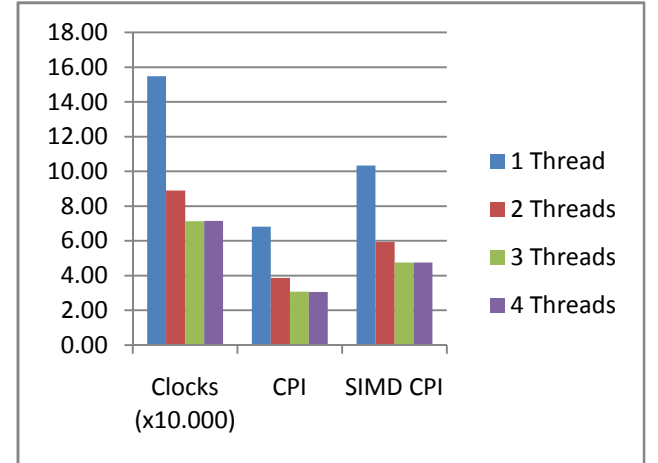


Fig. 4. AES test results for different execution threads

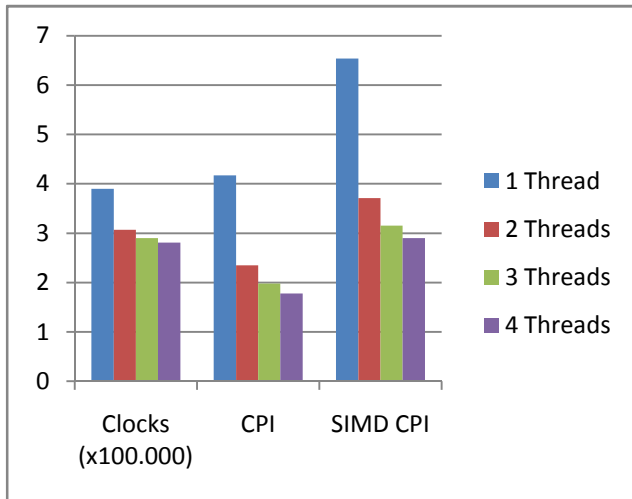


Fig. 5. FFT test results for different execution threads

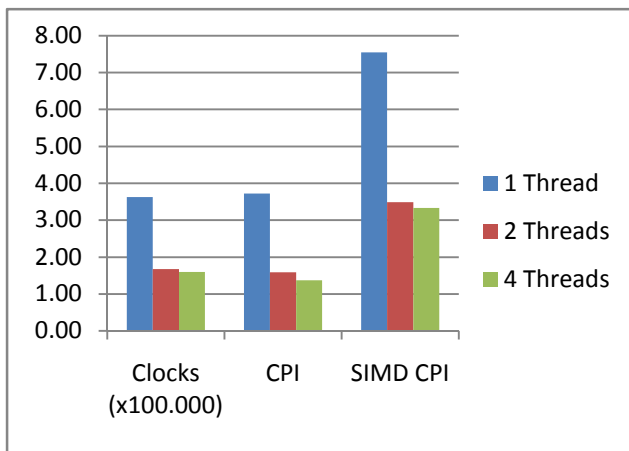


Fig. 6. Blur Filter test results for different execution threads

V. CONCLUSIONS

We have implemented a thread-interleaved controller for a vector processor. Tests on several computational kernels with different characteristics have shown significant increases in SIMD engine utilization when compared to single-threaded execution, as shown in Fig. 7. All tests show some limiting factor for performance beyond a certain number of threads. This depends on test instruction profile and IO requirements. SIMD engine utilization was at least doubled in the multi-threaded scenario compared to single-threaded execution of the same workload.

The modifications required to convert the SIMD engine to support multi-threading are minimal and easily achieved. It has also been shown that several factors can limit efficiency gains of multi-threaded execution, most notably IO DMA operations but also scalar loads and stores. These however are strongly dependent on memory interface bandwidth and latency and future work should focus on the effect of memory interface speed on multi-threaded pipeline efficiency.

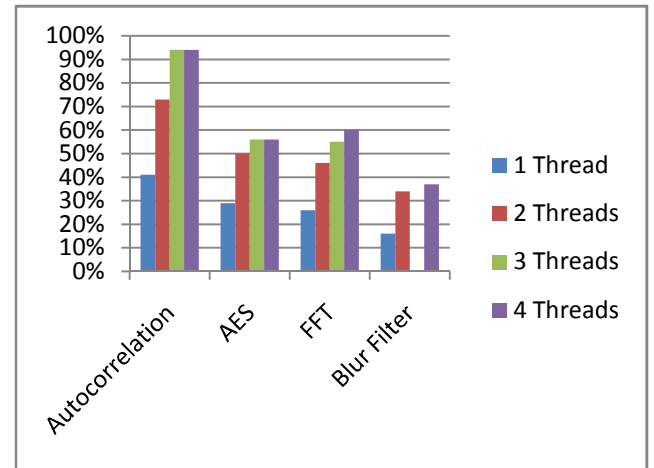


Fig. 7. Array Utilization for each kernel and different execution threads

In real life applications, the scalar interleaved CPU can also execute additional tasks if the SIMD engine utilization is saturated with less than 4 threads further increasing overall system throughput.

VI. FUTURE WORK

We are considering increasing efficiency, both in means of speed and energy consumption, by matching threads with complementary characteristics. This can be achieved by a hybrid software-hardware scheduling mechanism that takes into consideration both the machine characteristics as well as the dynamic of the algorithm.

Also, reduction operations can be masked by proper compilers by reordering instructions so that the reduction is done in background.

REFERENCES

- [1] Asanovic, K., "Vector microprocessors", PhD Thesis, University of California, 1998
- [2] Stefan, G., "The ca1024: A massively parallel processor for cost-effective HDTV", Spring Processor Forum: Power-Efficient Design, 2006
- [3] Grosser, T. and Zheng, H. and Raghes, A. and Simburger, A. and Grosslinger, A. and Pouchet, L.N., "Polly-Polyhedral optimization in LLVM", First International Workshop on Polyhedral Compilation Techniques (IMPACT'11), 2011
- [4] Shahbahrani, A. and Juurlink, B. and Borodin, D. and Vassiliadis, S., "Avoiding conversion and rearrangement overhead in SIMD architectures", International Journal of Parallel Programming, 2006
- [5] Hennessy, J.L. and Patterson, D.A., "Computer architecture: a quantitative approach, 4th Edition", Morgan Kaufmann, 2007
- [6] Valeriu, C. and Hobincu, R., "Performance Gain from Data and Control Dependency Elimination in Embedded Processors", Proceedings of ISETC, 2010
- [7] Laudon, J., "Performance/watt: the new server focus", ACM SIGARCH Computer Architecture News, vol. 33, nr 4, 2005
- [8] J. D. Markel and A. H. Gray, Linear Prediction of Speech. Berlin: Springer Verlag, 1976.
- [9] Lórentz I., Malița M. and Andonie R., "Fitting FFT onto an energy efficient massively parallel architecture", Proceedings of the Second International Forum on Next-Generation Multicore/Manycore Technologies IFMT'10

Parallel Treebank From Word-Aligned Bilingual Corpus. Language Engineering for Phrasal Alignments

Mihaela Colhon

Abstract—In this paper we describe a mechanism for parallel treebank generation between an intense studied language (i.e. English) and a less studied language, like Romanian. The Romanian constituents of the treebank are induced from the corresponding constituents of the English part taking into account the words alignments of the corpus. The proposed mechanism reuses and adjusts existing tools and algorithms for automatic Part-Of-Speech annotation and syntactic trees alignment.

I. INTRODUCTION

PROBABLY the most important trend in linguistics in the last decade is the massive use of large natural language corpora ([5]). In any Natural Language Processing system (NLP system), *corpora* is often used to provide empirical and statistical data ([6]). Typically, NLP applications that use corpora as basic linguistic resource are WSD programs and all types of parsers. Machine Translation (MT) represents the usage of computers as tools for translating texts from a source language to a target language ([20]). The vast majority of current approaches to MT systems are also corpus-based. Among these, Phrase-Based Statistical MT (PBSMT) are by far the most dominant paradigm ([17]). In this case, the linguistic resource is in the form of a aligned *parallel corpora* for the Source Language (SL) and Target Language (TL) texts of the translation.

The most common type of linguistic annotation is Part-Of-Speech (POS) tagging, which can comprise grammatical tagging and morphosyntactic annotation. A *treebank* is a corpus that has been grammatically annotated in order to identify and label different syntactic components.

The treebank generation mechanism presented in this article constructs a syntactically annotated parallel corpus automatically developed from a bilingual word-aligned corpus annotated with Part-Of-Speech (POS) data. Many studies consider that POS tags contain enough syntactical information to support word abstraction in any NLP system training. For example, the search space of a translation rules database can be greatly reduced by focusing only on POS tags instead of real words ([20]).

The mechanism for treebank generation presented in this paper is built upon the SEE-ERA.net English-Romanian parallel corpus ([18]). The corpus came along with word-

alignments, is tokenized, POS-tagged and lemmatized. The English texts are processed with one of the existed English parsers while, for Romanian texts, an induced tree generation algorithm was implemented. The algorithm for induced tree generation depends greatly on the word-alignments of the bilingual corpus as will be detailed in the following sections.

Current practice in phrase-based translation extracts regular phrases and translation rules from word-aligned parallel texts ([9]). Also, it is well-known that more and more researchers have devoted themselves to syntax-based MT systems ([8], [21], [12]). Treebanks like the one we describe here are successfully used in various NLP applications but their main scope is to enhance syntax-based translation performance of the corpora language pairs. Also, based on the alignment mechanism encoded in the treebank annotations, rich and robust set of translation rules for the corpus languages can be identified.

The method we propose here is not restricted to the languages pair we choose in order to describe it. The involved mechanism for treebank generation works on the abstract level of syntactic components and thus, all the particular information about the languages lexicon are discarded.

II. PARALLEL TREEBANK FROM BILINGUAL CORPUS

A. State of the Art

The Acquis Communautaire is the total body of European Union (EU) law applicable in the EU Member States. It is available in 22 official languages (including Romanian) of the EU and represents the biggest parallel corpus existent at this moment, taking into account both its size and the number of covered languages ([10]). A significant part of these parallel texts have been compiled by the Language Technology Group of The European Commission into an aligned parallel corpus, called *JRC-Acquis Multilingual Parallel Corpus* ([11]). In most bilingual corpora derived from JRC-Acquis corpus, we find English paired with a less studied language – called in what follows as Minority Language (shortly ML).

Also, for a parallel bilingual corpus extracted from JRC-Acquis, word-alignments could be provided (manually or with automatic tools such GIZA++ [15]) and in this case such resource can be used for an automatic treebank generation mechanism as it will be detailed in the subsequent

Manuscript received June 18, 2011.

M. Colhon is with the University of Craiova, Department of Computer Science, CODE 200585 RO (e-mail: mcolhon@inf.ucv.ro).

pages.

Treebanks are large collections of syntactically parsed sentences being considered as valuable resources not only for computational tasks such as grammar induction and automatic parsing, but for traditional linguistic and philological pursuits as well ([11]). Syntactic annotation is the practice of adding syntactic information to a text by incorporating into it markers indicating syntactic dependencies relations. In order to obtain a parallel treebank from the bilingual corpus each sentence has to be annotated with POS data. This kind of annotation is usually resulted with a POS Tagger tool.

The second step requires building annotations with syntactic phrase labels for the both parts of the corpus and then, following the word-alignments of the corpus, to align the corresponding syntactic phrases annotations.

A great progress has been done in the MT development from manually crafted linguistic models to empirically learned statistical models, from word-based models to phrase-based models and from string-based to tree-based models ([13]).

Two segments of texts from a bitext which represent reciprocal translations make a *translation unit* ([17]). A translation unit may contain, in one or both paired languages, one or more textual units (paragraph, sentence, phrase, word). In literature, there are several translation theories formalized on parallel corpora with word-level alignments. In [7] is defined a generative process by means of which a symbol tree over a target language is derived from a string of source symbols. In order to distinguish between good and bad derivations, the notion of alignment is implemented. The triples (*source_string*, *target_tree*, *word-alignment*) are depicted in special structures named *alignments graphs* from which the set of derivation rules are inferred. Many translation models pay little attention to the context and to the syntactic structures of the translated phrases. The theory of alignments, spans and crossings is discussed in [7] where the phrasal coherence across two language is studied. It is proved that incorporating syntactic information into translation models presents several advantages by the fact that syntactic phrases in one language tend to say together (i.e. cohere) during translation. Also, several studies have reported alignment or translation performance for syntactically augmented translation models.

B. The Method Description

The method described in this article expects a language with treebank generation tool as the source language and a language with POS tagging annotations for the target language. The proposed mechanism constructs a parallel treebank from a bilingual corpus only in the following scenario:

- one language of the bilingual corpus has to be an intensive studied language, based on which the syntactic

- annotation for the less studied language can be induced
- it expects a treebank format for the intensive studied language
- for the less used or minority language sub-corpus, it needs POS annotations
- the bilingual parallel corpus must be word-aligned.

English language is the best supported language, at this moment there are many large corpora, syntactic trees resources and testing language processing tools. At the opposite corner, Romanian language is a relative small language spoken by 20 million people. The language technology resources for such a small language are naturally limited. Still, there are some parallel tagged corpora available for Romanian language, mostly paired with English language.

The mechanism described in this article pairs English language with Romanian language, being built upon the Southeast European ERA-Net (SEE-ERA.net) corpus. The SEE-ERA.net corpus is part of JRC-Acquis corpus, consisting of 1.5 million words. The purpose of SEE-ERA.net project is to investigate the feasibility of machine translation research and development for several South Slavic and Balkan languages ([18]). The POS tagsets vary among different languages, but most tagset definitions can be translated from one language to another using simple translation tables. We will annotate the treebank using the PENN treebank Santorini et al. tagset ([14]). The parse trees labeled with PENN tagsets consist of words in leaves, POS tags for the preterminal nodes and phrase tags for the next levels. Each word has a corresponding POS tag. The inner nodes denote grammatical symbols (for example NP for *noun phrase*, VP for *verb phrase*, etc.). Abstraction of words in syntactic trees represents almost no informational loss from syntactical point of view.

The method we implemented can be summarized as follows: given a parse tree T_s for a source language sentence, s , and its target sentence t (that is, the translation of the source sentence in the target language) together with a word-level alignment, the parse tree of the sentence t , noted with T_t has to be induced. The algorithm for inducing target tree maps for T_t in a bottom-up fashion constituents of T_s onto contiguous substrings of t via lexical alignments.

For a lexical alignment, the most frequent alignment category is 1:1 in which one word in the source text is translated exactly to one word in the target text. However, there are other alignment categories ([3]), such as *omissions* (0:1 or 1:0), *expansions* ($n:m$, with $n < m$; $n, m \geq 1$), *contractions* ($n:m$, with $n > m$; $n, m \geq 1$) or *unions* ($n:n$, with $n > 1$).

A very popular way for visualization the parse tree of a source language sentence ending in leaf nodes – the sentence words connected by alignment links to the target sentence's words, is the *alignment graph*. An alignment link is a function $A(n) \rightarrow \{1, \dots, m\}^*$ that maps a source leaf node n of

$$\text{span}(t_s) = \text{leaf}(\text{tree})$$

where tree is a subtree of T_t , $\text{tree} \subseteq T_t$ and $n_t = \text{root}(\text{tree})$.

Algorithm 1. Target parse tree induction

Input: a bilingual corpus¹ $C=\{(s, t)\}$, a word aligner WA^2 , a source language parser³ P and target POS annotations⁴ for target sentences
Output: syntactic trees for target sentences

```

1. APPLY  $WA$  on the corpus  $C$  to obtain the word
   alignments
2. FOR each sentence pair  $(s, t)$  of  $C$ 
3.   FOR each word  $s_i$  of  $s = (s_1, \dots, s_m)$ 
4.     FOR each word  $t_j$  of  $t = (t_1, \dots, t_n)$ 
5.       IF  $s_i$  IS ALIGNED WITH  $t_j$ 
6.          $\text{aligns}(i, j) \leftarrow 1$ 
7.   PARSE sentence  $s$  with parser  $P$ 
8.   LET  $T_s$  the parse tree of  $s$ 
9.   LET  $\text{leaf}(T_s)$  the leaf nodes set of  $T_s$ 
10.   $\text{leaf}(T_s) \leftarrow \{s_i \mid 1 \leq i \leq m\}$ 
11. LET  $\text{nonT}(T_s, 1)$ : non-terminal nodes set in
     $T_s$ 
12.   $\text{nonT}(T_s, 1) \leftarrow \{\text{POS}(s) \mid s \in \text{leaf}(T_s)\}$ 
13.   $\text{nonT}(T_s, 1) \leftarrow \{\text{parent}(n) \mid$ 
     $n \in \text{nonT}(T_s, l-1), l \geq 2\}$ 
14. LET  $T_t$  the parse tree of  $t$ 
15. LET  $\text{leaf}(T_t)$  the leaf nodes set of  $T_t$ 
16.   $\text{leaf}(T_t) \leftarrow \{t_j \mid 1 \leq j \leq n\}$ 
17. LET  $\text{nonT}(T_t, 1)$ : non-terminal nodes set in
     $T_t$ 
18.   $\text{nonT}(T_t, 1) \leftarrow \{\text{POS}(n) \mid n \in \text{leaf}(T_t)\}$ 
19.   $\text{nonT}(T_t, 1) \leftarrow \emptyset, l \geq 2$ 
20. FOR each node  $s_i$  IN  $\text{leaf}(T_s)$ 
21.   $\text{span}(s_i) \leftarrow \{t_j \mid \text{aligns}(i, j) = 1\}$ 
22. FOR each node  $N$  in  $\text{nonT}(T_s, 1), l \geq 2$ 
23.   $\text{span}(N) \leftarrow \{\text{span}(s_i) \mid s_i \in \text{leaf}(T_s) \wedge$ 
     $s_i \in \text{descendant}(N)\}$ 
24. CALL  $\text{remove\_unalignedNodes}(T_s, \text{span})$ 
25. FOR each node  $t_j$  IN  $\text{leaf}(T_t)$ 
26.   $\text{parent}(t_j) \leftarrow \text{POS}(t_j)$ 
27.  $\text{level} \leftarrow 2$ 
28. WHILE  $(\text{level} \leq \text{max\_level}(T_s))$ 
29.   $\text{nonT}(T_t, \text{level}) \leftarrow \emptyset$ 
30.  FOR each node  $t_j$  IN  $\text{leaf}(T_t)$ 
31.    IF  $\exists N \in \text{nonT}(T_s, \text{level}): t_j \in \text{span}(N)$ 
32.       $\text{parent}(\text{last\_parent}(t_j)) \leftarrow N$ 
33.       $\text{nonT}(T_t, \text{level}) \leftarrow \text{nonT}(T_t, \text{level}) \cup \{N\}$ 
34.   $\text{level} \leftarrow \text{level} + 1$ 

```

The alignment of syntactic trees is the process of finding the correspondences between internal and leaf nodes of two parsing trees representing parallel sentences in different languages. For example, Prime Factorization and Alignment (PFA) algorithm assigns prime numbers to terminal nodes and spreads them to the rest of the tree from the leaf nodes towards the roots by assigning the product of child values

¹ we work with English-Romanian word-aligned SEE-ERA.net corpus

² we use the alignments provided for the SEE-ERA.net corpus

³ Stanford Natural Language Processing Group, Stanford Parser, <http://nlp.stanford.edu:8080/parser/>

⁴ the SEE-ERA.net corpus is morpho-syntactically annotated

to their fathers ([1]).

Because the target parse tree T_t is constructed taking into consideration the structure and the nodes of T_s our mechanism also include the alignments or correspondences between internal and leaf nodes of the two parallel trees, so it could be also considered as an *alignment algorithm*.

In any translation process, the one-to-zero lexical alignment is undesirable. One-to-zero means the lack of equivalent lexical translation in the target language. This phenomena is called “*lexical hole*”. This is the reason we choose to remove all the nodes from the source part that are not aligned with nodes from the target part before to determine the non-terminal nodes for the target tree (step 24 of Algorithm 1).

The time complexity of our algorithm is $O(n^{\text{max_level}(T_s)})$ because of the while loop that includes a linear browsing of the leaf nodes from T_t .

Algorithm 1 assumes that all the spans of the non-terminal nodes of T_s , are continuous lists of nodes and there are no crossing alignments. These issues will be discussed in the next section.

C. Crossing alignments

In any translation process, lexical mapping is inevitable. Crossing lexical alignments between a source sentence and a target sentence happen when the order between the source words and their translations is not preserved.

In Figure 2 there is one crossing among the word-alignments links, indicating one instance of reversing syntactic constituents during translation process. This particular crossing involves reversal of the prepositional word and the noun phrase. Depending on how often this reversal is encountered, in a translation process we could consider to invert all the TO constituents that appear before NP constituents.

The crossing alignments can be easily identified using the aligns matrix constructed by the previous algorithm.

Algorithm 2. Crossing alignments detect

Input: alignment matrix: $\text{aligns}_{m \times n}$, for m and n the words number of the two parallel sequences: $s=(s_1, \dots, s_m)$, $t=(t_1, \dots, t_n)$
Output: pairs of crossing alignments $\{((i_1, j_1), (i_2, j_2)) \mid 1 \leq i_1, i_2 \leq m, 1 \leq j_1, j_2 \leq n\}$

```

cross_align  $\leftarrow \emptyset$ 
FOR  $i = 1$  TO  $m$  DO
  FOR  $j = 1$  TO  $n$  DO
    IF  $(\text{aligns}(i, j) = 1)$ 
      IF  $(\exists \text{aligns}(i_{l+1} \leq i_1 \leq m, j_{l+1} \leq j_1 \leq j-1) = 1)$ 
        APPEND  $\{((i, j), (i_1, j_1))\}$  TO  $\text{cross\_align}$ 

```

Usually, the crossing alignment problem implies reorder the source tree such that the lexical order of the leaf nodes matches the order of the target sentence.

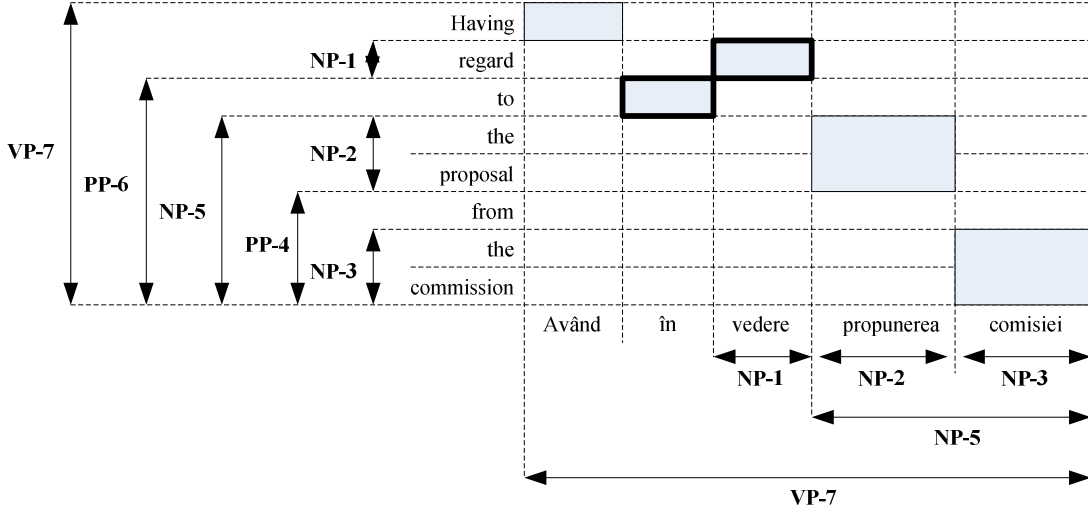


Fig. 2 The alignment matrix (the crossing alignments are marked with a thick border)

For this version of target tree induction algorithm, we choose to avoid crossing alignments, by eliminating that alignment with most crossing alignments and smallest syntactical significance. Thus, if we consider:

$$((i1, j1), (i2, j2)) \in cross_aln$$

the alignment $(i, j) \in \{(i1, j1), (i2, j2)\}$ that fulfills most of the following conditions will be removed:

- **C1.** determines most crossing alignments in the set $cross_aln$;
- **C2.** the depth of the node corresponding to the word with i -th index is smallest then the depth of the paired source word
- **C3.** the POS of the word with index i is “smallest” from the syntactic significance point of view than the POS of the paired word from the source part

As an exemplification, we consider the alignments depicted in the alignment matrix of Fig. 2. Because we have crossing alignments corresponding to *(regard, vedere)* and *(to, în)*, we will not take into consideration the alignment between the English preposition *to* and its corresponding translation in the Romanian part, the preposition *în* (consequence of the condition **C3**). This is the reason why the PP-6 phrase constituent from the English sentence cannot be induced for the Romanian sentence.

D. Parallel Treebank annotations

After the parse trees are constructed we have to flatten the hierarchical language representations for the parse trees into linear string representations, which can be easily input to many feature-like probabilistic models. Thus, during model-training, these string representations together with the alignment information can generate statistics needed to build translation grammars. Our final goal is to extract rich and robust set of English-Romanian translation rules.

In line with the Penn parse tree format used by the Stanford Parser we propose a format in which the aligned

phrase tags for the inner nodes of the trees are indexed by the same number. The common notation for the phrase nodes accompanied by the lexical alignments for the leaf nodes make easier to find the alignments between the parallel parse trees.

For the parallel English-Romanian sentences of Fig. 2, the treebank representations of the corresponding parse trees are as follows:

- for the English sentence “*Having regard to the proposal from the commission*”

```
(VP7 (VBG having)
  (NP1 (NN regard))
  (PP6 (TO to)
    (NP5
      (NP2 (DT the) (NN proposal))
      (PP4 (IN from)
        (NP3 (DT the)
          (NN commission))))
    )))
```
- for the Romanian sentence “*Având în vedere propunerea comisiei*”

```
(VP7 (VBG Având)
  (NP1 (NN vedere))
  (NP5 (NP2 (NN propunerea))
    (NP3 (NN comisiei))))
```

III. CONCLUSION

The proposed mechanism provides a way to obtain syntactic representations for a language without many parsing tools (like Romanian) by reusing tools of an intense studied language (English) to which word-alignments can be provided. It is well-known that the lexical alignments influence greatly the alignment of internal nodes in the two parallel syntactic trees. From our description it can be easily observed that wrong or incomplete alignments can affect greatly the induced target tree.

Still, the method has to be improved in order to deal with crossing alignments which, at this moment, are eliminated. We consider to take into consideration re-ordering of the

source sentence words by using a language model over Part-Of-Speech source tags ([4]).

It is important to say that although the treebank generation mechanism presented in this paper was carried out on a specific language pair, that is English and Romanian, it is so far language independent.

ACKNOWLEDGMENT

The author M. Colhon has been funded for this research by the strategic grant POSDRU/89/1.5/S/61968, Project ID 61986 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007-2013.

REFERENCES

- [1] J.G. Araújo and H.M. Caseli, "Alignment of Portuguese-English syntactic trees using part-of-speech filters", in: *Workshop on Natural Language Processing and web-based technologies (IBERAMIA-2010)*, Bahía Blanca, pp. 1-10, 2010.
- [2] D. Bamman, M. Passarotti, G. Crane, and S. Raynaud, "Guidelines for the Syntactic Annotation of Latin Treebanks" (v. 1.3), Technical report, Tufts Digital Library, Medford, 2007, Available: <http://nlp.perseus.tufts.edu/syntax/treebank/1.3/docs/guidelines.pdf>
- [3] H. de Medeiros Caseli, A. M. de Paz Silva and M. das Graças Volpe Nunes, "Evaluation of Methods for Sentence and Lexical Alignment of Brazilian, Portuguese and English Parallel Texts", in: *Brazilian Symposium on Artificial Intelligence - SBIA*, pp. 184-193, 2004
- [4] A. Ceașu, "Rich morpho-syntactic description for factored machine translation with highly inflected languages as target", in: *Workshop on Machine Translation and Morphologically-rich languages*, University of Haifa, January, 2011
- [5] J. Cuřín, M. Čmejrek, J. Havelka, V. Kuboň, "Building a Parallel Bilingual Syntactically Annotated Corpus", in K.-Y. Su et al. (Eds.): *IJCNLP 2004*, LNAI 3248, pp. 168–176, 2005.
- [6] R. Edqvist, "Developing a Core Lexicon for a Corpus-based Machine Translation System", Master's thesis in Computational Linguistics, Uppsala University, Department of Linguistics and Philology, January 2005.
- [7] H. J. Fox. 2002, "Phrasal cohesion and statistical machine translation", in *Proceedings of EMNLP-02*, pages 304–311, 2002
- [8] M. Galley, M. Hopkins, K. Knight and D. Marcu, "What's in a translation rule?," in *Proceedings of HLT-NAACL 2004*, Publisher: *Association for Computational Linguistics*, Boston, USA, pp. 273-280, 2004.
- [9] M. Galley, J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang, I. Thayer, "Scalable Inference and Training of Context-Rich Syntactic Translation Models", in: *ACL*, pp. 961–968, 2006.
- [10] A. de Gispert, J. Pino, W. Byrne, "Hierarchical Phrase-based Translation Grammars Extracted from Alignment Posterior Probabilities", in *Proceedings of EMNLP'2010*, pp.545-554, 2010
- [11] E. Irimia, "EBMT Experiments for the English-Romanian Language Pair", in *Recent Advances in Intelligent Information Systems*, ISBN 978-83-60434-59-8, pp. 91-102
- [12] JRC-Acquis, Available: <http://langtech.jrc.it/JRC-Acquis.html>
- [13] Y. Liu, Y. Huang, Q. Liu, S. Lin, "Forest-to-string statistical translation rules", in: *ACL*, pp. 704–711, 2007.
- [14] M. P. Marcus, B. Santorini and M. A. Marcinkiewicz, "Building a Large Annotated Corpus of English: The Penn Treebank", in *COMPUTATIONAL LINGUISTICS*, vol. 19(2), pp. 313-330, 1993
- [15] S. De Neefe and K. Knight, "Synchronous Tree Adjoining Machine Translation", in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing EMNLP 2009* vol 2, 2009.
- [16] F.J. Och, and H. Ney, A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics* 29 (2003), pp. 19-51, 2003
- [17] J. Tinsley, M. Hearne and A. Way, "Exploiting Parallel Treebanks to Improve Phrase-Based Statistical Machine Translation", in K. De Smedt, J. Hajič and S. Kübler (Eds.), *Proceedings of the Sixth International Workshop on Treebanks and Linguistic Theories*, 2007.
- [18] D. Tufiş, R. Ion, "Parallel Corpora, Alignment Technologies and Further Prospects in Multilingual Resources and Technology Infrastructure", in C. Burileanu and H-N Teodorescu (Eds.), *Proceedings of the 4th Conference on Speech Technology and Human-Computer Dialogue, SpeD 2007*, Iași, Romania, May 10–12 2007
- [19] D. Tufiş, S. Koeva, T. Erjavec, M. Gavrilidou, and C. Krstev, "Building Language Resources and Translation Models for Machine Translation focused on South Slavic and Balkan Languages", in M. Tadić, M. Dimitrova-Vulchanova and S. Koeva (eds.) *Proceedings of the Sixth International Conference Formal Approaches to South Slavic and Balkan Languages (FASSBL 2008)*, pp. 145-152, Dubrovnik, Croatia, 2008. ISBN 978-953-55375-0-2.
- [20] M. Volk, J. Lundborg, and M. Mettler, "Alignment tools for parallel treebanks", in *Proc. of The Linguistic Annotation Workshop at the Association for Computational Linguistics (LAW-ACL)*, 2007
- [21] J. Vičič, and A. Broddnik, "Parse Tree Based Machine Translation for Less-used Languages", Available: <http://mrvar.fdv.uni-lj.si/pub/mz/m25.1/abst/vicic.htm>
- [22] G. Maillette de Buy Wenniger, M. Khalilov and K. Sima'an, "A Toolkit for Visualizing the Coherence of Tree-based Reordering with Word-Alignments", in *The Prague Bulletin of Mathematical Linguistics* no. 94, 2010, pp. 97–106. doi: 10.2478/v10108-010-0024-4.
- [23] D. Zhang, M. Li, C.-h. Li, M. Zhou, "Phrase reordering model integrating syntactic knowledge for SMT", in: *EMNLP/CoNLL*, pp. 533–540, 2007.

Parallel Tools and Techniques for Biological Cells Modelling

Salvatore Cuomo, Pasquale De Michele and Marta Chinnici

Abstract—In this paper we show a way to use high performance computing techniques in order to achieve a more suitable neural network simulation approach. Several mathematical models and softwares for biological cells modelling are described. The performance of a modified and optimized biological neuron computational model, based on multi-thread and message passing tools, are reported.

I. INTRODUCTION

The human central nervous system involves complex large-scale interconnected neural networks, with the brain serving as the central element of this network system. The brain contains approximately 15-33 billion neurons, each of which is linked with up to 10^4 synaptic connections. Advances in neuroscience have been closely linked to the use of mathematical models, which are able to exhibit a high degree of similarity with the qualitative behavior of the system under consideration. A mathematical model is as much complex as it is close to the biological behaviors, in this context the computational neuroscience community has developed several softwares that allow to simulate many models. For each mathematical model a lot of computational models exist and are stored in several annotated databases (e.g.: BioModels¹ or ModelDB²).

The main goal is to build biological neural networks and it is a complex task. In fact, we observe that doing a search on known model repositories by using the key “biological neural network”, the number of returned research paper items is significantly lower than the search for the string “single neuron”. For example, on ModelDB repository, if we search single neuron models with the key “CA1³”, we obtain as result more than 20 computational models, while we have 2 CA1 models of neural network.

Prototyping and developing computational codes of biological neurons, in terms of reliability, efficient, and portable building blocks allow computational neuroscientists to simulate real cerebral behaviors and to validate theories and experiments. The simulations of mathematical models with single cells or neural networks are carried out by means of high performance computing methodologies. A pivotal role is played by the combined use of high performance computing

techniques, including parallel scientific computing, multi-cores programming, GPU programming, etc. In this paper we consider the pyramidal neurons of the hippocampus CA1 region and we show a combined use of scientific computing techniques in a problem solving environment.

The paper is organized as follows: Section II presents a short review of definitions and properties of neuroscience mathematical models and softwares; Section III handles the parallel tools for modelling single cells or neural networks in NEURON; Section IV is devoted to show the results related to parallelization techniques; finally, conclusions and future works are drawn in Section V.

II. BRIEF OVERVIEW OF THE MATHEMATICAL MODELS AND SOFTWARES

A mathematical model, corresponding to a particular physical system S , consists of one or more equations whose individual solutions, in response to a given input, represent a good approximation of the variables that are measured in S . A neuron can be modelled at different levels of complexity: if we consider the propagation effects, then we have compartmental models defined by means of partial differential equations (PDEs); if, instead, it is assumed that the action potential propagation is almost instantaneous compared to the time scale of the generation of itself, then we have single compartment models defined by means of ordinary differential equations (ODEs) and algebraic equations.

The first example of mathematical model has been proposed by McCulloch and Pitts (1943): by means of this very simple model it is possible to calculate the three basic logic operations NOT, AND and OR.

Integrate&Fire (I&F) is an other simple model that tries to get more closer to the biological context: in this model the neuron is considered as an electrical circuit, in which only the effects of the membrane capacitance are evaluated. The circuit is represented by the time derivative of the capacitance law ($Q = CV$), that is $I(t) = C_m(\frac{dV_m}{dt})$, where C_m is the membrane capacitance, V_m is the membrane potential and $I(t)$ is the ionic current. Unfortunately this is not realistic in physical/biological terms.

Hodgkin-Huxley (HH) (1952) [1], [2], [3] is a more complex model, in which the membrane is modelled considering the ionic currents that cross the “active” channels (i.e. those which are subject to the opening/closing) and the loss effects due to the “passive” channels (called *pores*). Consequently, the current flow that crosses the membrane has two principal components: the first one relating to the membrane capacity loading (I_{ext}) and the second one relating to the passage of specific types of ions across the membrane (I_{ion}). The ionic

S. Cuomo and P. De Michele are with the Department of Mathematics and Applications “Renato Caccioppoli”, University of Naples “Federico II”, Monte S. Angelo, Via Cintia - 80126 Napoli, Italy {salvatore.cuomo, pasquale.demichale}@unina.it

M. Chinnici is with ENEA-UTICT-PRA, Casaccia Research Center, S. Maria di Galeria (Roma), Italy martachinnici@gmail.com

¹<http://www.ebi.ac.uk/biomodels-main>

²<http://senselab.med.yale.edu/modeldb>

³Pyramidal neurons of the hippocampus CA1 region

current is also further divided in three distinct components: a sodium current I_{Na} , a potassium current I_K , and a small leakage current I_L , conveyed primarily by the chloride ions (in fact, in literature it is also shown with I_{Cl}). The electrical circuit can be described by means of the differential equation $C_m \frac{dV_m}{dt} + I_{ion} = I_{ext}$, where C_m is the membrane capacitance, and V_m is the membrane potential. The total ionic current $I_{ion} = \sum_k I_k = \sum_k G_k (V_m - E_k)$, represents the algebraic sum of the individual contributes relating to the types of the considered ions. Each ionic component I_k , therefore, has an associated conductance value G_k (the conductance is the reciprocal of the resistance, $G_k = \frac{1}{R_k}$) and an equilibrium potential E_k (the equilibrium potential that allows the flow of ions across the membrane is zero). The system of equations that describes the HH model is the following:

$$\begin{cases} C \frac{dV}{dt} = -\overline{g_{Na}} m^3 h (V_m - E_{Na}) + \\ \quad -\overline{g_K} n^4 (V_m - E_K) + \\ \quad -G_L (V_m - E_L) + I_{ext} \\ \frac{dm}{dt} = \alpha_m (1 - m) - \beta_m m \\ \frac{dh}{dt} = \alpha_h (1 - h) - \beta_h h \\ \frac{dn}{dt} = \alpha_n (1 - n) - \beta_n n \end{cases} \quad (1)$$

The values of the constants $\alpha_m, \alpha_h, \alpha_n, \beta_m, \beta_h, \beta_n$, of the maximal conductance and of the equilibrium potential have been obtained experimentally by Hodgkin and Huxley [1]. It is possible to obtain a modified version of the HH model, which describes the dynamics of the membrane potential of the biological neurons. By means of the “cable-theory” [4] it is possible to introduce the potential dependence as a function of the space (as well as of the time), integrating in the first equation of the system (1) the so-called “cable equation”, that is a second degree PDE:

$$\lambda^2 \frac{\partial^2 V(x, t)}{\partial x^2} = \tau \frac{\partial V(x, t)}{\partial t} + V(x, t)$$

In our implementations, starting from a modified HH model to simulate experiments on the hippocampal pyramidal neurons of CA1 region, we observe the effects of CREB-mediated gene expression on the excitability of these [5], [6].

There are many softwares that allow to simulate the neuronal dynamics. The simulations of single neurons and neural networks are typically based on the approach of compartmental modelling. Each cell is divided into many isopotential compartments, and each compartment is modelled with equations describing electrical currents [7]. In this way, there are no restrictions on the neuronal membrane properties described. In Tab. I we report ([8], [2], [9], [10], [7]) the main softwares and their major features. From our knowledge, we deduce that the softwares are divided into two main categories: general and special purpose. We observe that the widely diffused softwares are NEURON [8] and GENESIS [2]. These frameworks have large communities where the users collect and maintain databases of published computational models. In our work we choose NEURON,

TABLE I
FEATURES OF SEVERAL COMPARTMENTAL MODELLING PACKAGES

Question	NEURON	GENESIS	Nodus	SNNAP	Surf-Hippo	NeuronC	HHSim
Parallel	MPI	MPI, PVM	NO	NO	NO	NO	NO
Impl. lang.	C	C	Fortran	Java	Lisp	C	Matlab
Progr. lang.	Hoc, NMODL	C, Genesis scripting	NO	NO	NO	NO	NO
OS	Unix, Win, MacOS	Unix	MacOs	any	Unix	Unix	Unix, Win, MacOS
GUI	YES	YES	YES	YES	YES	NO	YES
Interpreter	YES	YES	NO	NO	NO	YES	NO
Mod. def.	Interpreter	Interpreter	Editor	Editor	Files	Interpreter	GUI
Ionic ch.	YES	YES	YES	YES	YES	YES	HH
Syn. ch.	YES	YES	YES	YES	YES	YES	NO
Neural net.	YES	YES	Limited	YES	YES	YES	NO
Int. method	Impl.	Expl., Impl.	Expl.	Expl.	Impl.	Expl., Impl.	No info
Web site	YES	YES	YES	YES	YES	YES	YES
User group	SenseLab	BABEL	NO	NO	NO	NO	NO
Book	YES	YES	NO	NO	NO	NO	NO

The list of questions, where: **Parallel**: does it support parallel processing? **Impl. lang.**: was it implemented with which language? **Progr. lang.**: has it a programming language? **OS**: does it run on what OS? **GUI**: has it a GUI? **Interpreter**: has it an interpreter? **Mod. def.**: by means of what is defined the model? **Ionic ch.**: does it manage the ionic channels? **Syn. ch.**: does it manage the synaptic channels? **Neural net.**: does it manage the neural networks models? **Int. method**: what integration method(s) has it? (answers: *Impl*: implicit; *Expl*: explicit) **Web site**: has it a web site? **User group**: has it a users group? **Book**: are there published books about the simulator?

that can solve problems at several levels of detail. This software is very flexible and can provide many advantages: it enables the users to add new elements to the model without to rewrite all the existing code and to re-use previously developed elements in other computational models [7]. In particular, NEURON is an environment for creation and use of empirically-based models of biological single neurons and neural networks [11]. It was developed for simulation of the “nerve equations with cable geometry”, or complex ionic channels. Moreover, unlike the other softwares mentioned in Tab. I, NEURON does not deal directly with compartments, but it is designed around the notion of continuous cable “sections” which are connected together to form any kind of branched tree structure [8]. Each section is divided into a number of segments of equal length: this number is denoted by the section parameter `nseg` [8]. The NEURON’s GUI contains several tools which can be used to construct models, exercise simulations, and analyze results. Moreover, users can define density mechanisms and point processes for each individual model, which can be linked into NEURON by using the model description language NMODL. One of NEURON’s advantages is that it enables the user to specify a 3D-topology, which is useful when the model is based on the anatomical reconstruction of data. NEURON provides a choice between a first-order fully implicit integration method (backward Euler) and a more accurate second-order variant of the Crank-Nicholson time step [8], with the improvements in computation speed for simulations of arbitrarily branched cables with HH kinetics.

III. PARALLEL TOOLS FOR CELLS MODELLING IN NEURON

High performance computing tools are very interesting in cell modelling. For example, in the case of single cell “CA1 model” (Poirazi et al.) [12], applying an external electrical stimulus, for a period of $1500ms$, with 200 artificial synapses, requires an average execution time of $T_{stim} \approx 131s$, on a desktop PC (“Intel P8800” 2.66GHz, dual-core). The complete execution of an experiment requires the application of 25 different stimuli of external current, then the average time for the simulation of a single experiment is $T_{exp} \approx 25 * 131s \approx 3275s$. It is needed to repeat the experiment at least 10 times, by placing the synapses in a random way, to validate the computational results. The full time for this specific example is given by $T_{tot} \approx 10 * 3275s \approx 9h$.

Our goal is to design a reliable scheme of biological neural networks. This can be achieved developing computational codes including message passing tools (MPI, PVM, etc.) and multi-cores programming (OpenMP, P-threads, CUDA, multi-threading, etc). We adopt a distribution scheme of neural network on the computing environment NEURON. The parallelization is made by modifying procedures and developing new functions in a well known NEURON model published on ModelDB [12]. We implement a computational strategy based on two connected layers. In the **layer 0** we distribute the model of an individual neuron by splitting the cell compartments on multi-cores architectures; in the **layer 1** we distribute the network model, where each cell is assigned to one host, in a suitable topology, that depends on biological cell interconnection. In other words, we implement a parallel scheme (Fig. 1). Our idea is to set up a neural network with M neurons assigned to M hosts, and for each neuron, assigned to a host, the sequential code sections are suitable splitted on the N -cores architecture, taking into account the load balancing.

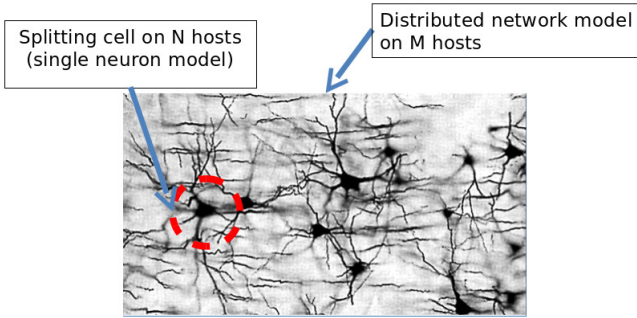


Fig. 1. Parallel techniques combination.

We implemented the “layer 0” by using a NEURON’s technique called “multisplit” [13]. This technique splits the cell into the minimum number of pieces required for load balance. It leads several advantages: it is not necessary to change the code that defines the cell type (that is the template); it allows you to exploit, in the best possible way, the available computing resources, obtaining both a load

balance, otherwise impossible, and a significant decrease of the execution time. Multisplit needs to use the NEURON’s class `ParallelNetManager` that manages the setting and running of a network simulation on a cluster of workstations or any parallel computer that has MPI installed. This class contains a collection of methods that can be used to manage the splitted cell. We have adapted the basic functions of multisplit introducing necessary changes to its features. The following `proc batchrun()` is an example of procedure that uses the `ParallelNetManager` class, in the multisplit technique.

```
proc batchrun() { local i, sum, sum_2
...
for i=0, n_simul-1 {
  {pc.multisplit()}
  {pc.set_maxstep(100)}
  ...
  statrun(tstop)
  ...
  if (section_exists("soma", 0))
    potential_hh_syn_print3($1,$s2,i+1,$4)
  ...}pc.done() }
```

Remarks: `pc` is an instance of the `ParallelNetManager` class. The NEURON’s function `section_exists()` is used in order to check the access to various cell sections. This returns 1 if the section exists and can be used, otherwise it returns 0.

Increasing the performance in the “layer 0” is the first step in order to construct a biological neural network and it depends on load balancing, as showed in section IV. We develop the “layer 1” changing the serial version of a neural network with sequential topology into a parallel version. The parallelization approach chosen for network model consists in distributing all cells between the different hosts available. The mechanism of allocation, that is implemented in the `proc mkcells()`, is the following one: the first cell is allocated to the first host, and the cell `nh+1`, if it exists, is allocated to the first host as well (where `nh` is the total number of hosts), and so on; same procedure for the remaining cells/hosts. It is most important to underline that the first step in order to implement a parallel version of a neural network in NEURON is to create an instance of the `ParallelContext` class, that contains a collection of methods that can be used to manage the post-synaptic cells. The following `proc mkcells()` is an example of procedure that uses the `ParallelContext` class.

```
proc mkcells() {local i localobj cell, nc, nil
  cells = new List()
  ranlist1 = new List()
  ...
  for (i=pc.id; i < $1; i += pc.nhost) {
    cell = new Ca1()
    cells.append(cell)
    ranlist1.append(new RandomStream(i))
    ...
    pc.set_gid2node(i, pc.id)
    nc = cell.connect2target(nil)
    pc.cell(i, nc)
  } }
```


Remarks: `pc` is an instance of the `ParallelContext` class.

We observe that the loop `for` does not iterate over all the network cells, but only on those present on the specific host that invoked it. This construct is responsible for associating between each host and its group of cells. `pc.set_gid2node(i, pc.id)` associates the cell with “gid” `i` to the host (`pc`) with the identifier `id`. `nc = cell.connect2target(nil)` associates the position of the “spike detector” (`nc`) to the cell: in this case the `NetCon` does not need a real target, so this is null. `pc.cell(i, nc)` associates the identifier `i` of the cell (gid) to the “spike detector” `nc`.

In the next section we report the results on the proposed scheme.

IV. NUMERICAL EXPERIMENTS

Numerical experiments are conducted on both models, the single neuron “CA1” and the neural network. We setup the single neuron model of Poirazi [12] by considering the assumptions and results from the experiments [6] (for 200, 300, 400 and 500 artificial and real synapses) where:

- the current injection steps are applied for 25 times in the center of the soma (technique “whole-cell” of the patch-clamp), for each simulation;
- the neuron resting potential at $0ms$ is considered to $-70mV$, setting `v_init=-70`;
- the number of excitatory synapses mediated by NMDA receptors (`n_NMDA`) is, on average, 20% more than the number of synapses mediated by AMPA receptors (`nsyn`), so `n_NMDA=nsyn*(1.2+cc)`.
- the number of inhibitory synapses mediated by $GABA_A$ (represented by `n_GabaA`) and $GABA_B$ (represented by `n_GabaB`) receptors is the same, so `n_GabaA=nsynI*(1+b)` and `n_GabaB=nsynI*(1+b)`.

The application of external current stimuli it is considered for a period of $1500ms$. In the neural network model we use a simplified version of the CA1 pyramidal cells. We implement a neural network composed by 20 neurons, connected in a sequential topology (Fig. 3). In this configuration we set up the resting potential at $0ms$ to $-65mV$ (`v_init=-65`) and the application of external current stimuli is considered for a period of $600ms$. All simulations are carried out with NEURON v7.1. The parallel functionality multisplit is available configuring NEURON with the `--with-paranrn` option which requires a pre-installation of MPI. Performance tests for simulations are carried out on S.Co.P.E. Grid infrastructure of University of Naples “Federico II”, which consists of 300 8-cores blade servers “Intel Xeon E5410” at 2.33GHz (2400 cores in total), connected with several kinds of link (Infiniband, Fiber Channel, 1Gb and 10Gb Ethernet); the OS is Scientific Linux (x86_64), release 2.6.18-194.32.1.el5, version #1 SMP Tue Jan 4 12:47:36 EST 2011.

The performance are evaluated in terms of *speedup* and *efficiency*. The speedup is defined as the ratio of the running time on a single processor $T(1)$ to the running time on

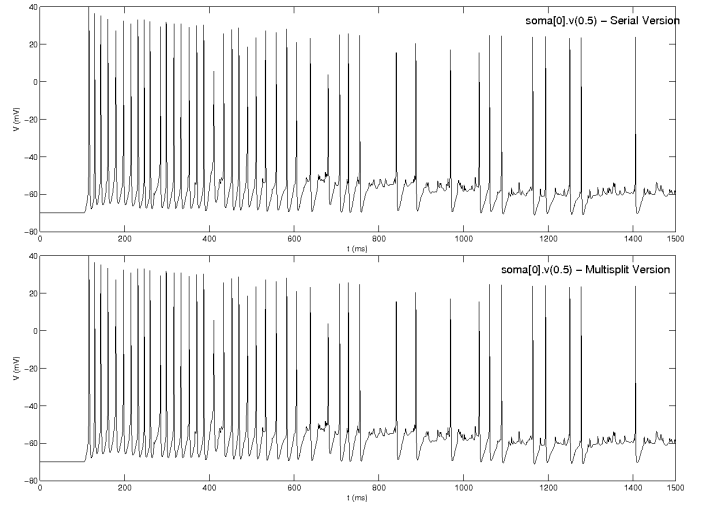


Fig. 2. Single neuron model: graphs corresponding to the cell membrane potential with 200 artificial synapses, later the application of several external current stimuli for a period of $1500ms$, in the serial (above) and parallel (below) versions.

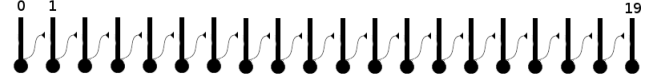


Fig. 3. Neural network composed by 20 neurons in sequential topology.

the parallel configuration $T(n)$: $S(n) = \frac{T(1)}{T(n)}$. We use the *relative speedup* definition, where $T(1)$ is the running time of the parallel program executed on a single host. The efficiency is given by $E(n) = \frac{S(n)}{n}$, and its value, typically, varies between 0 and 1 (or between 0 and 100 if it is expressed in percentage), and it is a measure of how the parallel algorithm take advantages from the available hosts.

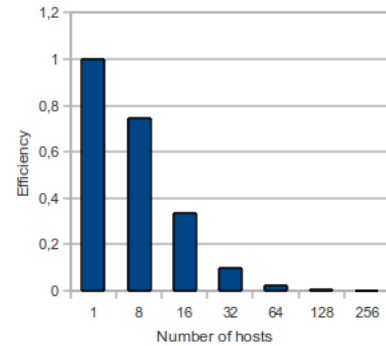


Fig. 4. Graph corresponding to the efficiency performance (TOTAL) obtained by multisplit.

The serial and parallel versions of the “CA1 model” produce identical results (Fig. 2) and different performance (Tab. II and Fig. 4). The “total running time” (TOTAL) for a multisplit execution is given by the contribution of two components, which are the “time required for setting the split of the cell” (SETUP) and the “time required to run the program” (RUN). RUN is given by the contribution of other three components, which are the “time spent by the hosts

TABLE II

PERFORMANCE IN FUNCTION OF THE NUMBER OF INVOLVED HOSTS FOR A MULTISPLIT SIMULATION WITH 200 ARTIFICIAL SYNAPSES, AFTER THE APPLICATION OF ELECTRICAL STIMULI FOR A PERIOD OF 1500ms

HOSTS	SETUP	RUN	TOTAL	WAIT	STEP
1	0	131,01	131,01	0	131,01
8	0,51	21,53	22,04	0	18,5
16	1,2	23,22	24,41	0	9,81
32	1,97	39,38	41,35	0,01	5,44
64	3,87	77,98	81,85	0,02	3,32
128	7,84	257,55	265,39	0,05	2,23
256	15,63	659,19	674,82	197,25	1,76

HOSTS	SCCT	LI	MF	S-UP	EFF
1	0	0%	0,8	1	1
8	3,03	0,8%	0,1	5,96	0,75
16	13,4	77,5%	0,4	5,38	0,34
32	33,93	255,5%	0,9	3,18	0,1
64	74,62	604,5%	0,9	1,61	0,03
128	255,22	1299,9%	0,7	0,5	0
256	459,86	2695,9%	0,9	0,19	0

All the values are expressed in seconds (s), and these refer only to a single loop of the complete execution.

Performance features: **HOSTS:** number of involved hosts; **SETUP:** average time required for setting the split of the cell; **RUN:** average time required to run the program ($RUN = WAIT + STEP + SCCT$); **TOTAL:** total average time required for setting the split of the cell and run the program ($TOTAL = SETUP + RUN$); **WAIT:** average time spent by the hosts waiting for the other hosts; **STEP:** average time required so that each host completes its part of the calculation; **SCCT (Split-Cell Communication Time):** average time spent by the hosts to communicate with each other; **LI:** load imbalance in percentage; **MF:** max factor; **S-UP:** relative speedup value; **EFF:** efficiency.

waiting for the other hosts” (WAIT), the “time required so that each host completes its part of the calculation” (STEP) and the “time spent by the hosts to communicate with each other” (SCCT, or Split-Cell Communication Time). By these positions we derive that the multisplit involves a substantial reduction of the total average running time for 8 hosts, running on the same 8-cores blade server. From 16 to 256 hosts we observe a decay of the performance by means of the “load imbalance” (LI) and the increase of the “split-cell communication Time” (SCCT), running on several blade servers. SCCT is due to the latency of the S.Co.P.E. network architecture. Moreover, we note that parameter `maxfactor` (MF), that influences how to split the cell, has an impact on the performance of the multisplit technique. We observe that the value of LI depends on MF and the “number of the involved hosts” (HOSTS).

The serial and parallel versions of the neural network with sequential topology produce identical results (Fig. 5). The “running time” (RUN) for an execution of the parallel version of the network is given by the contribution of two components: the “time spent by the hosts waiting for the other hosts” (WAIT) and the “time required so that each host completes its part of the calculation” (STEP). We observe (Tab. III) that the neural network model scales with increasing of the host number up to 32: in fact STEP does not substantially change. The unoptimized memory usage determines a slight increase of STEP values from 64 to

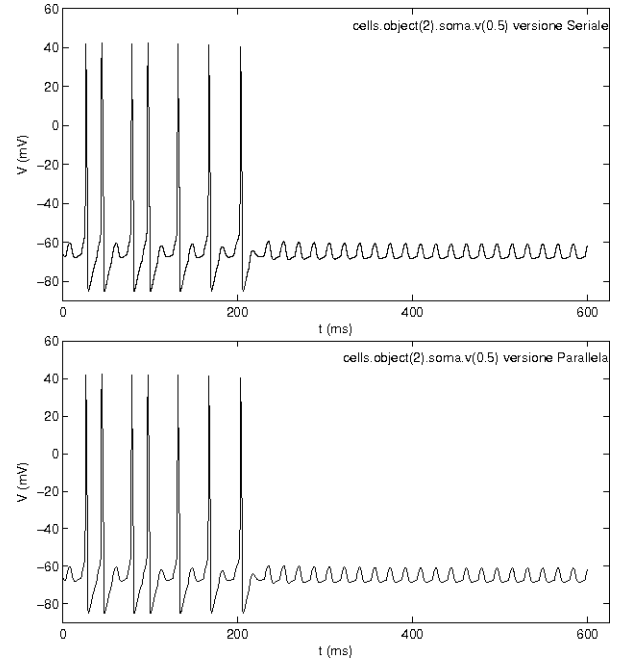


Fig. 5. Neural network model: simulation results related to a network cell in the serial (above) and parallel (below) versions, later the application of several external current stimuli on the first neuron of the network (neuron 0) for a period of 600ms.

TABLE III

SCALABILITY OF THE NEURAL NETWORK PARALLEL VERSION IN FUNCTION OF NUMBER OF CELLS AND INVOLVED HOSTS, AFTER THE APPLICATION OF ELECTRICAL STIMULI FOR A PERIOD OF 600ms

HOSTS	CELLS	RUN	WAIT	STEP
8	8	3,76	0,34	3,4
16	16	8,07	4,32	3,73
32	32	16,92	13,1	3,8
64	64	53,29	47,79	5,45
128	128	110,16	104,04	6,06
256	256	301,71	293,79	7,76

All the values are expressed in seconds (s), and these refer only to a single loop of the complete execution.

Performance features: **HOSTS:** number of involved hosts; **CELLS:** number of the network cells; **RUN:** average time required to run the program ($RUN = WAIT + STEP$); **WAIT:** average time spent by the hosts waiting for the other hosts; **STEP:** average time required so that each host completes its part of the calculation.

256 hosts. In Tab. IV and Fig. 6, we observe that the parallelization of the network involves improvements of the performance up to 128 hosts; later, the latency of the network architecture degrades the performance. This worsening is due to the network topology and to the fact that the sequential calculation fraction for each host is relatively little.

V. CONCLUSIONS AND FUTURE WORKS

The analysis and the implementation of all features of a neuronal modeling software is essential to conduct experiments that simulate the biological reality. As already mentioned, this requires large computing power. Developing tools for scientific and parallel computing to improve the per-

TABLE IV

PERFORMANCE IN FUNCTION OF 256 CELLS AND NUMBER OF INVOLVED HOSTS FOR NEURAL NETWORK PARALLEL VERSION, AFTER THE APPLICATION OF ELECTRICAL STIMULI FOR A PERIOD OF 600ms

HOSTS	RUN	WAIT	STEP	S-UP	EFF
1	6880,11	0	6879,46	1	1
8	2877,37	21,92	2855,14	2,39	0,30
16	1848,1	843,46	1004,35	3,72	0,23
32	666,31	505,4	160,77	10,33	0,32
64	286,85	254,59	32,12	23,99	0,37
128	231,19	218,45	12,64	29,76	0,23
256	301,71	293,79	7,76	22,80	0,09

All the values are expressed in seconds (s), and these refer only to a single loop of the complete execution.

Performance features: **HOSTS:** number of involved hosts; **RUN:** average time required to run the program ($RUN = WAIT + STEP$); **WAIT:** average time spent by the hosts waiting for the other hosts; **STEP:** average time required so that each host completes its part of the calculation; **S-UP:** relative speedup value; **EFF:** efficiency.

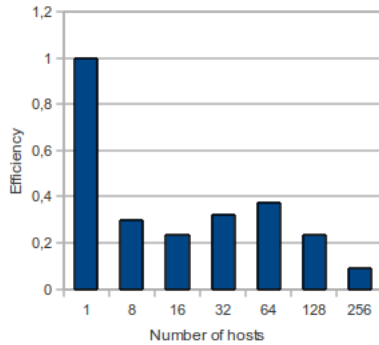


Fig. 6. Graph corresponding to the efficiency performance (RUN) obtained by parallel version of the neural network with sequential topology.

formance of computational models is a possible solution. In this work we add new elements to a NEURON model and we adapt previously developed procedures for implementing new models and improving the performance. By using softwares like NEURON, which allow the integration of computer codes in programming languages and scientific libraries, is possible to fit neuronal models to new and different computing architectures. We have a working progress for porting our computational model to the GPU architecture of the data center ENEA-CRESCO by using CUDA library.

ACKNOWLEDGMENTS

This paper is financial supported by Multicentre Research Project funded by Compagnia di San Paolo: “Molecular Mechanisms of Memory: Identification and Modeling”. The authors gratefully acknowledge the contribution of reviewers’

comments.

REFERENCES

- [1] A.L. Hodgkin and A.F. Huxley, “A quantitative description of membrane current and its application to conduction and excitation in nerve,” *J PHYSIOL*, vol. 117, pp. 500–544, 1952.
- [2] James M. Bower, David Beeman, *et al.*, *The Book of GENESIS*. Internet Edition, 2003.
- [3] Ji-Huan He *et al.*, “A modified Hodgkin-Huxley model,” *CHAOS SOLITON FRACT*, vol. 29, pp. 303–306, 2005.
- [4] W. Rall and H. Agmon-Snir, *Cable Theory for Dendritic Neurons in “Methods in Neuronal Modeling”*. Koch, C. and Segev, I., MIT Press, 1998.
- [5] M. Lopez de Armentia *et al.*, “cAMP Response Element-Binding Protein-Mediated Gene Expression Increases the Intrinsic Excitability of CA1 Pyramidal Neurons,” *J NEUROSCI*, 2007.
- [6] H. Marie *et al.*, “CREB modulates excitability of nucleus accumbens neurons,” *NAT NEUROSCI*, pp. 475–477, 2006.
- [7] O. Kroupina and R. Rojas, “A Survey of compartmental modelling packages,” Free University of Berlin, Tech. Rep. B-04-08, June 2004.
- [8] N.T. Carnevale and M.L. Hines, *The NEURON book*. Cambridge University Press, 2006.
- [9] L. Graham, “The Surf-Hippo Neuron Simulation System, v3.5,” 2004. [Online]. Available: <http://www.neurophys.biomedicale.univ-paris5.fr/~graham/surf-hippo-files/Surf-Hippo.README.html>
- [10] Robert G. Smith, “NeuronC: a computational language for investigating functional architecture of neural circuits,” *J NEUROSCI METH*, vol. 43, pp. 83–108, 1992.
- [11] Romain Brette, Michelle Rudolph, Ted Carnevale, Michael Hines, *et al.*, “Simulation of networks of spiking neurons: A review of tools and strategies,” *J COMPUT NEUROSCI*, vol. 23, pp. 349–398, 2007.
- [12] P. Poirazi, “CA1 pyramidal neuron: as a 2-layer NN and subthreshold synaptic summation,” 2003. [Online]. Available: <http://senselab.med.yale.edu/ModelDB/showmodel.asp?model=20212>
- [13] M.L. Hines, H. Markram, and F. Schürmann, “Fully Implicit Parallel Simulation of Single Neurons,” *J COMPUT NEUROSCI*, vol. 25, no. 3, pp. 439–448, 2008.

MIMO Conditional Integrator Control for a Class of Nonlinear Systems

Gilney Damm and Van Cuong Nguyen

Abstract—This work develops a Multi-Input Multi-Output (MIMO) Conditional Integrator (CI) controller to a class of MIMO nonlinear systems, in the case of asymptotically constant references, motivated by airspace applications. These results are then applied to an aircraft control case and are compared by computer simulations with previous results from the literature that represent the starting point of the current paper.

The obtained controller allows in a first step finite time semiglobal stability to a residual region, followed by exponential stability since entering this region.

I. INTRODUCTION

The control theory known as Conditional Integrator (CI) was developed in a series of papers from Khalil and co-workers ([4], [2], [9], [8] and [6]). This controller acknowledges a saturation (natural or not) on the control signal, and takes advantage on that to behave, under some conditions, as an sliding mode controller (SMC). On other conditions, when not saturated, the controller behaves as a dynamical feedback with an important integral term. This approach has some interesting features, For example well known drawbacks of integrators like performance degradation and in particular the problem of integrator wind-up are avoided by the conditional nature of such control scheme. The integral action is then only present inside a boundary given by the saturations. In this way the control scheme assures the good properties of robustness and performance of sliding modes controllers for large errors, while allowing a smooth behavior given by its continuity what avoids chattering. Furthermore, the robustness of the SMC-like nature of the system while saturated is combined with the “adaptive” characteristic of the integral term when closer to equilibrium. In this way, such technique is very interesting in the cases of poorly known systems or with large uncertainties.

More recently ([12], [8] and [6]), efforts were consecrated to extend these results for the Multi-Input Multi-Output (MIMO) case, with good results for some classes of MIMO nonlinear systems. The present work follows this line and can be seen as an extension of those. This present work was motivated by airspace applications, that have poorly known models and parameters. For these applications CI is an interesting control strategy, but unfortunately they are also an example of a class of nonlinear systems not addressed by previous results.

This paper will introduce in section II the considered class of MIMO nonlinear systems, and will present the main results that develop the CI controller for this class, mostly

based on the nonlinear theory found in [3], [5] and [1]. These results are applied in III for the control of a MIMO nonlinear aircraft model. This work should be seen in the optics of the recent papers ([10] and [12]) that have applied the conditional integrator controller to airspace, but in a SISO framework for the first and a MIMO framework for a linearized case on the second. Computer simulations are presented in section IV to illustrate the theoretical results. These simulations also compare the present results to those of [12] in order to illustrate the extension attained in the current work. The paper is then wrapped up by conclusions and perspectives in section V.

II. CONDITIONAL INTEGRATOR CONTROL DESIGN

The conditional integrator controller designed for the output regulation of a class of minimum-phase nonlinear systems in case of asymptotically constant references is studied in [6], [8] and [9]. The works of these papers concern a servo-compensator performing as a sliding mode controller outside a boundary layer, and performing as a conditional one that provides servo-compensation only inside the boundary layer; achieving asymptotic output regulation.

However, these works have addressed the case of linearized systems. The main objective of this section is to present the conditional integrator controller design for the output regulation of a class of MIMO nonlinear systems, in the case of asymptotically constant references, providing the proof of the exponential stability of the system under some assumptions.

Consider the nonlinear multi-input multi-output system in canonical form:

$$\begin{cases} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(x_1, x_2) + g(x_1, x_2)u \\ y &= x_1 \end{cases} \quad (1)$$

where $x_1 \in \mathbb{R}^n$ and $x_2 \in \mathbb{R}^n$ are the state vector, $y \in \mathbb{R}^n$ the output vector, $u \in \mathbb{R}^n$ the control input and $f(x_1, x_2) \in \mathbb{R}^n$, $g(x_1, x_2) \in \mathbb{R}^{n \times n}$ are continuous functions.

Let $y_{ref} = x_{1ref}$ be a prescribed reference output function considered as constant such that their derivatives are null.

Define the tracking error vector $e_1(t)$ as the difference between the actual system output $y(t) = x_1(t)$ and the reference output: $e_1 = x_1 - x_{1ref}$, $e_2 = \dot{e}_1 = x_2 - \dot{x}_{1ref} = x_2$ such that (1) can be rewritten as¹:

$$\begin{cases} \dot{e}_1 &= e_2 \\ \dot{e}_2 &= f(e_1, e_2) + g(e_1, e_2)u \end{cases} \quad (2)$$

¹For a simpler notation, we skip x_{ref} , $\dot{x}_{ref}$ and $\ddot{x}_{ref}$ inside $f(\cdot)$ and $g(\cdot)$.

Gilney Damm is with IBISC - Université d'Evry Val d'Essonne, Evry, France gilney.damm@ibisc.fr

Van Cuong Nguyen is with IBISC - Université d'Evry Val d'Essonne, Evry, France vancuong.nguyen@ibisc.fr

Let us impose the sliding surface as:

$$s = k_0\sigma + K_1e_1 + e_2 \quad (3)$$

where $\sigma \in \mathbb{R}^n$ is the output of the conditional servo-compensator

$$\dot{\sigma} = -k_0\sigma + \mu \text{sat}(s/\mu) \quad (4)$$

in which μ is the boundary layer, k_0 is a positive parameter, $K_1 \in \mathbb{R}^{n \times n}$ is chosen such a way that $K_1 + sI_n$ is Hurwitz., I_n is the $n \times n$ identity matrix. The saturation function is determined as:

$$\text{sat}(s/\mu) = \begin{cases} s/\|s\| & \text{if } \|s\| \geq \mu \\ s/\mu & \text{if } \|s\| < \mu \end{cases} \quad (5)$$

The derivative of the sliding surface can be expressed as:

$$\dot{s} = k_0\dot{\sigma} + K_1\dot{e}_1 + \dot{e}_2 \quad (6)$$

Equation (6) may be written again from (2) and (3)

$$\begin{aligned} \dot{s} &= k_0(-k_0\sigma + \mu \text{sat}(s/\mu)) + K_1\dot{e}_1 + \dot{e}_2 \\ &= -k_0s + k_0\mu \text{sat}(s/\mu) + K_1\dot{e}_1 + \dot{e}_2 + k_0(K_1e_1 + e_2) \end{aligned} \quad (7)$$

Now by letting

$$\Delta(e_1, e_2) = k_0(K_1e_1 + e_2) + K_1\dot{e}_1 + \dot{e}_2 \quad (8)$$

Equation (7) becomes

$$\dot{s} = -k_0s + k_0\mu \text{sat}(s/\mu) + \Delta(e_1, e_2) + g(e_1, e_2)u \quad (9)$$

The controller u is defined:

$$u = -\Pi(e_1, e_2)\text{sat}(s/\mu) \quad (10)$$

This controller allows to robustly stabilize the system (2) in a semi-global manner.

Proof: We will now demonstrate that the control law defined in (10) can stabilize the class of nonlinear MIMO systems defined in (2). This proof is decomposed in two parts representing the internal and external regions of the boundary layer.

Part 1: Inside the region $\|s\| \geq \mu$, $\text{sat}(s/\mu) = s/\|s\|$.

In this part, we demonstrate that the control law in (10) with $\Pi(\cdot)$ defined later in (15) will bring the sliding surface inside the boundary layer. Before proceeding further, we introduce the following assumption.

Assumption 1: $\Delta(e_1, e_2)$ is bounded by a function of $\gamma(e_1, e_2)$ (where $\gamma(\cdot)$ is a class $\mathcal{K}$ function) and a positive constant Δ_0 :

$$\|\Delta(e_1, e_2)\| \leq \gamma(e_1, e_2) + \Delta_0 \quad (11)$$

and then,

$$\|\Delta(e_1 = 0, e_2 = 0)\| = \|f(0, 0)\| \leq \Delta_0 \quad (12)$$

for $(e_1, e_2) \in \mathbb{R}^n \times \mathbb{R}^n$ and while the sliding surface does not enter the boundary layer, i.e. $\|s\| \geq \mu$. $\square$

Lets now consider the product $s^T \dot{s}$

$$\begin{aligned} s^T \dot{s} &= -s^T k_0s + k_0\mu s^T \text{sat}(s/\mu) \\ &\quad + s^T \Delta(e_1, e_2) + s^T g(e_1, e_2)u \end{aligned} \quad (13)$$

This product $s^T \dot{s}$ can be developed with the previous assumption and the definition of saturation function (5):

$$\begin{aligned} s^T \dot{s} &= -s^T k_0s + k_0\mu s^T s/\|s\| + s^T \Delta(\cdot) - \Pi(\cdot)s^T g(\cdot)s/\|s\| \\ &\leq -k_0s^T s + k_0\mu s^T s/\|s\| + \|\Delta(\cdot)\|\|s\| \\ &\quad - \|\Pi(\cdot)\|\|s\| \\ &\leq -k_0\|s\|^2 - (\lambda\Pi(\cdot) - \gamma(\cdot) - k_0\mu - \Delta_0)\|s\| \\ &\leq -k_0\|s\|^2 - \lambda\Pi_0\|s\| \end{aligned} \quad (14)$$

where we define

$$\Pi(\cdot) = \Pi_0 + (\gamma(\cdot) + k_0\mu + \Delta_0)/\lambda \quad (15)$$

and Π_0 is a positive constant.

The product $s^T \dot{s}$ is then not positive and

$$\begin{cases} s^T \dot{s} \leq -k_0\|s\|^2 - \lambda\Pi_0\|s\| \leq -\lambda\Pi_0\|s\| \\ \frac{d\|s\|}{dt} = \frac{\sqrt{s^T \dot{s}}}{\|s\|} \leq -\lambda\Pi_0 \Rightarrow \|s(t)\| \leq \|s(0)\| - \lambda\Pi_0 t \end{cases} \quad (16)$$

Then the sliding surface $s(t)$ reaches the set $\|s(t)\| \leq \mu$ in finite time. $\square$

Part 2: In the region $\|s\| \leq \mu$, $\text{sat}(s/\mu) = s/\mu$.

In the following we denote $\mathcal{O}_\mu$ the region in the neighborhood of $(0, 0)$ with a radius R_μ .

$$\mathcal{O}_\mu = \{e = (e_1, e_2) \in \mathbb{R}^n \times \mathbb{R}^n \mid \|e\| \leq R_\mu\} \quad (17)$$

Consider again (3), (16), (9) and control law (10), which inside the boundary layer may be rewritten as (18) (remind that $\dot{e}_1 = e_2$).

$$\begin{cases} \dot{\sigma} = -k_0\sigma + s \\ \dot{e}_1 = -K_1e_1 + s - k_0\sigma \\ \dot{s} = \Delta(\cdot) - g(\cdot)\Pi(\cdot)s/\mu \end{cases} \quad (18)$$

Here we state the following assumption:

Assumption 2: function $g(e_1, e_2)$ satisfies two hypothesis:

Hypothesis 1: for $(e_1, e_2) \in \mathbb{R}^n \times \mathbb{R}^n$

$$g(e_1, e_2) + g^T(e_1, e_2) \geq 2\lambda I_n, \text{ the constant } \lambda > 0 \quad (19)$$

Hypothesis 2: Inside the region $(e_1, e_2) \in \mathcal{O}_\mu$, function $g(e_1, e_2)$ satisfies the Lipschitz-like condition:

$$\|g(e_1, e_2) - g(0, 0)\| \Delta_0 \leq \|g(0, 0)\|v(e_1, e_2) \quad (20)$$

A development of (20) gives us an alternative form for this hypothesis:

$$\|g(e_1, e_2)g^{-1}(0, 0) - I_n\| \Delta_0 \leq v(e_1, e_2) \quad (21)$$

in which, $v(e_1, e_2)$ is a suitable function satisfying

$$v(e_1, e_2) = v_1\|e_1\| + v_2\|e_2\| \leq K_v \quad (22)$$

where v_1, v_2 and K_v are suitable positive constants. $\square$

It can be shown that system (18) has an equilibrium point: $e_1 = e_2 = 0$, $s = \bar{s}$, $\sigma = \bar{\sigma}$ with $\bar{s} = k_0\bar{\sigma} = \mu \frac{g^{-1}(0, 0)f(0, 0)}{\Pi(0, 0)}$.

We can then conclude the design condition:

$$\|\bar{s}\| \leq \mu \Rightarrow \|(g^{-1}(0, 0)f(0, 0))\| \leq \Pi(0, 0) \quad (23)$$

where $\Pi(0, 0) = \Pi(e_1, e_2)|_{e_1=0, e_2=0}$

This condition can be satisfied by control law (15) using assumptions 1 and 2.

System (18) may be rewritten with respect to $\bar{s}$ and $\bar{\sigma}$:

$$\begin{cases} \dot{\bar{\sigma}} = -k_0\bar{\sigma} + \bar{s} \\ \dot{e}_1 = -K_1e_1 + \bar{s} - k_0\bar{\sigma} \\ \dot{\bar{s}} = \Delta(\cdot) - \Pi(\cdot)g(\cdot)\bar{s}/\mu - \Pi(\cdot)g(\cdot)\bar{\sigma}/\mu \end{cases} \quad (24)$$

where $\bar{\sigma} = \sigma - \bar{\sigma}$, $\bar{s} = s - \bar{s}$.

In the following we will show that state variables of the system (24) are driven to the equilibrium point when the control law (10) is applied, with $\Pi(\cdot)$ defined later in (15), when the system is inside the boundary layer. To attain this objective we introduce the following assumptions.

Assumption 3: Inside the boundary layer, i.e. $\|s\| \leq \mu$, function $f(\cdot)$ is Lipschitz, such that:

$$\|f(e_1, e_2) - f(0, 0)\| \leq l_1\|e_1\| + l_2\|e_2\| \quad (25)$$

where l_1 and $l_2 \in \mathbb{R}^+$.

Assumption 4: Inside the boundary layer, i.e. $\|s\| \leq \mu$, function $\Pi(\cdot)$ satisfies:

$$\Pi(\cdot) - \Pi(0, 0) \leq \chi(\cdot) = \lambda\chi_1\|e_1\| + \lambda\chi_2\|e_2\| \quad (26)$$

where χ_1, χ_2 are suitable positive constants and λ is defined earlier in (19).

We would like to demonstrate that every trajectory starting inside the boundary layer, will approach the equilibrium point as time tends to infinity. Toward that end, we take

$$W = \frac{\lambda_1}{2}\bar{\sigma}^T\bar{\sigma} + \frac{\lambda_2}{2}e_1^Te_1 + \frac{\bar{s}^T\bar{s}}{2} \quad (27)$$

as a Lyapunov candidate, where λ_1 and λ_2 are positive constants.

Its derivative can be easily calculated as:

$$\begin{aligned} \dot{W} &= \lambda_1\bar{\sigma}^T\dot{\bar{\sigma}} + \lambda_2e_1^T\dot{e}_1 + \bar{s}^T\dot{\bar{s}} \\ &= \lambda_1\bar{\sigma}^T(-k_0\bar{\sigma} + \bar{s}) + \lambda_2e_1^T(-K_1e_1 + \bar{s} - k_0\bar{\sigma}) \\ &\quad + \bar{s}^T(\Delta(\cdot) - \Pi(\cdot)g(\cdot)\bar{s}/\mu - \Pi(\cdot)g(\cdot)\bar{\sigma}/\mu) \end{aligned} \quad (28)$$

Since $(e_1, e_2) \in \mathcal{O}_\mu$, $\Delta(\cdot)$ can be expressed:

$$\Delta(\cdot) = k_0\bar{s} - k_0^2\bar{\sigma} - K_1^2e_1 + K_1\bar{s} - k_0K_1\bar{\sigma} + f(\cdot) \quad (29)$$

then,

$$\begin{aligned} \dot{W} &= \lambda_1\bar{\sigma}^T(-k_0\bar{\sigma} + \bar{s}) + \lambda_2e_1^T(-K_1e_1 + \bar{s} - k_0\bar{\sigma}) \\ &\quad + \bar{s}^T(k_0\bar{s} - k_0^2\bar{\sigma} - K_1^2e_1 + K_1\bar{s} - k_0K_1\bar{\sigma} \\ &\quad - \Pi(\cdot)g(\cdot)\bar{s}/\mu) + \bar{s}^T(f(\cdot) - \Pi(\cdot)g(\cdot)\bar{s}/\mu) \end{aligned} \quad (30)$$

In order to express the derivative of the Lyapunov function candidate more clearly, we consider firstly the term:

$$\begin{aligned} \|f(\cdot) - \Pi(\cdot)g(\cdot)\bar{s}/\mu\| &= \|f(\cdot) - \frac{\Pi(\cdot)}{\Pi(0,0)}g(\cdot)g^{-1}(0,0)f(0,0)\| \\ &= \|f(\cdot) - f(0,0) - \frac{\Pi(\cdot)}{\Pi(0,0)}[g(\cdot)g^{-1}(0,0) - I_n]f(0,0) \\ &\quad - \frac{\Pi(\cdot) - \Pi(0,0)}{\Pi(0,0)}f(0,0)\| \\ &\leq \|f(\cdot) - f(0,0)\| + \frac{\Pi(\cdot)}{\Pi(0,0)}\|g(\cdot)g^{-1}(0,0) - I_n\|\|f(0,0)\| \\ &\quad + \frac{\Pi(\cdot) - \Pi(0,0)}{\Pi(0,0)}\|f(0,0)\| \end{aligned} \quad (31)$$

using assumptions 1, 2, 3, 4 and the relation in (22), the previous equation can be expressed as:

$$\begin{aligned} &\|f(\cdot) - \Pi(\cdot)g(\cdot)\bar{s}/\mu\| \\ &\leq l_1\|e_1\| + l_2\|e_2\| + \frac{\Pi(\cdot)}{\Pi(0,0)}v(\cdot) + \frac{\chi(\cdot)\Delta_0}{\lambda\Pi(0,0)} \\ &\leq l_1\|e_1\| + l_2\|e_2\| + v(\cdot) + \frac{\chi(\cdot)v(\cdot)}{\lambda\Pi(0,0)} + \frac{\chi(\cdot)\Delta_0}{\lambda\Pi(0,0)} \\ &\leq (l_1 + v_1 + (\Delta_0 + K_v)\frac{\chi_1}{\Pi(0,0)})\|e_1\| + (l_2 + v_2 \\ &\quad + (\Delta_0 + K_v)\frac{\chi_2}{\Pi(0,0)})\|e_2\| \end{aligned} \quad (32)$$

Let us define $c_1 = l_1 + v_1 + (\Delta_0 + K_v)\frac{\chi_1}{\Pi(0,0)}$ and $c_2 = l_2 + v_2 + (\Delta_0 + K_v)\frac{\chi_2}{\Pi(0,0)}$, this expression may be rewritten as:

$$\begin{aligned} \|f(\cdot) - \Pi(\cdot)g(\cdot)\bar{s}/\mu\| &\leq c_1\|e_1\| + c_2\|e_2\| \\ &\leq (c_1 + c_2\|K_1\|)\|e_1\| + c_2\|\bar{s}\| + k_0c_2\|\bar{\sigma}\| \end{aligned} \quad (33)$$

From (24) and (33) the derivative of W can be developed:

$$\begin{aligned} \dot{W} &= -\lambda_1k_0\bar{\sigma}^T\bar{\sigma} + \lambda_1\bar{\sigma}^T\bar{s} - \lambda_2e_1^TK_1e_1 + \lambda_2e_1^T\bar{s} - \lambda_2k_0e_1^T\bar{\sigma} \\ &\quad + k_0\bar{s}^T\bar{s} - k_0^2\bar{s}^T\bar{\sigma} - \bar{s}^TK_1^2e_1 + \bar{s}^TK_1\bar{s} - k_0\bar{s}^TK_1\bar{\sigma} \\ &\quad - \Pi(\cdot)\bar{s}^Tg(\cdot)\bar{s}/\mu + \bar{s}^T(f(\cdot) - \Pi(\cdot)g(\cdot)\bar{s}/\mu) \\ &\leq -\frac{\lambda_1}{2}(\|\bar{\sigma}\| - \|\bar{s}\|)^2 - \frac{\lambda_2}{2}(\|\bar{s}\| - \|e_1\|)^2 - \frac{\lambda_2k_0}{2}(\|\bar{\sigma}\| - \|e_1\|)^2 \\ &\quad - \frac{k_0}{2}(\|\bar{\sigma}\| - \|\bar{s}\|)^2 - \frac{\|K_1\|^2}{2}(\|e_1\| - \|\bar{s}\|)^2 \\ &\quad - \frac{k_0\|K_1\|}{2}(\|\bar{\sigma}\| - \|\bar{s}\|)^2 - \frac{c_1+c_2\|K_1\|}{2}(\|e_1\| - \|\bar{s}\|)^2 \\ &\quad - \frac{c_2\|K_1\|}{2}(\|e_1\| - \|\bar{s}\|)^2 - \frac{k_0c_2}{2}(\|\bar{\sigma}\| - \|\bar{s}\|)^2 \\ &\quad - (\lambda_1k_0 - \frac{1}{2}(\lambda_1 + \lambda_2k_0 + k_0^2 + k_0\|K_1\| + k_0c_2))\|\bar{\sigma}\|^2 \\ &\quad - (\lambda_2\|K_1\| - \frac{1}{2}(\lambda_2 + \lambda_2k_0 + \|K_1\|^2 + c_1 + c_2\|K_1\|))\|e_1\|^2 \\ &\quad - (\Pi(\cdot)\|g(\cdot)\|/\mu - \frac{1}{2}(\lambda_1 + \lambda_2 + k_0 + k_0^2 + \|K_1\|^2 + \|K_1\| \\ &\quad + k_0\|K_1\| + c_1 + c_2\|K_1\| + c_2 + k_0c_2))\|\bar{s}\|^2 \end{aligned} \quad (34)$$

It can be verified that by taking λ_1, λ_2 and $\Pi(\cdot)$ large enough and μ small enough, the following conditions are satisfied:

$$\begin{cases} \lambda_1k_0 - 1/2(\lambda_1 + \lambda_2k_0) > 1/2(k_0^2 + k_0\|K_1\| + k_0c_2) \\ \lambda_2\|K_1\| - 1/2(\lambda_2 + \lambda_2k_0) > 1/2(\|K_1\|^2 + c_1 + c_2\|K_1\|) \\ \Pi(\cdot)\|g(\cdot)\|/\mu - 1/2(\lambda_1 + \lambda_2) > 1/2(k_0 + k_0^2 + \|K_1\|^2 + \|K_1\| \\ + k_0\|K_1\| + c_1 + c_2\|K_1\| + c_2 + k_0c_2) \end{cases} \quad (35)$$

In this way, $W(t)$ satisfies $W(t) > 0$ and $\dot{W} < -w_0W$ (where w_0 is a positive constant) for all $\sigma \neq \bar{\sigma}$, $e_1 \neq 0$ and $s \neq \bar{s}$. Then $W(t)$ reaches exponentially zero when time tends to infinite. As consequence, the output error $e_1(t)$ tends to zero and σ and s tend to their equilibrium values as time tends to infinite. We may assure the exponential stability of the system in the region of $\|s\| \leq \mu$.

We can then state the results developed above in the form of the theorem:

Theorem 2.1: A class of Multi-Input Multi-Output non-linear systems described by (1), and satisfying assumptions (1-4) can be stabilized semi-globally to their constant reference by the controller (10-3-4-15) with tuning parameters (Π_0, k_0, μ and K_1 defined in the previous section) and function $\gamma(\cdot)$ conveniently set. Furthermore, the stability is exponential inside an error region defined in (17).

In the following section we will apply this result to a nonlinear MIMO aircraft control problem. The linearized version of this problem is already addressed in [12].

III. EXAMPLE: F-16 AIRCRAFT'S LATERAL MODE CONTROL DESIGN

In this section, we address the control of a nonlinear MIMO system applying the results obtained in the previous section. The considered system is the nonlinear MIMO model of an F-16 aircraft lateral mode. The work of [12] has addressed this case designing a conditional integrator based on the linearization of the system around an operating point. In the present case, we extend those results and those of [6] addressing the nonlinear MIMO system without linearization.

The F-16 aircraft lateral mode has two inputs (aileron and rudder) and two outputs (sideslip angle and roll angle). In this way, only lateral state variables are time varying. Others longitudinal state variables (like height and pitch) are considered as constant or null. The lateral dynamic model used to the control design procedure is consequently reduced as below (see [14] and [11]):

$$\begin{cases} \dot{\beta} = \frac{1}{mV} (-\cos(\alpha_0) \sin(\beta)(T + C_x(\alpha_0)\bar{q}S) \\ \quad + \cos(\beta)C_y(\beta)\bar{q}S - \sin(\alpha_0) \sin(\beta)C_z(\alpha_0, \beta)\bar{q}S) \\ \quad + \sin(\alpha_0)p - \cos(\alpha_0)r + \frac{\rho S}{4m} (\cos(\beta)C_{y_p}(\alpha_0)\bar{b}p \\ \quad + \cos(\beta)C_{y_r}(\alpha_0)\bar{b}r) + \frac{\rho}{V} (\cos(\alpha_0) \sin(\beta) \sin(\theta_0) \\ \quad + \cos(\beta) \cos(\theta_0) \sin(\phi) - \sin(\alpha_0) \sin(\beta) \cos(\phi)) \\ \dot{\phi} = p + \cos(\phi) \tan(\theta_0)r \\ \dot{p} = I_3 C_l(\alpha_0, \beta)\bar{q}S\bar{b} + I_4 C_n(\alpha_0, \beta)\bar{q}S\bar{b} + \frac{\rho V S \bar{b}}{4} [(I_3 C_{l_p}(\alpha_0) \\ \quad + I_4 C_{n_p}(\alpha_0))p + (I_3 C_{l_r}(\alpha_0) + I_4 C_{n_r}(\alpha_0))r] \\ \quad + \bar{q}S[(I_3 C_{l_{\delta_a}}(\alpha_0) + I_4 C_{n_{\delta_a}}(\alpha_0))\delta_a + (I_3 C_{l_{\delta_r}}(\alpha_0) \\ \quad + I_4 C_{n_{\delta_r}}(\alpha_0))\delta_r] \\ \dot{r} = I_4 C_l(\alpha_0, \beta)\bar{q}S\bar{b} + I_9 C_n(\alpha_0, \beta)\bar{q}S\bar{b} + \frac{\rho V S \bar{b}}{4} [(I_4 C_{l_p}(\alpha_0) \\ \quad + I_9 C_{n_p}(\alpha_0))p + (I_4 C_{l_r}(\alpha_0) + I_9 C_{n_r}(\alpha_0))r] \\ \quad + \bar{q}S[(I_4 C_{l_{\delta_a}}(\alpha_0) + I_9 C_{n_{\delta_a}}(\alpha_0))\delta_a + (I_4 C_{l_{\delta_r}}(\alpha_0) \\ \quad + I_9 C_{n_{\delta_r}}(\alpha_0))\delta_r] \end{cases} \quad (36)$$

in which, $\bar{b}$ is the equivalent length, S the equivalent wing surface, $I_{xx}, I_{yy}, I_{zz}, I_{xz}$ are the moments of inertia in kgm^2 . $I_3 = \frac{I_{zz}}{(I_{xx}I_{zz} - I_{xz}^2)}$, $I_4 = \frac{I_{xz}}{(I_{xx}I_{zz} - I_{xz}^2)}$, $I_9 = \frac{I_{xx}}{(I_{xx}I_{zz} - I_{xz}^2)}$. m is the mass of the system (kg) and g the gravity constant, α_0, θ_0 and V are angle of attack, pitch angle and airspeed considered as constant in the studied case, T is the thrust force. The state variables of the system are β, ϕ, p, r which represent the sideslip angle, roll angle, roll rate, yaw rate, respectively. $C_y(\alpha_0), C_{y_p}(\alpha_0), C_{y_r}(\alpha_0), C_l(\alpha_0, \beta), C_n(\alpha_0, \beta), C_{l_p}(\alpha_0), C_{n_p}(\alpha_0), C_{l_r}(\alpha_0), C_{n_r}(\alpha_0), C_{l_{\delta_a}}(\alpha_0), C_{n_{\delta_a}}(\alpha_0), C_{l_{\delta_r}}(\alpha_0), C_{n_{\delta_r}}(\alpha_0)$ are lateral aerodynamic coefficients taken from [7]. Finally, the control inputs are respectively the aileron (δ_a) and the rudder (δ_r).

The equations system (36) can be rearranged as:

$$\begin{cases} \begin{bmatrix} \dot{\beta} \\ \dot{\phi} \end{bmatrix} = f_{11}(\beta, \phi) + f_{12}(\beta, \phi) \begin{bmatrix} p \\ r \end{bmatrix} \\ \begin{bmatrix} \dot{p} \\ \dot{r} \end{bmatrix} = f_{21}(\beta, \phi) + f_{22}(\beta, \phi) \begin{bmatrix} p \\ r \end{bmatrix} + g_2(\beta, \phi) \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \end{cases} \quad (37)$$

where $f_{11}(\cdot), f_{12}(\cdot), f_{13}(\cdot), f_{21}(\cdot), f_{22}(\cdot)$, and $g_2(\cdot)$ represent the terms of (36) respectively. Equation (37) is mainly used for controller design and stability analysis.

Let us define $x_1 = [\beta, \phi]^T$, $x_2 = \dot{x}_1 = [\dot{\beta}, \dot{\phi}]^T$ and $u = [\delta_a, \delta_r]^T$. (37) can be written again as:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = F'(x_1, x_2) + G'(x_1, x_2)u \end{cases} \quad (38)$$

where

$$\begin{cases} F'(\cdot) = \left(\frac{\partial f_{11}(\cdot)}{\partial x_1} + (f(\cdot) + f_{12}(\cdot)f_{22}(\cdot))(f_{12}(\cdot))^{-1}x_2 \right. \\ \quad \left. - (f(\cdot) + f_{12}(\cdot)f_{22}(\cdot))(f_{12}(\cdot))^{-1}f_{11}(\cdot) + f_{12}(\cdot)f_{21}(\cdot) \right) \\ G'(\cdot) = f_{12}(\cdot)g_2(\cdot) \\ f(\cdot) \begin{bmatrix} p \\ r \end{bmatrix} = \frac{\partial(f_{12}(\cdot) \begin{bmatrix} p \\ r \end{bmatrix})}{\partial x_1} \end{cases} \quad (39)$$

Let $y_{ref} = x_{1ref} = [\beta_{ref}, \phi_{ref}]^T$ be a prescribed reference output considered as constant function and $y(t) = x_1(t)$ is the actual output. In order to design the MIMO conditional integrator controller, we define output error vector $e_1 = x_1 - x_{1ref}$ and $e_2 = \dot{e}_1$. (39) can be transformed into (40) with the intervention of two new state variables e_1 and e_2 .

$$\begin{cases} \dot{e}_1 = e_2 \\ \dot{e}_2 = F(e_1, e_2) + G(e_1, e_2)u \end{cases} \quad (40)$$

$G(x_1, x_2)$ is invertible for $\beta \in (-30^\circ, 30^\circ)$ and $\phi \in (-90^\circ, 90^\circ)$, $F(x_1, x_2)$ is Lipschitz for the entire domain of actuation of the system. Application of control law (10) for system (40) leads to the controller:

$$\begin{cases} u = -\Pi(e_1, e_2) \text{sat}(s/\mu) \\ \Pi(\cdot) = \Pi_0 + (\gamma(\cdot) + k_0\mu + \Delta_0)/\lambda \end{cases} \quad (41)$$

with

$$\begin{cases} s = k_0\sigma + K_1e_1 + e_2 \\ \dot{\sigma} = -k_0\sigma + \mu \text{sat}(s/\mu) \end{cases} \quad (42)$$

where $\lambda = \min(\|G(\cdot)\|)$ for $\beta \in (-30^\circ, 30^\circ)$ and $\phi \in (-90^\circ, 90^\circ)$. Δ_0 is a positive constant, Π_0 is a positive parameter large enough, k_0 is a positive parameter, μ is the boundary layer small enough and K_1 is chosen such a way that $K_1 + sI_2$ is Hurwitz. All Π_0, k_0, μ, K_1 and $\gamma(\cdot)$ are to be determined.

IV. SIMULATION RESULTS

In sections II and III, the design methodology of the conditional integrator controller to track sideslip angle and roll angle is proposed taking into account the full aerodynamic characteristics of the lateral mode of an F-16 aircraft. This section presents numerical simulation results for the controller to illustrate the performance of the proposed conditional integrator control laws.

As mentioned in section III, we use the lateral mode F-16 nonlinear model in this paper. This because its nonlinear model, wind tunnel informations and data are widely known and used for control design. It is important to remark that the model used in the following simulations is even more complete than that used in the control design, for example it includes actuator dynamics and their limitations. As a consequence, simulations also illustrate some properties of robustness to unmodeled dynamics.

In the following simulations, we have applied the MIMO conditional integrator controller for controlling the sideslip and roll angle of the lateral mode of the F-16 aircraft model. We may note that the control inputs are always limited by $|\delta_a| < 21.5^\circ$ and $|\delta_r| < 30^\circ$, which represent the physical limitations of these actuators.

The system is studied at the operating point $(V, h) = (153m/s, 4650m)$ corresponding to the trimmed angle of attack $\alpha_0 = 2.1^\circ$, pitch angle $\theta_0 = 2.1^\circ$, sideslip $\beta_0 = 0^\circ$, $\phi_0 = 0^\circ$ and to trimmed control surface values: aileron $\delta_a = 0^\circ$ and rudder $\delta_r = 0^\circ$. The control law in (41) whose $\Pi(\cdot)$ can be written in a more simple form as below:

$$\Pi(\cdot) = \Pi_0 + \gamma(\cdot) \quad (43)$$

in which, $\gamma(\cdot) = \gamma_1 \|e_1\| + \gamma_2 \|e_2\|$, γ_1 and γ_2 are positive constant.

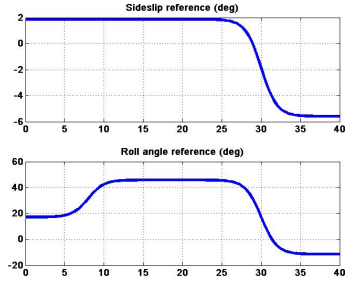


Fig. 1. Reference input of sideslip and roll angle

Π_0	μ	γ_1 and γ_2	k_0	K_1
$\begin{bmatrix} 20 & 0.0 \\ 0.0 & 20 \end{bmatrix}$	1.0	0.1 and 0.1	$\begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 2.0 & 0.0 \\ 0.0 & 2.0 \end{bmatrix}$

TABLE I

PARAMETERS FOR THE LATERAL MODE CONTROL

The reference input Fig. 1 is taken as in [12] and [13] with a small change in its amplitude. It consists of a step change in roll angle at $t = 8s$ and of a step change in sideslip and roll angles at $t = 30s$ as:

$$\begin{bmatrix} \beta_{ref}(t) \\ \phi_{ref}(t) \end{bmatrix} = \begin{bmatrix} 0.13(-0.25 + \frac{1}{1+e^{t-30}} - 0.5) \\ 1.0(-\frac{0.5}{1+e^{t-8}} + \frac{1}{1+e^{t-30}} - 0.2) \end{bmatrix} \quad (44)$$

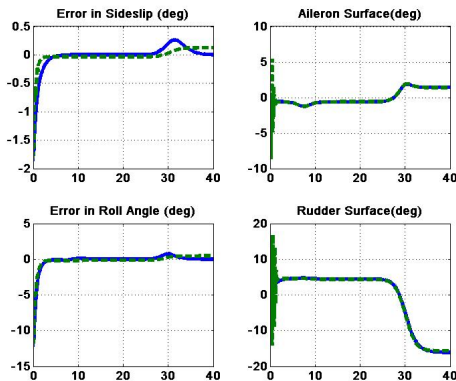


Fig. 2. Output Errors and Control Surfaces for a small initial condition. CI (solid) - SMC (dashed)

Two initial conditions of sideslip and roll angle are studied. The first corresponds to a small initial condition of sideslip

and roll angle ($\beta_0 = 0^\circ, \phi_0 = 5^\circ$) to the equilibrium point, and the second corresponds to a high one ($\beta_0 = -10^\circ, \phi_0 = 20^\circ$) from the equilibrium point.

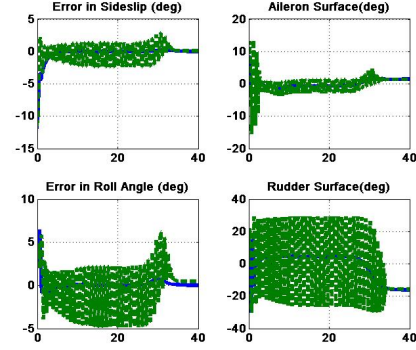


Fig. 3. Output Errors and Control Surfaces for a high initial condition. CI (solid) - SMC (dashed)

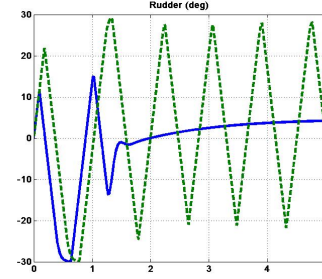


Fig. 4. Detail on Control Surface (Rudder) for a high initial condition. CI (solid) - SMC (dashed)

The plots in Fig. 2 and Fig. 3 are obtained with the controller structure described in section III (solid lines) which is compared to a sliding mode controller (dashed lines) where the sign function is replaced by a saturation function to avoid chattering. This controller is used to regulate sideslip angle and roll angle errors. Fig. 2 represents the system output errors in respect to references. The conditional integrator controller provides a convergence to zero of the output error (solid lines) for both sideslip and roll angle in the two cases of initial conditions. In contrast, the sliding mode controller (SMC) produces an output error (dash lines) with non-zero steady state error. This well illustrate the positive contribution of integral term to recover steady state information.

The benefit of the proposed controller is even more clear in the case of high initial conditions where the SMC bring the system to a not constant behavior, even if still stable.

Fig. 3 shows the control surfaces of the system for both controllers. The solid lines correspond to control surfaces (aileron and rudder) of conditional integrator controller. The dash lines represent control surfaces of conditional integrator controller.

We present a detail of Fig. 3 in Fig. 4. There it is quite clear the mechanism of the conditional integrator. One may observe that for large initial conditions, the system is driven to the boundary layer where it is captured by the exponential convergence property of the controller. From there on, the controller behaves in a very smooth way.

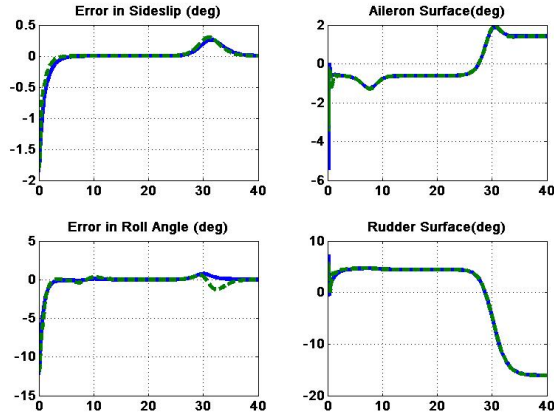


Fig. 5. Output Errors and Control Surfaces for a small initial condition. CI (solid) - Controller in [12] (dashed)

The plots in Fig. 5 to Fig. 6 are obtained with the controller structure described in section III and the controller designed in [12] for the F-16 aircraft lateral mode, linearized at the equilibrium point ($V = 153m/s, \alpha = 2.1^\circ$) used to regulate sideslip angle and roll angle errors. All parameters are obtained from control design in [12].

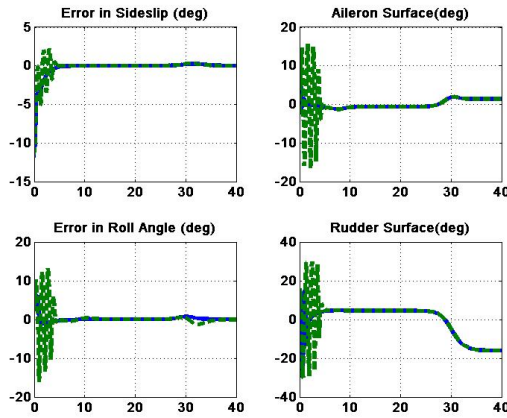


Fig. 6. Output Errors and Control Surfaces for a high initial condition. CI (solid) - Controller in [12] (dashed)

The two controllers give the same output error and control surfaces (aileron and rudder) for the case of small initial conditions (Fig. 5). That demonstrate the good performance of both controllers. For high initial conditions, the controller in [12] based on linearizations may have very large oscillations (up to 16° in Fig. 6). Moreover, Figs 7 that presents a closer view of Fig. 6 well illustrate the behavior of the controller, with an exponential convergence since entering a residual region, obtaining a better transitory for large initial conditions.

V. CONCLUSION

This paper has developed a conditional integrator controller for a class of MIMO nonlinear systems. Conditional integrators are an interesting solution for particular uncertain problems largely found in practice, and in particular in aerospace. This control technique has been developed in a series of papers found in literature in the last decade.

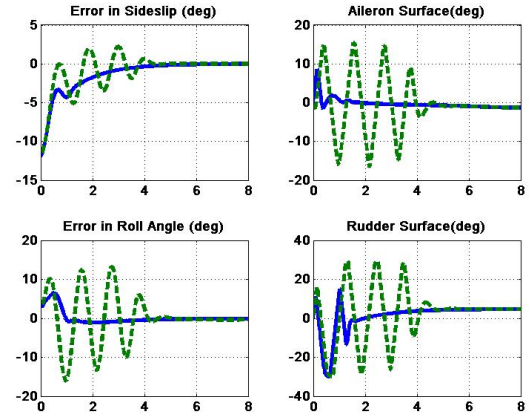


Fig. 7. Detail on Output Errors and Control Surfaces for a high initial condition. CI (solid) - Controller in [12] (dashed)

The present work has extended these results from the literature, for a class of MIMO nonlinear systems strongly motivated by aerospace problems. The main results are summarized in a theorem concerning a general class of systems, and shows that for mild and easily fulfilled assumptions, these systems can be semi-globally stabilized in finite time towards a region of the state space. Finally, the controller exponentially stabilizes the system inside this region to its equilibrium. These results are currently being applied to more general and complex aerospace problems with very interesting results. Flight tests are expected to be carried out in a near future to corroborate theoretical results.

REFERENCES

- [1] Alberto Isidori. *Nonlinear Control Systems*. Springer Verlag, 1995.
- [2] Hassan K. Khalil. On the design of robust servomechanisms for minimum phase nonlinear systems. *Robust Nonlinear Control 2000*, John Wiley & Sons, 10:339–361, 2000.
- [3] Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, 2001.
- [4] Nazmi A. Mahmoud and Hassan K. Khalil. Robust control for a nonlinear servomechanism problem. *IEEE - Transactions on Automatic Control*, 1996.
- [5] Riccardo Marino and Patrizio Tomei. *Nonlinear Control Design: Geometric, Adaptive and Robust*. Prentice Hall, 1st edition, 1995.
- [6] Attaullah Y. Memon and Hassan K. Khalil. Output regulation of nonlinear systems using conditional servocompensators. *Automatica*, 2010.
- [7] Eugene Morelli. Global nonlinear parametric modeling with application to f-16 aerodynamics. *American Institute of Aeronautics and Astronautics*, 1998.
- [8] Sridhar Seshagiri and Hassan K. Khalil. Robust output feedback regulation of minimum-phase nonlinear systems using conditional integrators. *Automatica*, 2005.
- [9] Sridhar Seshagiri and Hassan K. Khalil. Robust output regulation of minimum phase nonlinear systems using conditional servocompensators. *Wiley InterScience*, 2005.
- [10] Sridhar Seshagiri and Ekprasis Promtun. Sliding mode control of f-16 longitudinal dynamics. *American Control Conference*, pages 1770–1775, 2008.
- [11] Lars Sonneveldt. *Nonlinear F-16 Model Description*. Delft University of Technology, Netherlands, 2006.
- [12] Hoa Vo and Sridhar Seshagiri. Robust control of f-16 lateral dynamics. *Industrial Electronics, 2008. IECON 2008. 34th Annual Conference of IEEE*, pages 343–348, 2008.
- [13] Amanda Young, Chengyu Cao, and Naira Hovakimyan. An adaptive approach to nonaffine control design for aircraft applications. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2006.
- [14] Peter H. Zipfel. *Modeling and Simulation of Aerospace Vehicle Dynamics, 2nd edition*. American Institute of Aeronautics and Astronautics, 2000.

Optimal Positioning System with Fuzzy Logic Controller

Constantin-Catalin Dosoftei, Lucian Mastacan,

Abstract— Research undertaken for the behavior of "human expert" showed that it is specific to its highly nonlinear behavior, with the effects of anticipation, integration, prediction and adapting itself to the concrete conditions of operation. Control processes using fuzzy logic controllers from this point of view is more pragmatic and can be adapted to a much wider class of processes. This paper presents fuzzy control system of DC motor speed. Choice is justified by the large number of applications in electric drive systems. Fuzzy logic combined with optimal tracking theory is proposed for the positioning control of a DC motor.

I. INTRODUCTION

THE problems of automatic synthesis of systems that can be formulated about the performance and conditions that must be made, a distinct category of problems is the imposed response. In this case the desired behavior of the system isn't specified by some index of quality, but also through a trajectory that should be carried out by automatic system. For example, can be imposed desired shape of unit step response. Although this approach to the synthesis of automated systems is often natural for engineering practice, the procedure is less used because of a more laborious calculation, and because much wider spread of values-based methods impose certain performance indices.

In the case of imposed response method there are situations where we should not expect that getting accurate trajectory (which, actually, is not achievable) but on average as a better approximation of it. Approaching trajectory can be imposed in various ways. Among these is the using a fuzzy controller rules be formulated so as to obtain small deviations from the trajectory as imposed.

A particular case of systems with this structure is the optimal block consideration of a prescribed size variation in the system. In this way, it ensures a significant simplification of the implementation of optimal control systems. If the imposed trajectory tracking is provided by a fuzzy logic controller, the system is optimal-fuzzy [1].

Manuscript received June 18, 2011.

C.C. Dosoftei is with the Technical University "Gh. Asachi", Faculty of Automatic Control and Computer Science, Iași, 700050, Str. D. Mangeron 27, ROMÂNIA (phone: +40(0) 232 278680; fax: +40(0) 232 214290; e-mail: cdosoftei@ac.tuiasi.ro).

L. Mastacan is with the Technical University "Gh. Asachi", Faculty of Automatic Control and Computer Science, Iași, 700050, Str. D. Mangeron 27, ROMÂNIA (e-mail: lmastacan@ac.tuiasi.ro).

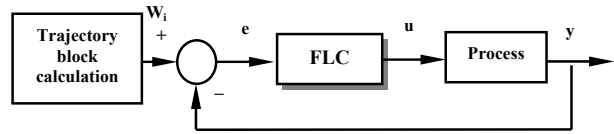


Fig. 1. Fuzzy tracking system

In carrying out an automatic control structure with fuzzy logic controllers information on the process may take the form of a mathematical model more or less accurately, or information about the process can be available in a linguistic representation.

II. OPTIMIZATION PROBLEM

In this paper authors focused on a particular case of the problem described above in fig. 1 and refer to a particular application tracking system optimal from energetic point of view to a DC motor. Optimization of energy efficient drives systems for positioning and presents not only the advantage of reducing energy consumption, but can offer - in some cases - and the possibility of reducing the motor's nominal power [2],[3]. In fact, the design stage, the rated power is determined from the condition of heating the motor, and by reducing losses due to the Joule heating to reach more favorable conditions, namely the possibility of lowering the rating. This leads to lower costs and reduce the gabarite and weight, which in some applications is an important advantage. Let us note that in some cases reducing the weight of a subset may have the effect of reducing energy consumption of the entire ensemble.

Optimal control process can be implemented as requiring a more accurate tracking of a size optimal trajectory. Of course such a system is suboptimal, but close to optimal behavior is very good. Implementation of this way of optimal control of the dynamic processes can be considerably simpler than that based on the application of optimal controls. In this procedure, the approximation of optimal and actual trajectory is provided by a fuzzy controller. It uses a conventional control structure, cascaded with some modifications.

The mathematical model of a drive system with DC motor with separate excitation is [4]:

$$\begin{aligned}\dot{\theta} &= r\omega \\ \dot{\omega} &= -\frac{\rho}{J}\omega + \frac{C_m}{J}i - \frac{1}{J}m \\ \dot{i} &= -\frac{C_e}{L}\omega - \frac{R}{L}i + \frac{1}{L}u\end{aligned}\quad (1)$$

where θ is the angular deviation of the controlled axis, ω is the motor speed, r is the gear constant, ρ is the viscous friction coefficient, J is the inertia, m is the load torque, C_e , C_m are the constants of the motor, and R , L are the resistance and inductance of the controlled circuit of the motor.

In general, resistant torque can be constant or can vary over time as a law some known or random, depending on the speed or position.

The system (1) can be written in the form [3]

$$\dot{\tilde{x}}(t) = A\tilde{x}(t) + Bu(t) + \tilde{w}(t) \quad (2)$$

where

$$\tilde{x}(t) = [\tilde{x}_1(t) \quad \tilde{x}_2(t) \quad \tilde{x}_3(t)]^T = [\theta(t) \quad \omega(t) \quad i(t)]^T \quad (3)$$

is the state vector (T denotes the transposition), $u(t)$ is the control variable, and

$$\tilde{w}(t) = \begin{bmatrix} 0 \\ -1/J_r \\ 0 \end{bmatrix} m(t) \quad (4)$$

is the disturbance vector. The matrices from (2) are

$$A = \begin{bmatrix} 0 & r & 0 \\ 0 & -\rho/J & C_m/J \\ 0 & -C_e/L & -R/L \end{bmatrix}; \quad B = \begin{bmatrix} 0 \\ 0 \\ 1/L \end{bmatrix} \quad (5)$$

An important simplification of the implementation is done if consider rotor current $i(t)$ as the control variable instead of control voltage $u(t)$. Inner loop having a fast response can be accepted that the control variable $i(t)$ changes instantly, as usually assumed these variables. In this way is reached at an ideal optimal system, which neglects changes in current. But knowing the ideal form for the changing variable of command is very useful, because it will require the system to work with the best possible approximation of the optimal variation. For a better adequacy of the system as the optimal real, it may be the ideal reference current variation, speed or position. In this paper was imposed the ideal form of the trajectory speed.

The equation of state is:

$$C_m \cdot i(t) = J \cdot \frac{d\omega}{dt} + m \quad (6)$$

where m is the load torque (perturbation).

Optimization problem is considered in terms of energy which requires minimizing losses in the electric motor.

At operation in stationary regime the electric motor copper losses predominate, followed by iron. In dynamic regime, the current is very high so that copper losses

significantly exceed other categories of losses, as such, it is justified to refer only to optimize the minimization of losses in copper and therefore the quality index is adopted in the optimization problem is form:

$$I = \int_0^T i^2(t) dt \quad (7)$$

The optimization problem is to determine $i(t)$ on the interval $[0, T]$ which transfers the system (6) of initial and final conditions imposed so as to minimize the criterion (7).

Initial conditions refer to speed and position in moment 0 and T:

$$\begin{aligned}\omega(0) &= 0; \quad \omega(T) = 0; \\ \theta(0) &= 0; \quad \theta(T) = \theta_f;\end{aligned} \quad (8)$$

According to [3] is reached by solving the problem, the following relationships to calculate the optimal values for state variables:

$$\begin{aligned}i^*(t) &= 6 \frac{J}{rC_m} \cdot \frac{\theta_f}{T^2} (1 - 2\frac{t}{T}) + \frac{m}{C_m} \\ \omega^*(t) &= 6 \frac{\theta_f}{rT^2} (t - \frac{t^2}{T}) \\ \theta^*(t) &= 6 \frac{\theta_f}{T^2} (3t^2 - 2\frac{t^3}{T})\end{aligned} \quad (9)$$

In (9) shows that the optimal speed is independent of torque m , that is the optimal speed variation remains the same regardless of the load torque. In other words, optimal management, the motor absorbs a current to compensate for torque is the trajectory speed so they always stay the same. This justifies the choice of optimal speed variation like set point as noted above.

III. SYSTEM IMPLEMENTATION

The fuzzy optimal system is presented in fig.2, where θ^* is the imposed value of the angular displacement, ω^* is the prescribed variation for speed, C_1 and C_2 are the controllers and M is the motor. The controllers are PI and their setting parameters not covered by this paper and are in conformity with [3] and [4].

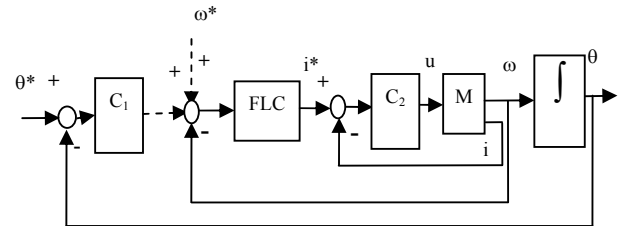


Fig. 2. The structure of the fuzzy optimal system

In fig. 2 is presented the block diagram where appear optimal positioning system with fuzzy controller.

Fuzzy controller implemented using the Fuzzy Logic

Toolbox of Matlab's input is the tracking error and error variation, and the output is the change in output quantity.[5]

The two entries on each of three fuzzy sets defined on each universe of discourse with triangular type membership functions and trapezoid. Output universe of discourse is defined by five fuzzy sets with triangular type membership functions.

The method of Mamdani-type inference is used: "min-max" and defuzzification method is the center of gravity.[6]

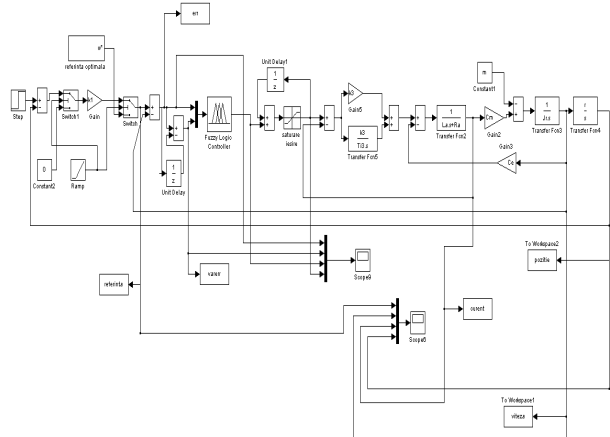


Fig. 3. Implementation of fuzzy optimal system in SIMULINK

Defined rule base is shown in fig. 4.

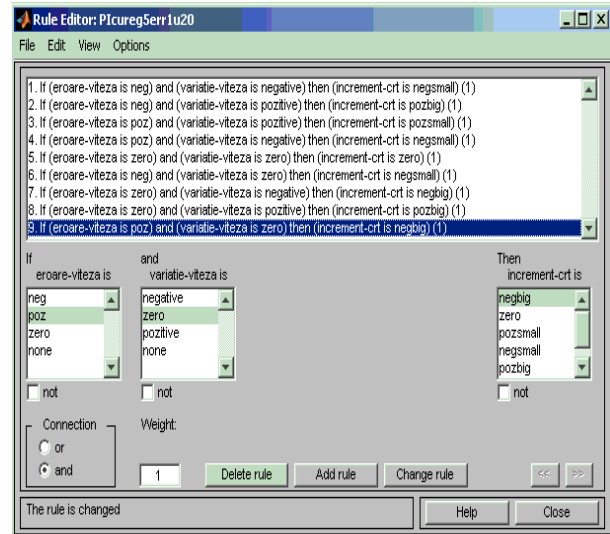


Fig. 4. Rule base implementation in Matlab

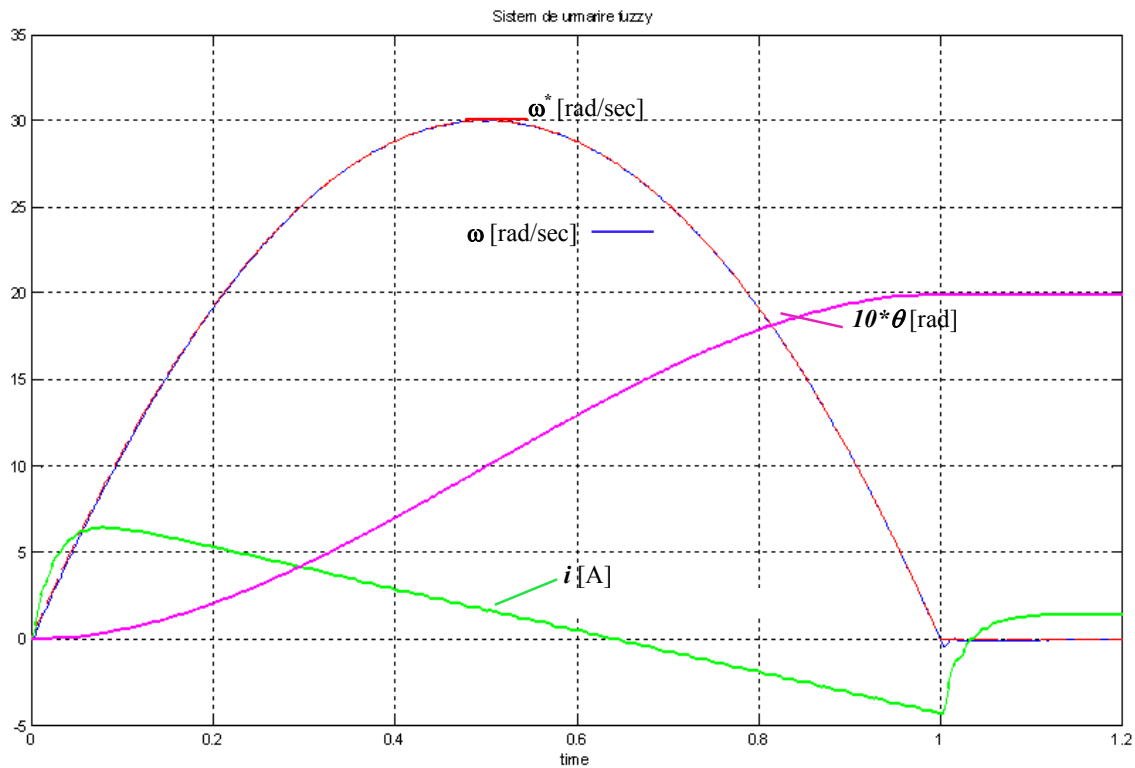


Fig. 5 Evolution in time of state values

At positioning systems, start from standstill, $\omega(0)=0$, and the final state of the system is same, the standstill, $\omega(T) = 0$.

To see all the state values on the same graph on a scale appropriate to the value represented on the chart position θ multiplied by 10.

The existence of potentially strong torque makes the current I , to change the sign.

Except for start and stop times, we see that current variation is relatively linear.

There is an optimal speed tracking accurately calculated with (9), tracking error is very small. The small variations observed around the point "t = 1 s, which is currently switching, where practically the position loop is closed (so far the system works only as a system to track optimal velocity calculated in terms of energy).

I. CONCLUSION

Calculate the amount of energy optimal speed for optimal positioning DC servo system and implemented in Matlab / Simulink structure fuzzy tracking controller. With fuzzy controller in internal loop was touched the evolution of $\omega(t)$ very closed by the desired optimal $\omega^*(t)$. The type of fuzzy controller is cvasi-PI.

REFERENCES

- [1] D.G. Burkhard and P.P. Bonissone, "Automated fuzzy knowledge base generation and tuning" *In: Proc. of the IEEE Int. Conf. on Fuzzy Systems*, San Diego, 1992, p. 179-196
- [2] B.D.O. Anderson, J.B. Moore, "Optimal Linear Quadratic Methods", *Prentice Hall*, Englewood Cliffs, New Jersey, 1990
- [3] C. Botan, F. Ostafi, A. Onea, "Fixed End-Point Problem for a Servo Drive System", *IEEE International Symposium on Signals, Circuits and Systems*, SCS 2003, p. 321 - 324, ISBN 0-7803-7979-9.
- [4] C. Mademlis, N.Margaris, "Loss Minimization in Vector-Controlled Interior Permanent-Magnet Synchronous Motor Drives", *IEEE Trans. on Ind. Electronics*, Vol. 49, No. 6, 2002, p. 1344-1347.
- [5] Z. Johanyák, D. Tikk, S. Kovács, K. W. Wong, "Fuzzy rule interpolation Matlab toolbox _ FRI toolbox", *15th Int. Conf. On Fuzzy Systems (FUZZ-IEEE'06)*, Vancouver, Canada, 2006, p 1427-1433
- [6] L. Reznik, "Fuzzy Controllers", *Oxford OX2 8DP*, Newnes, ISBN 0 7506 3429 4, 1997 pp. 78-93

Alternative Control Solutions for Vehicles with Continuously Variable Transmission. A Case Study

Claudia-Adina Dragoș, Stefan Preitl, *Senior Member, IEEE*, Radu-Emil Precup, *Senior Member, IEEE*, Emil M. Petriu, *Fellow, IEEE*, Mircea-Bogdan Rădac and Alexandra-Iulia Stînean

Abstract—The paper employs the sensitivity analysis with respect to the parametric variations of the controlled plant in the low-cost controller designs for a vehicle power train systems with spark-ignition engine and continuously variable transmission. The three control solutions suggested in order to ensure the asymptotic speed tracking include a PI controller, a PID controller and a one-degree-of-freedom Takagi-Sugeno fuzzy controller. The digital simulation results prove that the low-cost control solutions ensure good control system performance with respect to the modifications of the reference input and to a category of parametric disturbances caused by the modifications of the vehicle's mass. A comparison between the control solutions is offered.

I. INTRODUCTION

The paper treats aspects concerning the modeling and offers low-cost vehicle speed control solutions (CSs) based on classical PI(D) controllers and on Takagi-Sugeno fuzzy controllers (TS-FCs). The low-cost control solutions (LCCSs) are designed such that to ensure asymptotic speed tracking for a vehicle power train system with spark-ignition engine and continuously variable transmission (CVT).

The idea of low-cost automation (LCA) or of LCCSs has been introduced in the 1990's [1], and the LCCSs have many particularities and advantages. Various simple and effective CSs have been implemented as LCCSs in automotive applications. From the control point of view, the aim of LCA is to develop innovative and intelligent control solutions that integrate efficiently the human experience with current control techniques and not to accept low control system performance. LCCSs are reported in many applications of mechatronics systems that include the automotive ones [2], [3].

This paper offers the application-oriented analysis and development of two families of LCCSs concerning CSs

with PI(D) controllers and CSs with Takagi-Sugeno fuzzy controllers (TS-FCs). These solutions improve the control system performance while maintaining the low-cost idea. The constraints imposed in mechatronics applications and the changing of their parameters lead to the necessity of ensuring the stability and the robustness of CSs, which require the efficient selection of design methods.

The CSs with TS-FCs are accepted as LCA solutions due to the possibility to integrate linear (PI(D)) controllers and to use relatively simple and easily understandable fuzzy control rules [4]. These CSs belong to the class of LCCSs because they use a reduced number of linguistic terms, they have a simple structure and can be implemented and tested easily in laboratory equipment.

The controllers in our LCCSs are designed on the basis of locally linearized mathematical models (MMs) of the controlled plant. The PI(D) controllers are first developed in the frequency domain to ensure the desired shape and values of the sensitivity function $S(j\omega)$. The approximate equivalence between fuzzy controllers and the linear ones in certain conditions [3] is next included in TS-FC design to guarantee good system responses with respect to reference inputs and disturbance inputs as well. A good robustness with respect to model uncertainties can also be provided.

The paper is organized as follows. Using the MM of the plant given in Section II, Section III is focused on the design of the LCCSs. Section IV is dedicated to the sensitivity analysis in the frequency domain. Some digital simulation results for one main driving scenario are presented in Section V to validate the developed CSs; they show the improvement of the control systems' behaviors for some values of the reference input and for some operating points. Section V highlights the conclusions.

II. MATHEMATICAL MODELING

The schematic structure of a vehicular power train system for a front-engine front-wheel-driven automobile with CVT is presented in Fig. 1. The approach starts with an analysis of systems' components which include the electronic throttle, the spark-ignition internal combustion engine, the torque converter (TC), the CVT, and the vehicle dynamics [5]–[7].

Manuscript received April 21, 2011. This work was supported in part by the UEFISCDI.

C.-A. Dragoș, S. Preitl, R.-E. Precup, M.-B. Rădac and A.-I. Stînean are with the "Politehnica" University of Timisoara, Department of Automation and Applied Informatics, Bd. V. Parvan 2, RO-300223 Timisoara, Romania (phone: +40 2564032-29, -24; fax: +40 2564032-14; e-mail: claudia.dragos@aut.upt.ro, stefan.preitl@aut.upt.ro, radu.precup@aut.upt.ro, mircea.radac@aut.upt.ro, kassandra3107@yahoo.com).

E. M. Petriu is with the University of Ottawa, School of Information Technology and Engineering, 800 King Edward, Ottawa, ON, K1N 6N5 Canada (e-mail: petriu@site.uottawa.ca).

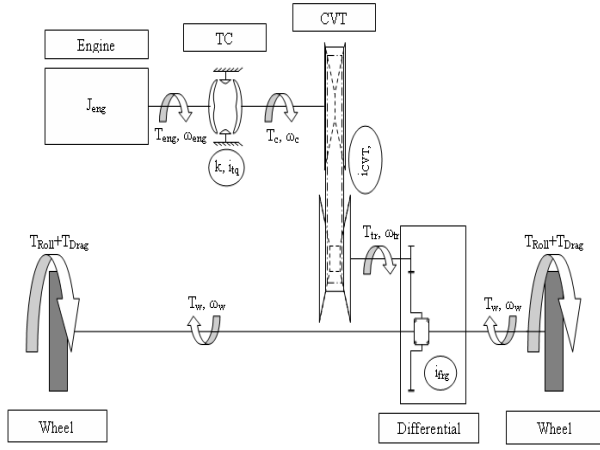


Fig. 1. The schematic structure of a vehicular power train system.

A. The Electronic Throttle

Since the acceleration pedal does not accept step modifications, to ensure a smooth motion it is necessary to introduce the electronic throttle to control the torque of the spark-ignition internal combustion engine. The dynamics of the electronic throttle is

$$\hat{throttle} = \frac{1}{1 + T_{t_p} s} \cdot \hat{TP}, \quad (1)$$

where $T_{t_p} = 0.03$ s is the time constant of the electronic

throttle, and $\hat{TP}$ [%] characterizes the normalized value of the desired throttle pedal position.

B. The Internal Combustion Engine

An internal combustion engine for a Honda Civic vehicle is considered to present our approach. The engine has the maximum power produced of 113 PS DIN at 6300 rpm and the maximum power torque of 143 Nm at 4800 rpm. The engine is characterized by its own moment of inertia $J_{eng} = 0.184$ kg m² and by the engine speed constraints

$$\frac{2\pi}{60} \cdot 1000 \leq \omega_{eng} \leq \frac{2\pi}{60} \cdot 6300. \quad (2)$$

The engine dynamics is modeled by [12], [3], [6]

$$J_{eng} \dot{\omega}_{eng} = T_{eng} - T_c, \quad (3)$$

where T_{eng} - the engine torque [N·m], T_c - the clutch torque [N·m], ω_{eng} - the engine speed [rad/s], and J_{eng} - the moment of inertia of the engine [kg m²].

C. The Torque Converter

A nonlinear MM of the torque converter uses the capacity factor/torque ratio k / i_{tq} , versus speed ratio steady-state curves Γ_2 and Γ_3 expressed as two look-up tables [8],

[9]:

$$T_c = \omega_{eng}^2 / k^2, \quad k = \Gamma_2(\omega_c, \omega_{eng}), \quad (4)$$

$$i_{tq} = \Gamma_3(\omega_c, \omega_{eng}).$$

where ω_c - the torque converter speed [rad/s].

D. The Continuously Variable Transmission

The CVT is characterized by an infinite number of effective gear ratios i_{CVT} between the minimum and maximum limits, $0.727 \leq i_{CVT} \leq 3.143$, to reduce the fuel consumption of spark-ignition internal combustion engine and the jerks associated to the manual and to the traditional automatic transmissions [10], [11]:

$$i_{CVT} = \Gamma_4(v_v), \quad T_{tr} = i_{CVT} \cdot i_{tq} \cdot T_c, \quad (5)$$

$$T_w = i_{fzg} \cdot T_{tr}, \quad \omega_c = i_{CVT} \cdot \omega_w.$$

where: Γ_4 - a function of vehicle speed v_v , T_{tr} - the transmission input torque [N·m], T_w - the transmission output torque [N·m], and i_{fzg} - the differential ratio.

E. The Vehicle Dynamics

The vehicle is a compact hatchback with the equivalent rotational moment of inertia $J_{veh} = r_w^2 m_v$ [kg m²]. The modification of the moment of inertia J_{veh} , as parametric disturbance caused by the modifications of the vehicle's mass m_v , according to the additional weight $0 \leq \Delta m_v \leq 431$ kg affects directly the vehicle performance in terms of settling time to achieve a speed of 100 km/h. The expression of vehicle's mass is

$$m_v = m_0 + \Delta m_v \quad (6)$$

where $m_0 = 1024$ kg is the vehicle mass without the passengers and luggage, and Δm_v is the additional mass.

The wheels are of the most common size, of 15" with 185/85 tires. Therefore the dynamics of the vehicle becomes [3], [6]

$$J_{veh} \dot{\omega}_w = T_w - T_{Drag} - T_{Roll} - T_{rez}, \quad (7)$$

$$T_{Drag} = 0.5 c A \rho r_w^3 \text{sgn}(\omega_w),$$

$$T_{Roll} = c_{Roll} m_v g r_w,$$

$$v_v = 3.6 r_w \omega_w,$$

where T_{Drag} - the aerodynamic resistance torque [N·m], T_{Roll} - the friction torque [N·m], c - the drag coefficient, A - the reference area, ρ - the density of the fluid, c_{Roll} - the rolling resistance, r_w - the tire radius, and ω_w - the wheel

speed [rad/s].

On the basis of partial MM of the components of vehicular power train system the following nonlinear mathematical model (NL-MM) is determined to develop the proposed control solutions:

$$\begin{aligned}
\dot{x}_1 &= -\frac{1}{T_{t_p}} \cdot x_1 + \frac{1}{T_{t_p}} \cdot u \\
\dot{x}_2 &= \frac{1}{J_{eng}} \cdot \left[T_{max} - \frac{T_{max} - T_p}{(\omega_p - \omega_M)^2} \cdot (x_2 - \omega_M)^2 \right] \cdot x_1 - \\
&\quad - \frac{1}{J_{eng}} \cdot \frac{x_2^4}{3.6^2 \cdot k^2 \cdot k_f^2 \cdot x_3^2 \cdot k_v^2 \cdot r_w^2 \cdot x_3^2}, \\
\dot{x}_3 &= \frac{1}{J_{veh}} \cdot \frac{i_{FRG} \cdot k_{tq} \cdot k_t \cdot x_2^3}{k^2 \cdot k_f^2 \cdot x_3} - \\
&\quad - \frac{1}{J_{veh}} (0.5 \cdot \rho \cdot A \cdot c_{air} \cdot r_w^3 \cdot x_3^2 - T_{Roll} - T_{rez}), \\
y &= 3.6 \cdot 0.01 \cdot r_w \cdot x_3,
\end{aligned} \tag{8}$$

where the characteristic variables are the control signal (input variable) $u = \hat{TP}$ (numerically equal to voltage), the state variables $x_1 = throttle$, $x_2 = \omega_{eng}$ and $x_3 = \omega_w$, and the controlled output (output variable) y [km/h].

Due to the nonlinearities of the system, the nonlinear model (1) is linearized at several operating points and the following linearized state-space model (Ln-MM) is obtained:

$$\begin{aligned}
\Delta \dot{\underline{x}} &= \underline{A} \Delta \underline{x} + \underline{b} \Delta u, \\
\Delta y &= \underline{c}^T \Delta \underline{x}, \\
\underline{A} &= \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad \underline{b} = \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \end{bmatrix}, \\
\underline{c}^T &= [c_{11} \quad c_{12} \quad c_{13}],
\end{aligned} \tag{9}$$

where the elements of the matrices $\underline{A}$, $\underline{b}$ and $\underline{c}^T$ are

$$\begin{aligned}
a_{11} &= -33.3, \quad a_{12} = 0, \quad a_{13} = 0, \\
a_{21} &= -\frac{1}{J_{eng}} \left(T_{max} - \frac{T_{max} - T_p}{(\omega_p - \omega_M)^2} (x_{20} - \omega_M)^2 \right) \\
a_{22} &= -\frac{1}{J_{eng}} \left(\frac{T_{max} - T_p}{(\omega_p - \omega_M)^2} (2x_{20} - 2\omega_M)x_{10} \right) - \\
&\quad - \frac{1}{J_{eng}} \left(\frac{4x_{20}^3}{3.6^2 \cdot k^2 \cdot k_f^2 \cdot k_v^2 \cdot r_w^2 \cdot x_{30}^4} \right),
\end{aligned} \tag{10}$$

$$\begin{aligned}
a_{23} &= \frac{1}{J_{eng}} \cdot \frac{4x_{20}^4}{3.6^2 \cdot k^2 \cdot k_f^2 \cdot k_v^2 \cdot r_w^2 \cdot x_{30}^5}, \\
a_{31} &= 0, \quad a_{32} = \frac{1}{J_{veh}} \left(\frac{i_{FRG} k_t k_{tq}}{k^2 k_f^2} \cdot \frac{3x_{20}^2}{x_{30}} \right), \\
a_{33} &= -\frac{1}{J_{veh}} \left(\frac{i_{FRG} k_t k_{tq}}{k^2 k_f^2} \cdot \frac{x_{20}^3}{x_{30}^2} + c_p A r_w^3 x_{30}^3 \right), \\
b_{11} &= 33.3, \quad b_{21} = 0, \quad b_{31} = 0, \\
c_{11} &= 0, \quad c_{12} = 0, \quad c_{13} = 3.6 \cdot 0.01 r_w.
\end{aligned} \tag{11}$$

The obtained transfer functions (t.f.s) of the system for all operating points

$$H_p^{(i)}(s) = \frac{y_m(s)}{u(s)} = \frac{k_p^{(i)}}{(1 + T_1^{(i)}s)(1 + T_2^{(i)}s)(1 + T_3^{(i)}s)} \tag{12}$$

are presented in Table I, where $i \in \{1, 2, 3\}$ is the index of the operating points.

TABLE I.
TRANSFER FUNCTIONS AT ALL OPERATING POINTS

Operating points	Transfer functions $H_p(s)$
$u_{01}=5$	$\frac{1.47}{(1+15.13 \cdot s)(1+0.3 \cdot s)(1+0.03 \cdot s)}$
$u_{02}=7$	$\frac{1.65}{(1+11.8 \cdot s)(1+0.23 \cdot s)(1+0.03 \cdot s)}$
$u_{03}=9$	$\frac{1.75}{(1+9.82 \cdot s)(1+0.3 \cdot s)(1+0.03 \cdot s)}$

III. CONTROL SOLUTIONS

Three LCCSs are developed for the vehicular power train systems with CVT, a PI controller, a PID controller and a TS-FC. On the basis of the Modulus Optimum method [12] two control solutions with PI(D) controllers are designed using linearized MMs for three operating points and the pole-zero cancellation technique. Our solution switches three algorithms developed separately for each of the three linearized MMs.

The use of PI and PID controllers with the t.f.s (13) and (14) respectively:

$$H_c^i(s) = \frac{k_c^i}{s} (1 + T_c^i s), \quad i = \overline{1, 3}, \tag{13}$$

$$H_c^i(s) = \frac{k_c^i}{s} (1 + T_{c1}^i s)(1 + T_{c2}^i s), \quad i = \overline{1, 3}, \tag{14}$$

is justified by the following ideas:

- They are widely accepted and successfully used in industry [12].

- Advanced and more efficient CSs can be developed on the basis of their structure. The PI(D) controllers can represent a support for comparisons with other developed CSs [13].
- The control principle of PI(D) controllers can also be extended to other control methods such as robust control, adaptive control, predictive control and fuzzy control). The equivalence between 1-DOF controllers extended with reference/feedback filters and linear or fuzzy two-degree-of-freedom (2-DOF) controllers can also be established.

The tuning parameters of the designed PI and PID controllers developed for the model (8) linearized at three operating points and with respect to the modifications of the vehicle's moment of inertia, are synthesized in Table II and in Table III, respectively.

TABLE II.
TUNING PARAMETERS OF PI CONTROLLERS

$u_0 / \Delta J_{veh}$	65.875		78.375		92.813	
	k_C^1	T_c^1	k_C^2	T_c^2	k_C^3	T_c^3
$u_{01}=5$	1.01	15	1.015	18.6	1.02	23.24
$u_{02}=7$	1.147	11.8	1.15	14.48	1.285	14.38
$u_{03}=9$	1.27	9.83	1.28	11.9	1.156	17.67

Due to the flexibility with respect to operating point modifications and to the parametric disturbances, TS-FCs with the integration of controllers' output are developed in [14]. The design of the TS-FCs starts with the design and tuning of continuous-time PI controllers which are next discretized resulting in quasi-continuous digital controllers. The discretized PI controllers are integrated in the consequents of the rules of the TS-FCs.

TABLE III.
TUNING PARAMETERS OF PID CONTROLLERS

$u_0 / \Delta J_{veh}$	65.875			78.375			92.813		
	k_C^1	T_{c1}^1	T_{c2}^1	k_C^2	T_{c1}^2	T_{c2}^2	k_C^3	T_{c1}^3	T_{c2}^3
$u_{01}=5$	11.26	15	0.3046	11.33	18.6	0.305	11.42	23.24	0.3055
$u_{02}=7$	10.07	11.8	0.2335	10.12	14.48	0.2338	10.17	17.67	0.234
$u_{03}=9$	9.489	9.83	0.193	9.52	11.9	0.193	9.56	14.38	0.193

The incremental versions of the recurrent equations of the discrete-time PI controller are

$$\Delta u_k^i = k_P^i (\Delta e_k + \alpha e_k), \quad (15)$$

$$k_P^i = k_C^i \left(1 - \frac{T_s}{2T_i^i}\right), \quad k_I^i = \frac{k_C^i T_s}{T_i^i}, \quad \alpha = \frac{k_I^i}{k_P^i},$$

where: i - the index of the controller used in the consequents of the rules, T_s - the sampling period, e_k - the control error, Δe_k - the increment of control error, and Δu_k^i - the increment of control signal of the rule i .

The following rules are used in the rule base of TS-FC:

$$\begin{aligned} &\text{IF}(e_k \text{ IS } \text{LTE}^i \text{ AND } \Delta e_k \text{ IS } \text{LTDE}^i) \\ &\text{THEN } \Delta u_k = \Delta u_k^i. \end{aligned} \quad (16)$$

Three linguistic terms with triangular membership functions $\text{LTE}^i, \text{LTDE}^i \in \{N, ZE, P\}$ are used for the two input linguistic variables e_k and Δe_k of the TS-FC. The parameters of the TS-FC with output integration are tuned according to the modal equivalence principle [15]. The TS-FC uses the MAX and MIN operators in the inference engine and the weighted average method for defuzzification.

All LCCSs are designed and adapted in order to fulfill

good tracking performance and to ensure good control system performance for a vehicle system with CVT and parametric disturbances. The complexity of the TS-FCs can be reduced using adequate techniques [16], [17].

IV. SENSITIVITY ANALYSIS

The sensitivity analysis with respect to the parametric disturbances of the controlled plant will be performed as follows. These parametric disturbances are represented by the variation of vehicle's moment of inertia J_{veh} that depend on vehicle's mass changes according to (6), so the expression of the parametric disturbances is

$$J_{veh} = J_{veh0} + \Delta J_{veh}. \quad (17)$$

The sensitivity analysis is performed in the frequency domain and presented here just for control solutions with PI controllers. The maximum value M_{s0} of sensitivity function's magnitude $S_0(j\omega)$, referred to as modulus margin, and its inverse M_{s0}^{-1}

$$M_{s0} = |S_0(j\omega)|_{\max}, \quad (18)$$

$$M_{s_0}^{-1} = \left| 1 + H_{OL}(z^{-1}) \right|_{\min} = \left| S_0^{-1}(z^{-1}) \right|_{\min} \\ = \left(\left| S_0(z^{-1}) \right|_{\max} \right)^{-1} = \left(\left| \frac{1}{1 + H_C(z^{-1})H_P(z^{-1})} \right|_{\max} \right)^{-1}, \quad (19)$$

are involved in the characterization of closed-loop system's robustness.

The Nyquist plots of the open-loop system, $h_+\{H_{OL}(j\omega)\}$, are calculated and plotted. The modulus margin defines the radius of the circle centered in critical point $(-1, j0)$ and tangent to $h_+\{H_{OL}(j\omega)\}$. Typical values of the modulus margin that ensure a good robustness are $M_{s_0}^{-1} \geq 0.5$ (-6 dB).

Table III illustrates the maximum sensitivity values and their inverses for the CSs with PI controllers designed for the Ln-MM (9) at three operating points with respect to the throttle pedal changes and with respect to the three cases of vehicle mass changes reflected in J_{veh} . In all nine cases the modulus margins are larger than 0.5 and the maximum value $M_{s_0}^{-1} = 0.67$ is obtained for the position throttle pedal at the 90% and for vehicle weight, only with the driver, without passengers and luggage.

TABLE IV.
MAXIMUM SENSITIVITY VALUES AND THEIR INVERSES

$u_0 / \Delta J_{veh}$		M_{s_0} and $M_{s_0}^{-1}$		
		65.875	78.375	92.813
5	M_{s_0}	1.912	1.932	1.941
	$M_{s_0}^{-1}$	0.523	0.5175	0.515
7	M_{s_0}	1.600	1.613	1.618
	$M_{s_0}^{-1}$	0.625	0.620	0.618
9	M_{s_0}	1.492	1.498	1.503
	$M_{s_0}^{-1}$	0.670	0.667	0.665

The Nyquist plots and the $M_{s_0}(u_0, J_{veh})$ circles were plotted for all nine cases marked with bold characters in Table IV. Fig. 2 shows the calculated Nyquist plots and the $M_{s_0}(u_0, J_{veh})$ circles only for the values of u_0 with bold characters and for the first value of J_{veh} in Table IV.

V. DIGITAL SIMULATION RESULTS

The designed control systems were tested with respect to:

- the position changes of the reference input, i.e., the throttle pedal,
- the parametric disturbances.

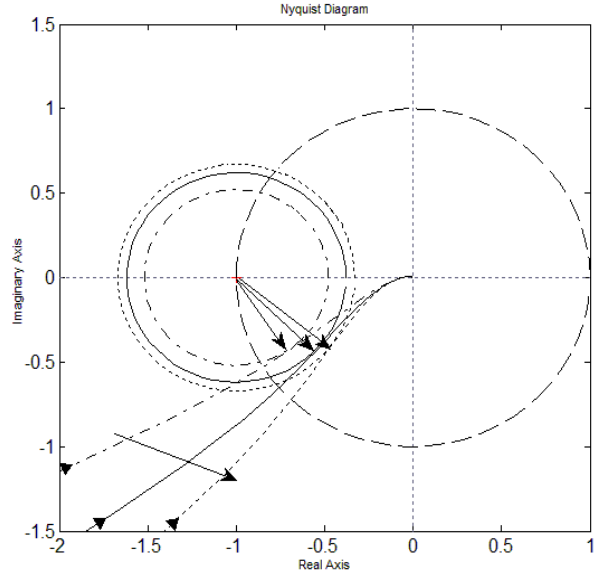


Fig. 2. Nyquist plots and $M_{s_0}(u_0, J_{veh})$ circles.

A part of the digital simulation results is presented in this section. The simulations point out the behavior of measured speeds with the proposed controllers for the vehicle power train system with spark-ignition internal combustion engine and CVT.

The simulation scenarios are characterized by:

- Three PI(D) controllers were designed for each Ln-MM with respect to the reference input changes and each of them were again tested for each parametric disturbances. The digital simulation results of the control system with PI(D) controllers designed for the nonlinear plant linearized around the throttle pedal position of 90% and for three vehicle mass changes are illustrated in Fig. 3 and Fig. 4, respectively.
- The TS-FC controller was designed for the Ln-MM with respect to the reference input changes, and it was tested for each parametric disturbance. The digital simulation results of the control system with TS-FC are presented in Fig. 5. They illustrate good behaviors as expected for similar control system structures [18]–[21].

VI. CONCLUSION

This paper has outlined the derivation of models for the vehicular power train system with CVT, the development of LCCSs assisted by sensitivity analysis. The modeling was conducted considering a simplification which allows the design of control solutions and the further extension to obtain improved control system performance.

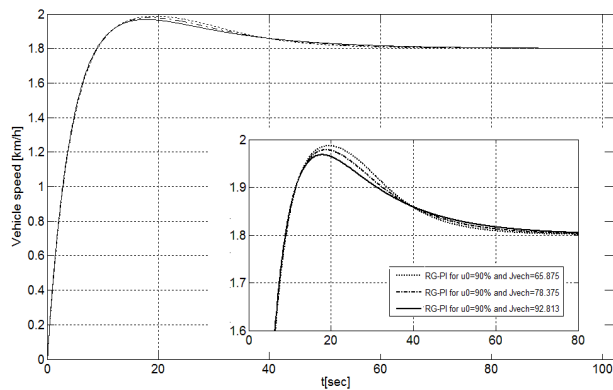


Fig. 3. Digital simulation results of control system with PI controller designed each vehicle mass changes and for throttle pedal position at 90%.

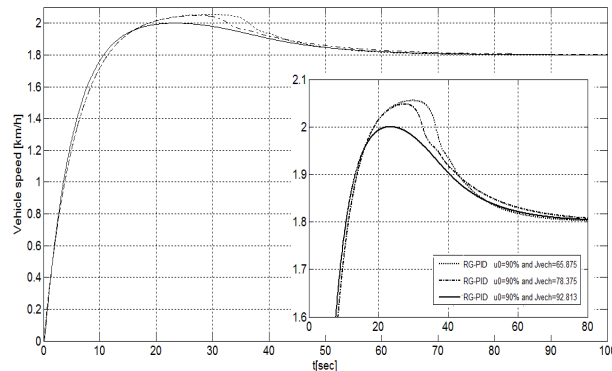


Fig. 4. Digital simulation results of control system with PID controller designed for each vehicle mass changes and for throttle pedal position at 90%.

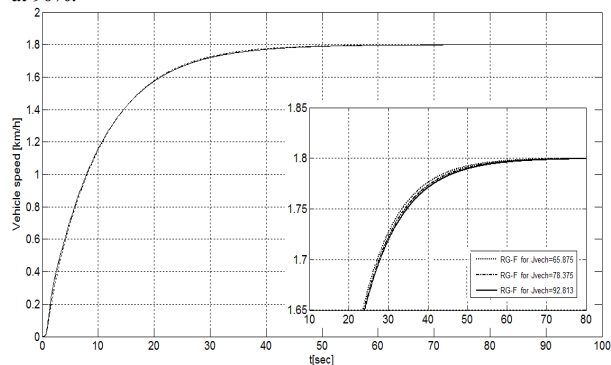


Fig. 5. Digital simulation results of control system with TS-FC designed for each vehicle mass changes.

The digital simulations results prove that our LCCSs ensure good control system performance with respect to the modifications of the reference input and to a category of parametric disturbances.

ACKNOWLEDGMENT

This work was supported by CNCSIS (IDEI 940/2008) and UEFISCDI (SICONA 12100/2008) and by the strategic grant POSDRU 6/1.5/S/13 (2008) of the Ministry of Labor, Family and Social Protection, Romania, co-financed by the European Social Fund – Investing in People.

REFERENCES

- [1] H.-H. Erbe, "Introduction to low cost/cost effective automation," *Robotica*, vol. 21, pp. 219–221, June 2003.
- [2] R. Isermann, *Mechatronic Systems: Fundamentals*. Berlin, Heidelberg, New York: Springer-Verlag, 2005.
- [3] U. Kiencke and L. Nielsen, *Automotive Control Systems for Engine, Driveline and Vehicle*, 2nd ed. Berlin, Heidelberg, New York: Springer-Verlag, 2005.
- [4] K. Tanaka and H. O. Wang, *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*. New York: John Wiley & Sons, 2001.
- [5] L. Guzzella and A. M. Schmid, "Feedback linearization of spark-ignition engines with continuously variable transmissions," *IEEE Trans. Control Syst. Tech.*, vol. 3, pp. 54–60, March 1995.
- [6] C. Lazăr et al., "Real-time informatics technologies for embedded-system-control of power-train in automotive design and applications," Research Report 2 of the SICONA CNMP Grant, "Gh. Asachi" Tech. Univ. Iași, Iași, Romania, 2010 (in Romanian).
- [7] J. Deur, J. Petri, J. Asgari, and D. Hrovat, "Recent advances in control-oriented modeling of automotive power train dynamics," *IEEE/ASME Trans. Mechatronics*, vol. 11, pp. 513–523, Oct. 2006.
- [8] C.-A. Dragoș, S. Preitl, R.-E. Precup, C.-S. Nes, D. Pirlea, and A. S. Paul, "Control Solutions for Vehicles with Continuously Variable Transmission," in *Proc. 11th IEEE Intl. Symp. Computational Intelligence and Informatics (CINTI 2010)*, Budapest, Hungary, 2010, pp. 157–162.
- [9] P. Setlur, J. R. Wagner, D. M. Dawson, and B. Samuels, "Nonlinear control of a continuously variable transmission (CVT)," *IEEE Trans. Contr. Syst. Tech.*, vol. 11, 2003, pp. 101–108.
- [10] M. Zhou, X. Wang, and Y. Zhou, "Modeling and simulation of continuously variable transmission for passenger car," in *Proc. 1st Intl. Forum Strategic Technology*, Ulsan, Korea, 2006, pp. 100–103.
- [11] J. Song and C. Wang, "Modeling and simulation of hydraulic control system for vehicle continuously variable transmission," in *Proc. 3rd IEEE Conf. Industrial Electronics and Applications (ICIEA 2008)*, Singapore, 2008, pp. 799–803.
- [12] K.-J. Åström and T. Hägglund, *PID Controllers Theory: Design and Tuning*. Research Triangle Park: Instrument Society of America, 1995.
- [13] J. Quevedo and T. Escobet, "Past, present and future of PID control (PID'00)," in *Prep. IFAC Workshop Digital Control*, Terrassa, Spain, 2000, pp. 5–7.
- [14] S. Preitl, Z. Preitl, and R.-E. Precup, "Low cost fuzzy controllers for classes of second-order systems," in *Proc. 15th World Congress of IFAC (b'02)*, Barcelona, Spain, 2002, paper index 416, 6 pp.
- [15] S. Galichet and L. Foulloy, "Fuzzy controllers: synthesis and equivalences," *IEEE Trans. Fuzzy Syst.*, vol. 3, pp. 140–148, May 1995.
- [16] P. Baranyi, K. F. Lei, and Y. Yam, "Complexity reduction of singleton based neuro-fuzzy algorithm," in *Proc. IEEE Intl. Conf. Systems, Man, and Cybernetics (SMC'00)*, Nashville, TN, USA, 2000, vol. 4, pp. 2503–2508.
- [17] P. Baranyi, D. Tikk, Y. Yam, and R. J. Patton, "From differential equations to PDC controller design via numerical transformation," *Comp. Ind.*, vol. 51, pp. 281–297, Aug. 2003.
- [18] D. H. Owens and J. Hätonen, "Iterative learning control - An optimization paradigm," *Annu. Rev. Control*, vol. 29, pp. 57–70, 2005.
- [19] Z. C. Johanyák and S. Kovács, "Fuzzy rule interpolation based on polar cuts," in *Computational Intelligence, Theory and Applications*, B. Reusch, Ed. Berlin, Heidelberg, New York: Springer-Verlag, 2006, pp. 499–511.
- [20] D. Hladek, J. Vaščák, and P. Sinčák, "Multi-robot control system for pursuit-evasion problem," *J. Electr. Eng.*, vol. 60, pp. 143–148, June 2009.
- [21] S. Blažič, D. Matko, and I. Škrjanc, "Adaptive law with a new leakage term," *IET Control Theory Appl.*, vol. 4, pp. 1533–1542, Sept. 2010.

Low Cost Embedded Solution for BLDC Motor Control

Radu Duma, Petru Dobra, Mirela Dobra and Ioan Valentin Sita

Abstract— The paper presents a low cost embedded solution for brushless direct current (BLDC) motor control. An embedded control algorithm for BLDC motor control is presented and the minimal hardware requirements that a microcontroller must have in order to be used for BLDC motor control are described. The advantages with respect to existing control solutions are presented. Using the described approach can be easily validate if a certain microcontroller is suited for BLDC motor control. The pseudo code of the BLDC motor control algorithm is described. A practical application using a Renesas M32C87 microcontroller is presented. A PID controller with anti-windup is implemented and tuned for a BLDC motor.

I. INTRODUCTION

BRUSHLESS direct current motors (BLDC) are a type of DC motors which do not use brushes for commutation but they are electronically commutated. Since BLDC motors do not use brushes they have longer operating life and noiseless operation. For BLDC motors the ratio between the torque and the size of the motor is high. This feature makes BLDC motors to be suited for applications where space and weight are critical. Some other characteristics of BLDC motor are: higher speed ranges, better speed versus torque characteristics and high dynamic response. BLDC motors are used in several industry fields: appliances, automotive, aerospace, consumer, medical, industrial automation equipment and instrumentation.

For BLDC motor control there are analog solutions, integrated solution and digital control solutions. Most integrated circuits (IC) producers offer analog ICs for BLDC motor control [1, 2]. There are also integrated control solutions which allow the customization of the control algorithm from a PC, using a graphical user interface [3]. The highest flexibility can be achieved by implementing the control algorithm on a microcontroller or

a Digital Signal Controller (DSC). Companies like Texas Instruments, Microchip and Renesas offer BLDC motor control evaluation kits based on a microcontroller or a DSC.

The analog control solution is a low cost one, but with a reduce degree of flexibility. The parameters of the control algorithm can be changed by external electronic components. In order to obtain high flexibility at low cost, a digital control solution based on a cheap microcontroller must be used. The purpose of this paper is to present a control algorithm which can be implemented on a low cost microcontroller, and to describe the minimal hardware requirements for a microcontroller or a DSP, in order to be used for BLDC motor control. A detailed description of the control algorithm is presented. Also a PID algorithm with anti-windup and the equations required for the implementation of this algorithm are described.

Texas Instruments, Renesas and Microchip described digital control solutions tailored for their processors [4, 5, 6]. This paper presents a general control solution which can be used for any microcontroller which has the minimal required hardware peripherals.

II. HARDWARE TOPOLOGY

The commutation of the BLDC motor is done electronically. Thus it is important to know the position of the rotor. For this reason BLDC motors have Hall sensors embedded in the stator. The Hall sensors generate a high or low signal when the N or S rotor magnetic pole passes near the sensor. Based on the sequence of the signals coming from the Hall sensors the producer defines the commutation sequence. According to the commutation sequence certain windings of the motor have to be controlled. For each motor the producer defines the commutation sequence.

In Fig. 1 is presented the basic inverter topology for driving three-phase BLDC motors.

The control electronics from Fig. 1 is the microcontroller used to control the three-phase bridge. This paper defines the minimal characteristics of a microcontroller in order to be used as control electronics for the control of BLDC motors.

There are various modulation methods for controlling inverter drives: upper modulation, lower modulation, rotating modulation, or balanced modulation. In this paper upper modulation is presented, because it is most suited

Manuscript received August 23, 2011.

Radu Duma is with Automatic Control Department, Technical University of Cluj-Napoca, str. Memorandumului, 400114 Romania. (phone: +40-264-401433, e-mail: radu.duma@aut.utcluj.ro).

Petru Dobra is with Automatic Control Department, Technical University of Cluj-Napoca, str. Memorandumului, 400114 Romania. (phone: +40-264-401433, e-mail: dobra@aut.utcluj.ro).

Mirela Dobra is with Automatic Control Department, Technical University of Cluj-Napoca, str. Memorandumului, 400114 Romania. (phone: +40-264-401433, e-mail: mirela.trusca@aut.utcluj.ro).

Ioan Valentin Sita is with Automatic Control Department, Technical University of Cluj-Napoca, str. Memorandumului, 400114 Romania. (phone: +40-264-401433, e-mail: valentin.sita@aut.utcluj.ro).

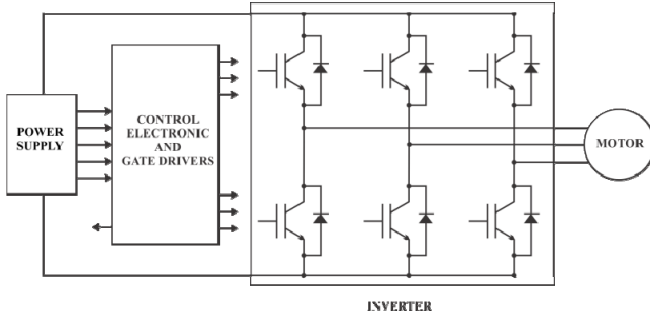


Fig. 1. Inverter topology.

for implementation on a low cost microcontroller. Using upper modulation technique, only the upper transistors of the three phase bridge are controlled using Pulse Width Modulation (PWM) signals, while for the lower transistors enable/disable signals are used.

In Fig. 2 is presented an example of upper modulation waveforms and the signature of the Hall sensors. In this figure H_W , H_V , H_U represent the signals generated by the Hall sensors, corresponding to the three phases of the motor U , V , W respectively; U_P , V_P and W_P represent the upper transistors of the three phase bridge corresponding to the three phases of the motor U , V , W respectively; while U_N , V_N and W_N represent the lower transistors of the three phase bridge corresponding to the three phases of the motor U , V , W respectively.

III. CLOSED LOOP CONTROL

The block diagram of a closed loop control system for BLDC motor control is presented in Fig. 3.

A PWM timer or module of the controller is used to generate three PWM signals for controlling the upper arm transistors of the three phase bridge, while three General Purpose Input Output (GPIO) pins are used to generate the enable/disable signals for the lower arm transistors of the three phase bridge.

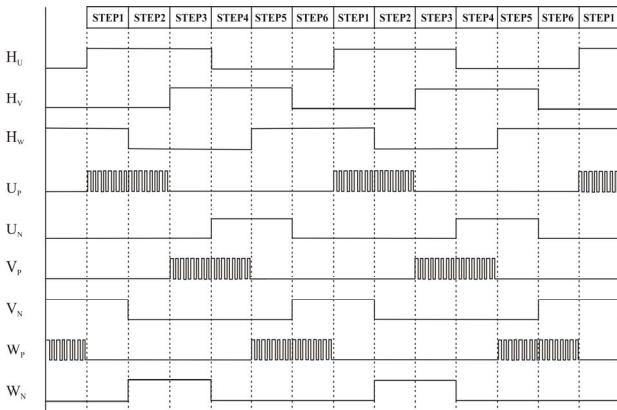


Fig. 2. Upper modulation technique.

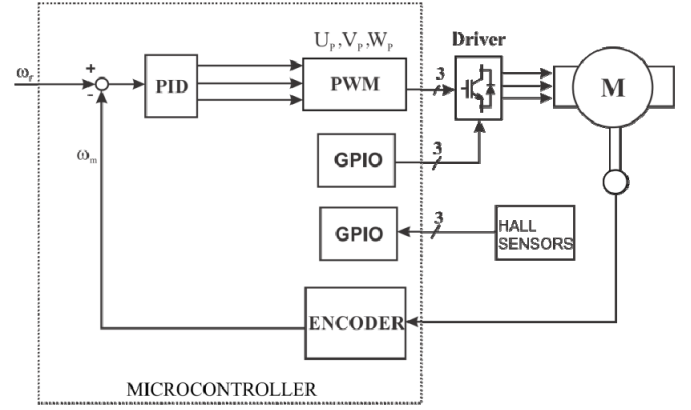


Fig. 3. Closed loop control diagram.

The speed of the motor can be determined using the signals from the Hall sensors. In this case a timer of the controller is used to count the clock cycles between two Hall transitions, and using the timer value the speed of the motor can be computed.

For applications which require a high accuracy of the measured speed an encoder must be used.

Based on the reference and measured speed, the PID controller computes the duty cycle for the PWM control signals.

IV. HARDWARE REQUIREMENTS

Taking into consideration the aspects presented in the previous sections, in this section the minimal hardware peripherals requirements for a microcontroller are defined.

A. Control Signals

In order to control the three-phase bridge using the upper modulation technique, the microcontroller must be able to generate three PWM signals. Thus the microcontroller must have a timer or a PWM module capable of generating three PWM signals. Also the upper modulation technique requires three enable/disable signals which can be generated using three GPIO pins.

Based on the commutation sequence of the BLDC motor, a certain PWM signal and a GPIO pin must be used to control the three-phase bridge.

B. Feedback Signals

The BLDC motor must be commutated electronically, thus it is important to know the position of the rotor. For this reason HALL sensors are embedded in the stator. The signals generated by the HALL sensors have to be applied to three external hardware interrupts pins, such that the microcontroller can commutate the motor. Thus another critical requirement is that the microcontroller has three hardware interrupts pins.

When the signal generated by the HALL sensor changes its state from low to high or from high to low, the hardware interrupt routine is triggered, the speed of the motor is

computed using a timer (if an encoder is no used for speed measurement), and the motor is electronically commutated.

If an encoder is used for speed measurement, beside the three external interrupts pins, two timer input pins are required for the signals generated by the encoder.

An important feature which must be implemented by every embedded control system, used for motor control, is the over current protection. For this an analog to digital converter or an analog comparator must be used. If an over current condition is detected, the motor must be stopped, by disabling the PWM signals.

V. SOFTWARE ARCHITECTURE

In this chapter the software architecture of the embedded control system is presented in detail. The system initialization, the interrupt routines and the commutation principle of the motor are described.

In order to avoid damages to the equipments which are driven by the motor, at power up, a slow start-up of the motor must be implemented. For a short period of time the duty cycle of the PWM signals is gradually incremented until the value of the duty cycle becomes higher than the value of the duty cycle calculated by the PID controller.

Also for proper operation, it is necessary to ensure that the high-side bootstrap capacitors of the driver are charged during initial start-up.

Taking into consideration the number of signals required for implementing a closed loop control algorithm for a BLDC motor the microcontroller must have at least eighteen pins.

A. System initialization

In this subsection the steps required for the initialization of the system are presented. The three PWM signals must be initialized; the switching frequency is usually in the range 10-20 kHz. Three GPIO pins must be configured as outputs for the generation of three enable/disable signals. In order to commutate the motor the signals generated by the HALL sensors must be applied to three external interrupts pins. For speed measurement the signals from the HALL sensors or an encoder can be used. Independently of the method used for speed measurement a timer must be configured. The pseudo code listed below summarizes the steps presented above.

```
initialize_three_PWM_signals;
```

```
configure_three_pins_as_output_pins;
```

```
configure_three_external_interrupts_for_HALL_sensors;
```

```
configure_timer_for_speed_measurement;
```

B. Interrupts

An embedded control system uses for ensuring real-time

constraints the interrupts of the microcontroller. For controlling a BLDC motor are required an asynchronous interrupt for motor commutation and a synchronous interrupt for PID controller computation.

HALL interrupt. The signals generated by the HALL sensors are asynchronous signals. Based on the sequence of the signals from the HALL sensors the corresponding transistors have to be controlled. These signals have to be applied on the hardware interrupts input pins of the used microcontroller. The interrupts must be configured to be triggered on both falling and rising edges of the signal generated by the HALL sensors. For the three signals the same interrupt routine must be triggered. In the associated interrupt routine the timer value is read, the speed of the motor is computed, the signature of the HALL sensors is read and the motor is commutated. The pseudo code for the interrupt service routine is presented below:

```
interrupt_routine
```

```
    readTimer();
```

```
    computeSpeed();
```

```
    readHardwareInterruptPins();
```

```
    comutateMotor();
```

```
end;
```

Synchronous interrupt. The synchronous interrupt can be triggered by the timer or by the PWM module. The main task of this interrupt routine is to perform the computation of the PID algorithm. Other secondary task can be performed, like over current monitoring.

C. Motor commutation

BLDC motors are a type of DC motors which are electronically commutated, thus it is important to know the position of the rotor in order to know which windings of the motor have to be controlled. For each motor the producer defines the commutation sequence. For the Hurst-Emerson motor used in the practical application presented in this paper the sequence of commutation is presented in Table I.

According to Table I, when the signature of the HALL sensors is 3 (3 corresponds to binary value 011, that is $H_W = 0$, $H_V = 1$ and $H_U = 1$) the V_P and W_N transistors of the three phase inverter bridge have to be controlled. The commutation sequence can be implemented in software using a switch statement.

VI. PID CONTROLLER

In the following subsections a PID tuning method and a closed loop control algorithm with a PID controller are presented.

TABLE I
HURST-EMERSON MOTOR SEQUENCE OF COMMUTATION

Hall signature ($H_W H_V H_U$)	Active tranzistors	
1	U_P	W_N
2	V_P	U_N
3	V_P	W_N
4	W_P	V_N
5	U_P	V_N
6	W_P	U_N

A. PID Controller Implementation

The standard equation of a PID controller is presented below:

$$u(t) = K_p \left(\varepsilon(t) + \frac{1}{T_i} \int_0^t \varepsilon(\tau) d\tau + T_d \frac{d\varepsilon(t)}{dt} \right)$$

where $\varepsilon(t) = y_{ref}(t) - y(t)$ is the error signal, y_{ref} is the reference signal and y is the output of the process.

In time several modifications have been done to the standard PID controller structure. A pure derivative gives a very large amplification of measurement noise. The gain of the derivative must thus be limited, by low-pass filtering the derivative term, resulting in a limited gain N at high frequencies. A fraction β of the reference signal acts on the proportional part, while the derivative component acts only on the output of the process.

The structure of the PID controller with integrator anti-windup is shown in Fig. 4.

Taking into consideration the above remarks, the PID algorithm can be described by the following equations:

$$\begin{aligned} P(n) &= K_p (\beta y_{ref}(n) - y(n)) \\ D(n) &= \alpha_d D(n-1) + \beta_d (y(n-1) - y(n)) \\ v(n) &= P(n) + I(n) + D(n) \\ I(n+1) &= I(n) + \beta_i (y_{ref}(n) - y(n)) + \beta_t (u(n) - v(n)) \end{aligned} \quad (1)$$

where

$$\begin{aligned} \alpha_d &= T_d / (T_d + Nh), & \beta_d &= (K_p T_d N) / (T_d + Nh), \\ \beta_i &= K_p T_s / T_i & \beta_t &= T_s / T_i. \end{aligned}$$

Equations (1) were used for the implementation of the PID algorithm on the microcontroller using C language.

B. Controller Tuning

Chen and More [7] proposed for controller tuning a robust tuning method. The method is based on the condition that the phase Bode plot at a specified frequency w_c (referred to as “tangent frequency”) at the point where sensitivity circle touches Nyquist curve is

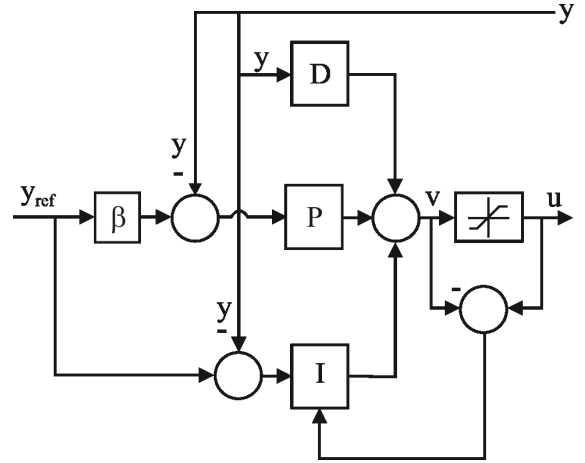


Fig. 4. The structure of the PID controller.

locally flat which implies that the system is more robust to gain variations. The paper presents a new tuning rule which gives a new relationship between T_i and T_d instead of the equation $T_i = 4T_d$, proposed in the modified Ziegler-Nichols methods [8, 9].

After the initial selection of the tangent frequency w_c and of the tangent phase γ_m using an iterative algorithm the phase $\angle P(jw_c)$ and the module $|P(jw_c)|$ of the process can be estimated. The derivative of the phase of the open loop system $s_p(w_c)$ can be approximated by Bode's Integral [10]:

$$s_p(w_c) \approx \angle P(jw_c) + \frac{2}{\pi} [\ln |K_g| - \ln |P(jw_c)|], \quad (2)$$

where $|K_g| = P(0)$ is the static gain of the process.

The equations for determining the parameters of the PID controller, as described in [7], are presented below:

$$\begin{aligned} K_p &= \frac{\cos(\gamma_m)}{|P(jw_c)| \sqrt{1 + \tan^2(\gamma_m - \angle P(jw_c))}} \\ T_i &= \frac{-2}{\omega_c \left[s_p(\omega_c) + \tan(\hat{\Phi}) + \tan^2(\hat{\Phi}) s_p(\omega_c) \right]} \\ T_d &= \frac{-T_i w_c + 2s_p(w_c) + \sqrt{\Delta}}{2s_p(w_c) w_c^2 T_i} \end{aligned} \quad (3)$$

where: $\hat{\Phi} = \gamma_m - \angle P(jw_c)$ and

$$\Delta = T_i^2 w_c^2 - 8s_p(w_c) T_i w_c - 4T_i^2 w_c^2 s_p^2(w_c)$$

In order to estimate the phase and the module of the process at the desired frequency a closed loop system with relay and time delay has to be implemented (Fig. 5).

The iterative algorithm described in detail [7] is

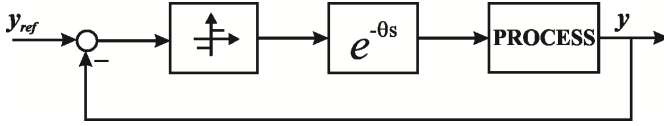


Fig. 5. Block diagram of a system with relay and dead time.

implemented in order to tune the parameters of the PID controller. The frequency of oscillation caused by the relay can be changed by introducing an artificial dead time on the direct path. Starting with the selected tangent frequency and using an iterative method, the dead time corresponding to the desired frequency is determined.

The selection of the value for w_c depends on the dynamic of the process for which the controller must be tuned. For the majority of the processes there is an interval for the selection of w_c in order to realize the flat phase condition. If this interval is not known the initial selection for w_c can be the cutoff frequency.

The desired frequency is computed using the relation: $\omega_n = 2\pi/T_u$, where T_u represent the ultimate period, which is determined from the limit cycle. The module of the process is estimated using the relation: $|P(j\omega_n)| = 1/K_u$, where K_u is the ultimate gain determined using (5). The phase is computed using the relation $\angle P(j\omega_n) = -\pi + \omega_n\theta$, where θ represents the dead time for which the desired frequency is obtained.

VII. APPLICATION EXAMPLE

The setup presented in Fig. 6 was used to test the approach proposed in this paper. A Renesas M32C87 processor member of the M16C microcontroller's family was used for BLDC motor control. The microcontroller has the required peripherals presented in this paper: two timers which can generate PWM signals, external hardware interrupts pins, up to eighty input pins, encoder interface. A Hurst-Emerson motor was used. On the shaft of the motor an Agilent HEDS-5500 encoder was mounted. The integrated power module IRAMS10UP60A was used as motor drive, which is optimized for electronic motor control.

For determining the parameters of the PID controller the algorithm described in the previous section is implemented.

The limit cycle corresponding to the desired frequency is presented in Fig. 7. It is transferred in real-time from the microcontroller using the CAN interface of the microcontroller and a USB to CAN converter on the PC side.

From the obtained limit cycle the phase and the module of the process are estimated and s_p is computed using (2).

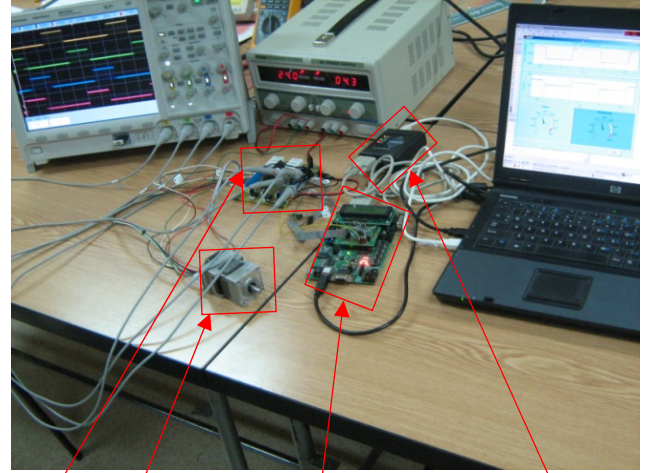


Fig. 6. Test setup.

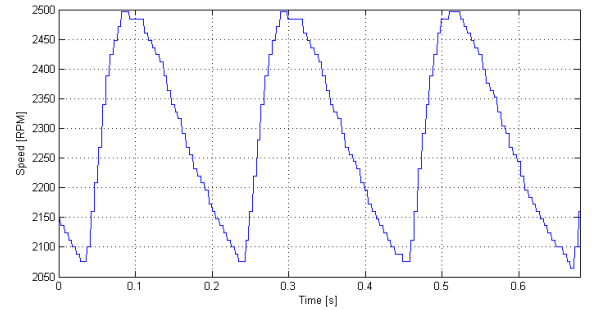


Fig. 7. Limit cycle of the BLDC motor.

The parameters of the controller are determined using (3).

Using the computed PID controller, a closed loop speed control algorithm for BLDC motor is implemented. In order to test the correctness of the commutation sequence the Hall sensors signature and phase current (Fig. 8) and the PWM control signals and an activation signal (Fig. 9) are captured using a digital oscilloscope.

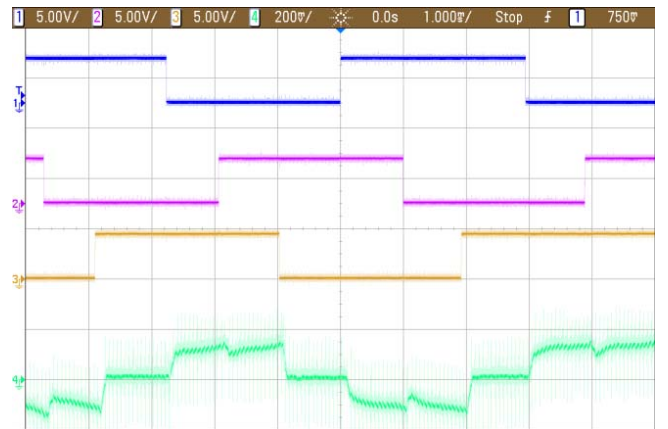


Fig. 8. Hall sensors signature and phase current.

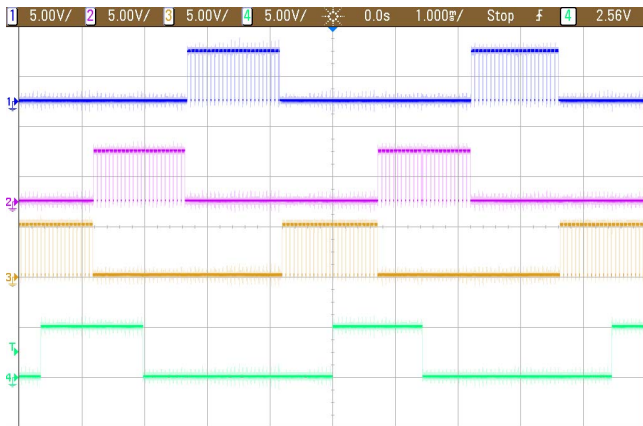


Fig. 9. PWM control signals and an activation signal.

In Fig. 10 is presented the step response of the BLDC motor to a rectangular reference signal, with the computed PID controller, and in Fig. 11 the duty cycle of the control PWM signal in percentages. On the Renesas M32C87 processor the CAN protocol was implemented and the data was retrieved using a USB to CAN converter.

The motor phase voltages during commutation are presented in Fig. 12. From Fig. 12 the six commutation steps can be observed.

VIII. CONCLUSION

The paper presented a control algorithm which can be implemented on a low cost microcontroller and described the minimal hardware requirements for a microcontroller or a DSC, in order to be used for BLDC motor control. A detailed description of the control algorithm was presented. A closed loop speed control with a PID controller was implemented.

Another contribution of the paper is the development of a general control solution which can be used for any microcontroller which has the minimal required hardware peripherals. The existing solutions on the market are tailored for a certain processor. The advantages with respect to existing control solutions are presented.

The practical application presented proves the effectiveness of the proposed method.

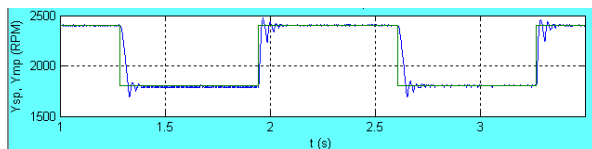


Fig. 10. Motor response with a PID controller.

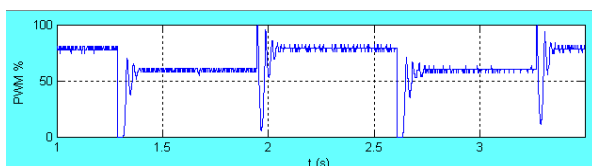


Fig. 11. PWM command signal.

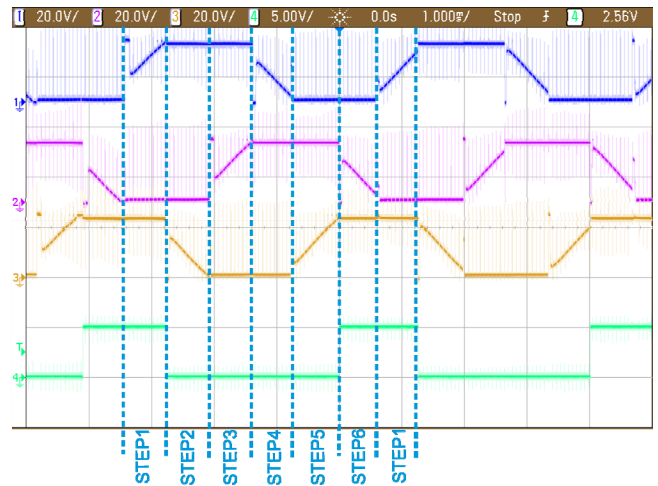


Fig. 12. Motor phase voltages during commutation.

ACKNOWLEDGMENT

This paper was supported by the project "Progress and development through post-doctoral research and innovation in engineering and applied sciences – PRiDE - Contract no. POSDRU/89/1.5/S/57083", project co-funded from European Social Fund through Sectorial Operational Program Human Resources 2007-2013.

REFERENCES

- [1] ***, "Motor drive system using SPM inverter," Fairchild, Semiconductor, Application Note 9019, 2001.
- [2] ***, "Full-featured power module for high voltage direct drive of 3-phase brushless dc motor," International Rectifier, DataSheet.
- [3] ***, "MC37110 Motor Control IC," Performance Motion Devices, MC37110 Motor Control Chip Datasheet, 2008.
- [4] ***, "Stellaris Brushless DC (BLDC) Motor Control Reference Design Kit with Ethernet and CAN," Texas Instruments, User Manual, 2009.
- [5] P. Yedamale, "Brushless DC Motor Control Using PIC18FXX31 MCUs," Application Note, 2004.
- [6] ***, "R8C/Tiny BLDC Motor Control Evaluation Kit (YMCRPR8C25)," Renesas, User's Manual, 2008.
- [7] Y. Chen and K. L. Moore, "Relay feedback tuning of robust PID controllers with iso-damping property," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 35, no. 1, pp. 23-31, 2005. [Online]. Available: <http://dx.doi.org/10.1109/TSMCB.2004.837950>
- [8] C. C. Hang, K. J. Astrom and W. K. Ho, "Refinements of the Ziegler-Nichols tuning formula," *IEEE Proceedings D Control Theory and Applications*, vol. 138, pp. 111-118, Mar. 1991.
- [9] T. Hagglund and K. J. Astrom. *PID Controllers: Theory, Design, and Tuning*. ISA -The Instrumentation, Systems, and Automation Society, 2nd ed., 1995.
- [10] A. Karimi, D. Garcia and R Longchamp, "Iterative controller tuning using Bode's integrals". In *Proceedings of the 41st IEEE Conference on Decision and Control*. pages 4227-4232, Las Vegas, Nevada, 2002.

Backstepping Control of Wheeled Mobile Robots

B. Dumitrascu, A. Filipescu, V. Minzu, A. Filipescu Jr.

Abstract— A trajectory tracking controller based on the backstepping technique for the PowerBot wheeled mobile robot using the kinematic model is proposed in this paper. The PowerBot is an automated differential drive guided vehicle specially designed and equipped for autonomous, intelligent delivery and handling of large payloads. The PowerBot is an ideal platform for laboratory and research tasks involving delivery, navigation and manipulation thanks to its top speed of 1.6 m/s and payload capacity of up to 100 kg. Simulations and real-time are used to validate the proposed controller and the results prove the effectiveness of the controller.

I. INTRODUCTION

WHEELED mobile robots are becoming a part of our daily lives due to the progress in technology that leads to more complex and also cheaper robots. Mobile robots were, at first, restricted to research or mostly military applications due to their high prices, but nowadays a lot of cheap robots are available that are affordable to the general consumer. Wheeled mobile robots are the most important type of mobile robots. They are easy to build reliable, cheaper than other types of mobile robots and can move along most surfaces. More and more household applications and industrial applications employ wheeled mobile robots. The industry uses them mainly to transport components from an assembly line to another, or from the assembly line to the storage and loading bay, but they are also used for various other tasks, like welding, painting or inspecting hard to reach areas or dangerous areas for a humans to enter, depending on the equipment attached to the wheeled mobile robot. Wheels allow robots to travel almost

anywhere in a household, the only unreachable area being the stairs. A lot of household wheeled mobile robots are already commercially available like autonomous cleaning robots or mowing robots, but we can expect a lot more household tasks to be performed by wheeled mobile robots as technology allows for cheaper robots. For this reasons research on wheeled mobile robots has been a very active field.

Wheeled mobile robots are characterized by nonholonomic systems, a class of systems that cannot be stabilized by smooth static state feedback laws [1].

Backstepping control is one of the most important methods for stabilizing a nonholonomic system. The control problem for nonlinear systems has been researched by Kolmanovsky et al. [5] and Slotine et al. [12].

Backstepping is a recursive procedure that breaks a design problem for the whole system into a sequence of design problems for lower order systems. Backstepping for nonlinear systems was researched by several researchers like Shi [2] and Khalil [11]. The method proved to be effective for the control problem of wheeled mobile robots [3], [4], [6], [8], [9], [10], [13].



Fig. 1. PowerBot wheeled mobile robot

In this paper a backstepping controller for the PowerBot (Fig. 1) wheeled mobile robot, from Adept Mobile Robots Inc. [14] is presented. The PowerBot is an automated differential drive guided vehicle specially designed and equipped for autonomous, intelligent delivery and handling of large payloads. It is a part of the MobileRobots Pioneer family of mobile robots, which are research and

This work of Bogdan Dumitrascu was supported by Project POSDRU/88/1.5/S/61445.

Bogdan Dumitrascu is PhD student in Control Systems with the Department of Automation and Electrical Engineering, University “Dunarea de Jos” of Galati, Domneasca 47, 800008, Galati, Romania (email: dumi_b20@yahoo.com).

Adrian Filipescu is prof. PhD supervisor with the Department of Automation and Electrical Engineering, University “Dunarea de Jos” of Galati, Romania (email: Adrian.Filipescu@ugal.ro).

Viorel Minzu is professor with the Department of Automation and Electrical Engineering, University “Dunarea de Jos” of Galati, Romania (e-mail: Viorel.Minzu@ugal.ro).

Adriana Filipescu Jr. is master student in Control Systems with the Department of Automation and Electrical Engineering, University “Dunarea de Jos” of Galati, (e-mail: adriana_filipescu@yahoo.ca).

development platforms that share a common architecture, foundation software, and employ artificial intelligence-based client-server robotics controls. The PowerBot has a top speed of 1.6 m/s and can carry a payload of up to 100 kg. It is an ideal platform for laboratory and research tasks involving delivery, navigation and manipulation.

The PowerBot has a sturdy aluminum body, balanced drive system (two-wheel differential with balancing casters), reversible DC motors, motor control and drive electronics, high resolution motion encoders, and a long-life battery power, all managed by an onboard controller and intelligent mobile-robot server software.

The BowerBot is controlled using an advanced interface for applications called ARIA. ARIA is released under GNU Public License and complete with fully documented C++ libraries and source code. ARIA is the ideal platform for integration of robot-control software, since it neatly handles the lowest-level details of client-server interactions, including serial communications, command and server-information packet processing, cycle timing, and multithreading.

II. PROBLEM FORMULATION

The model of the PowerBot wheeled mobile robot is presented in Fig. 2. The model takes into account the two diametrically opposed drive wheels of radius R , the distance between the wheels ($2L$), the angular speeds of the drive wheels (ω_L, ω_R), the center point (CP) of the robot. The pose of the robot is considered $P = (x_r, y_r, \theta_r)$, where x_r represents the position on the x axis, y_r the position on the y axis and θ_r the heading of the robot, v_r represents the linear velocity while ω_r represents the angular velocity of the robot.

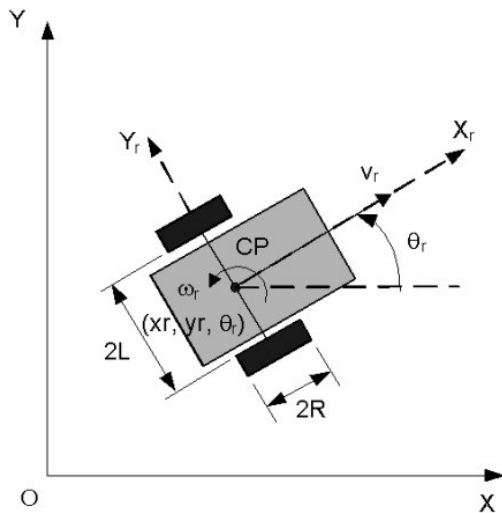


Fig. 2. Kinematic model of PowerBot

The kinematic model for the robot can be expressed as:

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} \cos \theta_r & 0 \\ \sin \theta_r & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_r \\ \omega_r \end{bmatrix}. \quad (1)$$

The kinematic model of the robot can also be expressed as:

$$\begin{cases} \dot{x}_r = v_r \cdot \cos \theta_r \\ \dot{y}_r = v_r \cdot \sin \theta_r \\ \dot{\theta}_r = \omega_r \end{cases} \quad (2)$$

In order to solve the trajectory tracking problem a controller, capable of calculating the inputs v, ω so that the robot tracks a desired trajectory, has to be designed. For this purpose a virtual robot, with the desired trajectory $q_d(t) = [x_d(t) \ y_d(t) \ \theta_d(t)]^T$, is considered. The kinematic model of the considered virtual robot is:

$$\begin{bmatrix} \dot{x}_d \\ \dot{y}_d \\ \dot{\theta}_d \end{bmatrix} = \begin{bmatrix} \cos \theta_d & 0 \\ \sin \theta_d & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_d \\ \omega_d \end{bmatrix}, \quad (3)$$

where $P_d = (x_d, y_d, \theta_d)$ represents the desired pose, v_d the desired linear velocity, ω_d the desired angular velocity.

Equation (3) can be expressed as:

$$\begin{cases} \dot{x}_d = v_d \cdot \cos \theta_d \\ \dot{y}_d = v_d \cdot \sin \theta_d \\ \dot{\theta}_d = \omega_d \end{cases} \quad (4)$$

The controller has to be designed to minimize the tracking errors presented in Fig. 3.

The tracking errors presented in Fig. 3 can be expressed as [7]:

$$\begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \cos \theta_d & \sin \theta_d & 0 \\ -\sin \theta_d & \cos \theta_d & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_r - x_d \\ y_r - y_d \\ \theta_r - \theta_d \end{bmatrix} \quad (5)$$

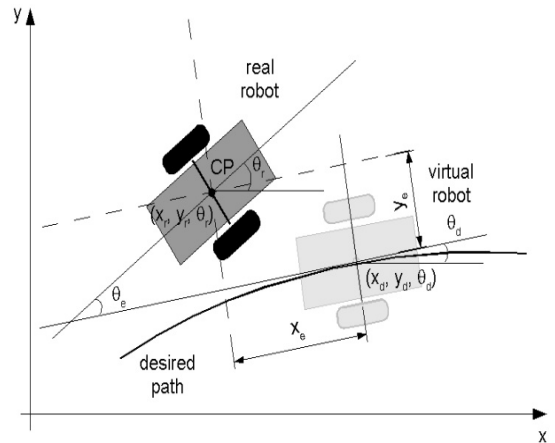


Fig. 3. Trajectory tracking errors

$$\begin{cases} \dot{x}_e = -v_d + v_r \cdot \cos \theta_e + \omega_d \cdot y_e \\ \dot{y}_e = v_r \cdot \sin \theta_e - \omega_d \cdot x_e \\ \dot{\theta}_e = \dot{\theta}_r - \dot{\theta}_d \end{cases} \quad (6)$$

III. BACKSTEPPING CONTROLLER

The backstepping controller for the PowerBot wheeled mobile robot for the global tracking problem [4] is designed using the kinematic model of the robot (2).

In order to design the backstepping controller the velocity control laws v and ω in the form:

$$v = v(x_e, y_e, \theta_e, v_d, \omega_d, \dot{v}_d, \dot{\omega}_d) \quad (7)$$

$$\omega = \omega(x_e, y_e, \theta_e, v_d, \omega_d, \dot{v}_d, \dot{\omega}_d) \quad (8)$$

must be found such that the closed-loop trajectory defined by (6) and (7), (8) are uniformly bounded and converge to zero.

The integrator backstepping is used to design the controller because y_e is not directly controlled in the error tracking model (6). The functions $x_e = k_1 \cdot \omega_d \cdot y_e$ and $\theta_e = 0$ are stabilizing functions for the y_e system of (6).

A change of variable is used:

$$\bar{x}_e = x_e - k_1 \cdot \omega_d \cdot y_e \quad (9)$$

where $k_1 > 0$ is a constant.

Using (9) in (6), the equation of x_e changes to:

$$\begin{aligned} \dot{\bar{x}}_e = & -v_d + v_r \cdot \cos \theta_e + \omega_d \cdot y_e - k_1 \cdot \dot{\omega}_d \cdot y_e - \\ & - k_1 \cdot \omega_d \cdot (-\omega_d \cdot x_e + v_r \cdot \sin \theta_e) \end{aligned} \quad (10)$$

A new change of variable is introduced:

$$\begin{aligned} u = & v_r \cdot \cos \theta_e + \omega_d \cdot y_e - \\ & - k_1 \cdot \omega_d \cdot (-\omega_d \cdot x_e + v_r \cdot \sin \theta_e) \end{aligned} \quad (11)$$

Consider the Lyapunov function:

$$V(t, x_e, y_e, \theta_e) = \frac{1}{2} \cdot \bar{x}_e^2 + \frac{1}{2} \cdot y_e^2 + \frac{1}{2 \cdot \lambda} \cdot \theta_e^2 \quad (12)$$

with $\lambda > 0$.

$$\begin{aligned} \dot{V}(t, x_e, y_e, \theta_e) = & -k_1 \cdot \omega_d^2 \cdot y_e^2 + \bar{x}_e (-\omega_d \cdot y_e + v_l - v) \\ & + \frac{1}{\lambda} \cdot \theta_e \cdot [\lambda \cdot y_e \cdot v_r \int_0^1 \cos(s \cdot \theta_e) ds + \omega_r - \omega_d] \end{aligned} \quad (13)$$

The resulting controls are:

$$v = u - y_e \cdot \omega_d + k_2 \cdot \bar{x}_e \quad (14)$$

$$\omega = \omega_r + \lambda \cdot y_e \cdot v_r \int_0^1 \cos(s \cdot \theta_e) ds + k_3 \cdot \lambda \cdot \theta_e \quad (15)$$

where $k_2, k_3 > 0$.

$$\dot{V}(t, x_e, y_e, \theta_e) = -k_1 \cdot \omega_d^2 \cdot y_e^2 + k_2 \cdot \bar{x}_e^2 - k_3 \cdot \lambda \cdot \theta_e^2. \quad (16)$$

IV. SIMULATION RESULTS

Simulation results are presented in this section in order to validate the proposed control law. Experiments carried out on the trajectory tracking problem of wheeled mobile robots shows the effectiveness of the proposed backstepping controller.

The control program is written in C++ and run on a Pc, with sampling time set to $T_s = 100$ ms. All the simulations were made using MobileSim. MobileSim is a software designed to simulate the behavior of MobileRobots/ActivMedia platforms produced by Mobile Robots Inc. and their environments, for debugging and experimentation with ARIA.

The control architecture used to command the robot is presented in Fig. 4. A trajectory planner is used to calculate the desired speed and angular velocity. The trajectory resulted after running the planner is presented in Fig. 5.

The simulation required the PowerBot to follow the trajectory from Fig. 5. The values of k_1, k_2, k_3 was set to 1.

In Fig. 6 the trajectory simulated in MobileSim is presented. Fig. 7 presents the real and the desired trajectory of the simulated robot. The robot is capable of tracking the desired trajectory with low tracking errors.

The trajectory tracking error on the X-axis (x_e) is presented in Fig. 8. The trajectory tracking error on the Y-axis (y_e) is presented in Fig. 9. In Fig. 10 the heading error (θ_e) is presented. From Fig. 9 and Fig. 10 is obvious that during the linear segment of the trajectory the lateral and heading errors were kept to zero and small lateral errors were present in the curved sections of the trajectory.

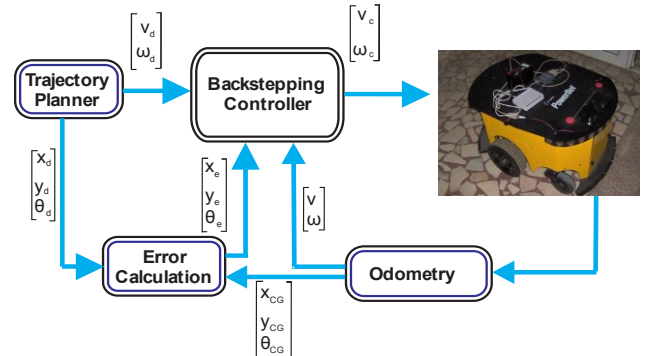


Fig. 4 Control architecture

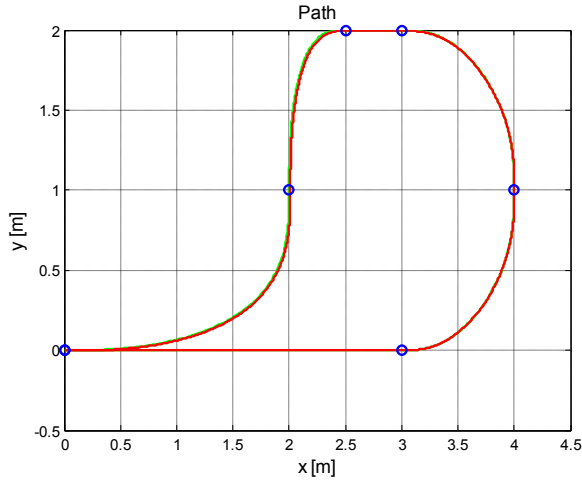


Fig.5 Desired Trajectory

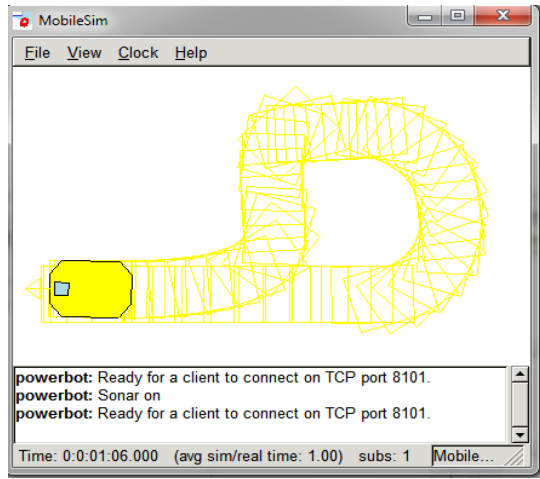


Fig.6 MobileSim Simulated trajectory

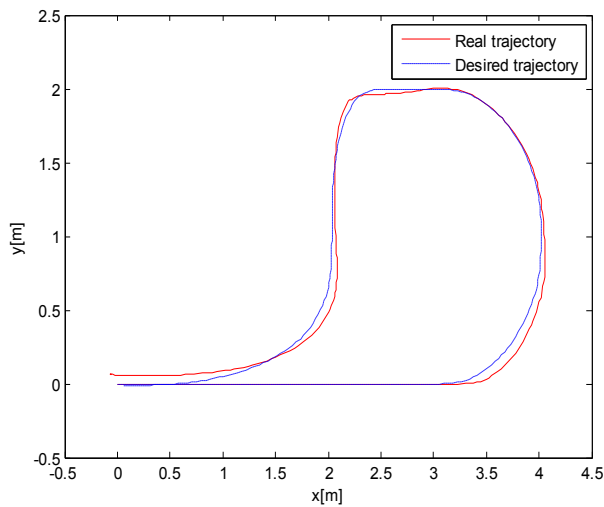


Fig. 7 Real and desired trajectories

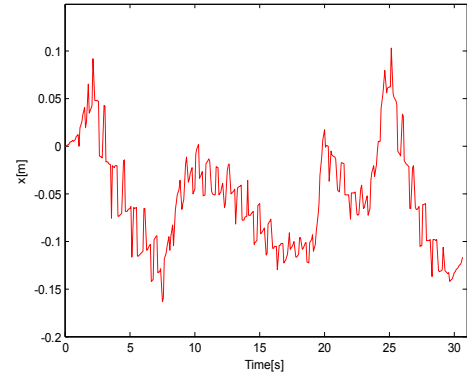


Fig. 8 Trajectory tracking error on the X-axis

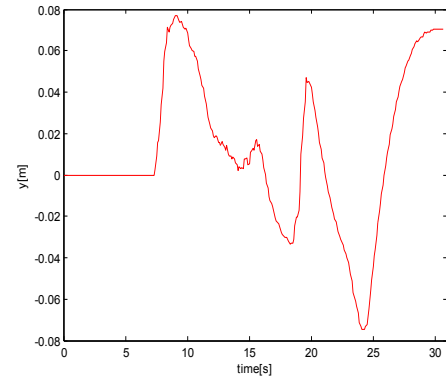


Fig. 9 Trajectory tracking error on the Y-axis

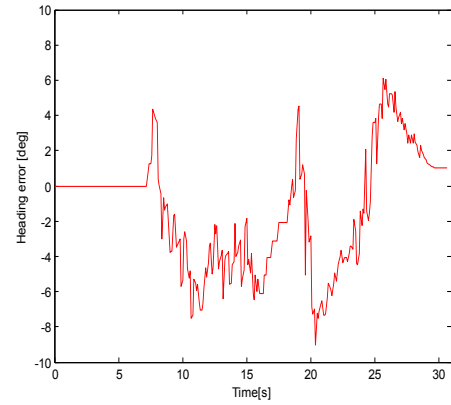


Fig. 10 Heading error

In Fig. 11 the real speed, desired speed and calculated speed are presented. Fig. 12 presents the angular velocity of the robot, the desired angular velocity and the calculated angular velocity.

It can be seen from figs. 7-10 that the trajectory tracking errors are small and the desired trajectory is followed, and from Fig. 11 that the real linear velocity is similar to the desired velocity and the calculated velocity and from Fig. 12 that the angular velocity is similar to the calculated and desired velocities.

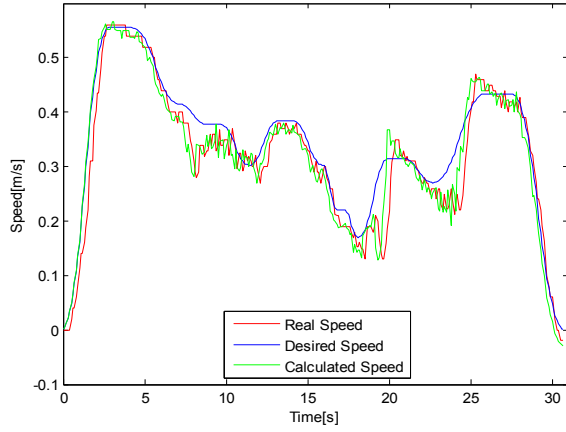


Fig. 11 Linear velocities

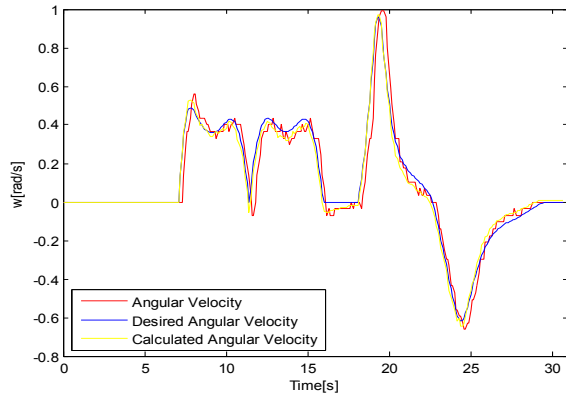


Fig. 12 Angular velocities

V. REAL TIME RESULTS

Real-time results are presented in this section in order to validate the proposed control law. Experiments carried out on the trajectory tracking problem of wheeled mobile robots shows the effectiveness of the backstepping controller.

The test was carried out on the PowerBot mobile platform. The control program was written in C++ and run on a Pc, with sampling time set to $T_s = 100ms$. A wireless network was used to facilitate communication between the robot and the PC. The command architecture is presented in Fig. 4 and the desired trajectory is identical to the one from the simulations.

Fig.13. shows the desired trajectory with a blue interrupted line and with a continuous red line the resulted trajectory. The desired trajectory is tracked by the robot with small positioning errors. Fig.14. shows the tracking error on the X-axis, an significant error is present in the trajectory tracking in the first 5 seconds of the trajectory but the error is reduced afterwards and kept close to zero. Fig. 15 shows the error on the Y-axis and the heading error is displayed in fig. 16, from these figures it can be seen that the lateral errors and heading errors are small and present

even in the linear section of the trajectory. In Fig. 17 the real speed, desired speed and calculated speed are presented. Fig. 18 presents the angular velocity of the robot, the desired angular velocity and the calculated angular velocity. It can be seen that the trajectory tracking errors, although larger than the simulated ones, are small, the real velocities are close to the desired velocities and the desired trajectory is followed.

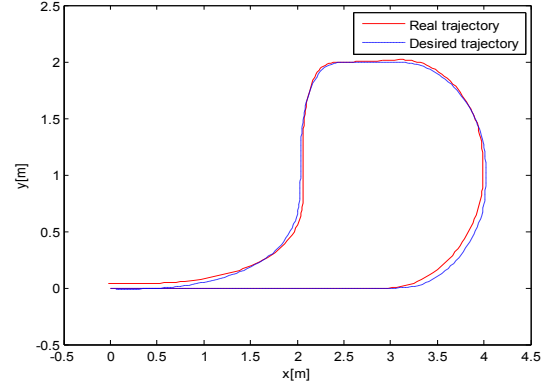


Fig. 13 Real and desired trajectories

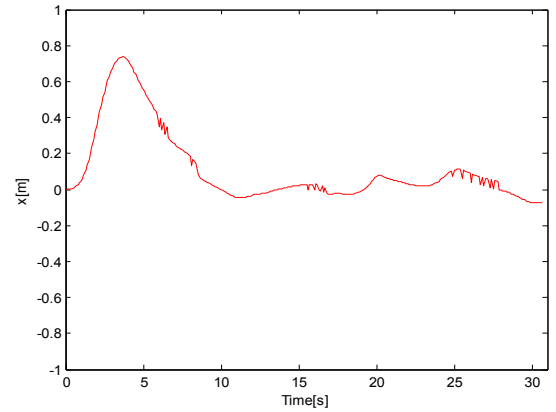


Fig. 14 Trajectory tracking error on the X-axis

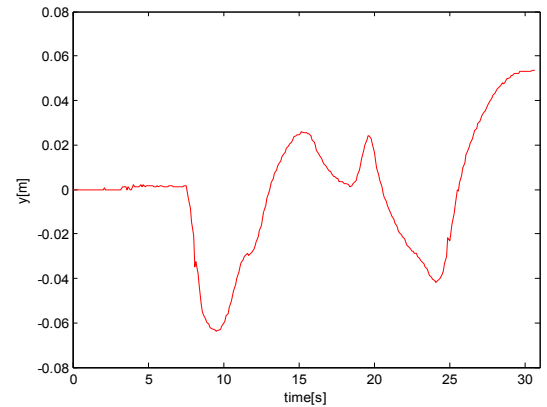


Fig. 15 Trajectory tracking error on the Y-axis

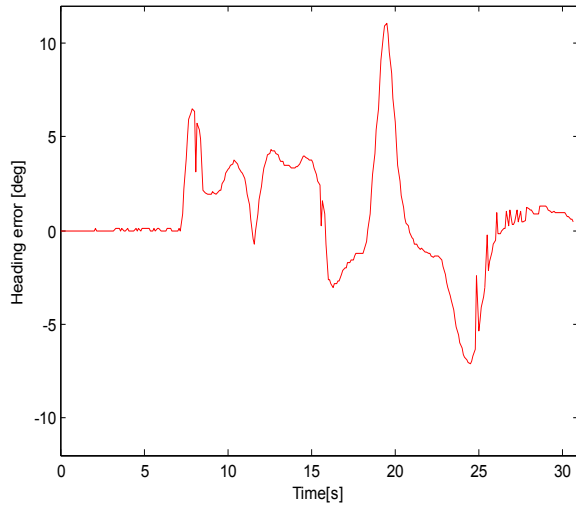


Fig. 16 Heading error

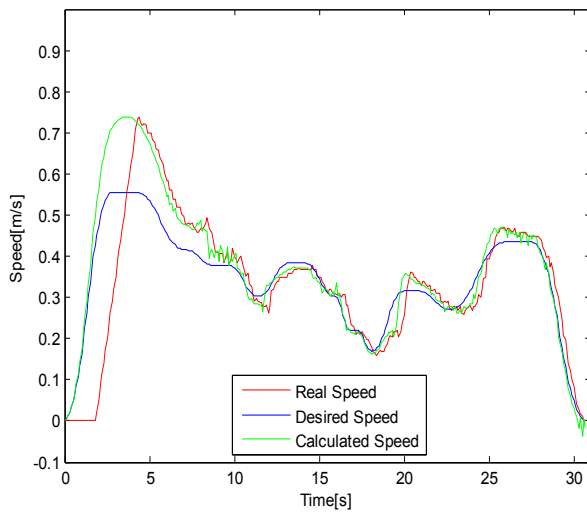


Fig. 17 Linear velocities

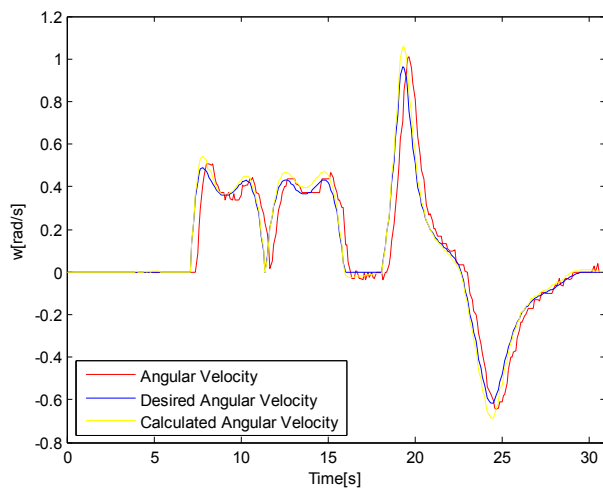


Fig. 18 Angular velocities

VI. CONCLUSION

A backstepping controller for the PowerBot wheeled mobile robot based on its kinematic model is presented. The controller is tested using both simulations and real-time tests.

A trajectory was generated using a trajectory planner and the results obtained both in simulation and real-time are presented.

In both cases the trajectory tracking errors are small enough and the velocities of the PowerBot match those of the desired trajectory.

From the presented tests it can be concluded that the backstepping algorithm proposed for the PowerBot is effective.

REFERENCES

- [1] R. Brockett, "Control theory and singular Riemannian geometry," in *New Directions in Applied Mathematics*, P. Hilton and G. Young, Ed. New York: Springer-Verlag, 1981, pp. 11-27.
- [2] H. Shi, "A novel scheme for the design of backstepping control for a class of nonlinear systems", *Applied Mathematical Modelling* vol. 35, 2011, pp. 1893-1903.
- [3] C. Chen, "Backstepping control design and its applications to vehicle lateral control in automated highway systems", University of California at Berkeley, 1996.
- [4] J. P. Jiang and H. Nijmeier, "Tracking control of mobile robots: a case study in backstepping", *Automatica*, vol. 33, 1997, pp. 1393-1399.
- [5] I. Kolmanovsky and N. McClamroch, "Developments in nonholonomic control problems", *IEEE Control Systems*, vol. 15, 1995, pp. 20-36.
- [6] Z. Qiang, L. Zengbo, C. Yao, "A back-stepping based trajectory tracking controller for a non-chained nonholonomic spherical robot", *Chinese Journal of Aeronautics* vol 21, 2008, pp. 472-480.
- [7] Y. Kanayama, Y. Kimura, F. Miyazaki and T. Noguchi, "A stable tracking control scheme for an autonomous mobile robot", *Proceedings of the IEEE International Conference on Robotics and Automation*, 1990, pp.384-389.
- [8] H.G. Tanner, K.J. Kyriakopoulos, "Backstepping for nonsmooth systems", *Automatica*, vol. 39, 2003, pp.1259-1265.
- [9] H.G. Tanner, K.J. Kyriakopoulos, "Discontinuous backstepping for stabilization of nonholonomic mobile robots, *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, may, 2002, pp. 3948-3953.
- [10] F. Mnif, "Recursive backstepping stabilization of a wheeled mobile robot", *International Journal of Advanced Robotic Systems*, vol. 1, 2004, pp. 287-294.
- [11] H.K. Khalil, "Nonlinear systems, 2nd edition, Prentice Hall, New Jersey, 1996
- [12] J. J. E. Slotine and W. Li, "Applied nonlinear control", Prentice Hall, London, 1991.
- [13] F. Takanori, A. Norihiko, h. Wang, "Adaptive Tracking control of nonholonomic Mobile robots: a backstepping approach", *Journal of Robotics Society of Japan*, vol.19, 2001, pp. 271-276.
- [14] <http://www.mobilerobots.com>

Experiments in Determining the Displacements and Angular Amplitudes of a Hyper-Redundant Robot

Sorin DUMITRU, Dorian COJOCARU, Florin MANTA

Abstract - This paper presents a series of experiments performed in order to determine the displacements and the angular amplitudes of a hyper-redundant robot. Two high-speed cameras were used to extract data related to the displacements of the robotic structure in order to obtain a control algorithm of the robot. Has been developed a parameterized model of the robot in order to observe the displacement and the rate of travel during its actuation.

Key-words: hyper-redundant robot, visual analysis, finite element

I. INTRODUCTION

The structure and design of modern industrial robots can be inspired by the human arm. This kind of mechanisms with serially arranged discrete links could be used in the industrial environment due to the fact that their workspace is specially designed for their structure. Industrial robots are placed in environments that are carefully controlled and where a very small number of people have access to. In the applications that require a robotic structure that is supposed to interact with the natural environment, the rigid structure of industrial robots is limited because of the workspaces that can be situated in small spaces, rough fields etc.

Behaviour similar to biological trunks, tentacles, or snakes may be exhibited by continuum or hyper-redundant robot manipulators. Hence these manipulators are extremely dexterous, compliant and are capable of dynamic adaptive manipulation in unstructured environments, continuum robot manipulators do not have rigid joints unlike traditional rigid-link robot manipulators. The movement of the continuum robot mechanisms is generated by bending continuously along their length to produce a sequence of smooth curves [1], [2]. This is opposite with discrete robot devices, which generate movement at independent joints are separated by supporting links.

The snake-like robots and elephant's trunk robots are also described as continuum robots, although these descriptions are restrictive in their definitions and cannot be applied to all snake-arm robots. A continuum robot can be defined as a continuously curving manipulator, similarly with the arm of an octopus [3] or an elephant's trunk.

A research team from the Faculty of Automation, Computers and Electronics in Craiova has developed a robotic structure that has the characteristics of a continuum robot. Has been developed a parameterized model of the structure with the possibility of changing the structure or the

materials of the flexible units, depending on the field of use. Modelling with finite elements in the dynamic regime replaced by the simulation of the flexible unit control has enabled the completion of constructive solution for the experimental model. The snake-like robotic system may have multiple utilizations in medicine or industry.

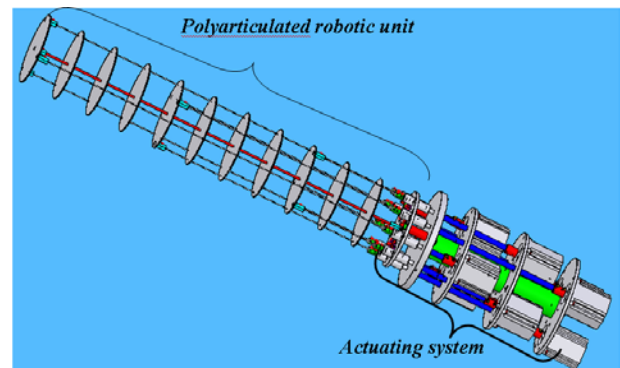


Fig. 1. Virtual model of the robotic system, with the support unit

The robotic unit is composed from two units: one is a flexible structure with cinematic possibilities similar with the snake's locomotion and another one for acting.

The flexible unit is conceived in three modules with decoupling possibilities for a controlled optimization of the working space. Each module is actuated by three steel wires situated at the extremities of each module and a central column mounted rigidly to all the disks. The intermediary wires are mounted equidistantly upon the central spine. The three actuating spines are rigidly joined only to the end flange, the joint between them and the intermediary flanges is like one translational joint.



Fig. 2. The flexible structure of the robotic unit

The wires are actuated by a system consisting in screw joints elements. The shape of the flexible unit is achieved by pulling or pushing the actuating spines.

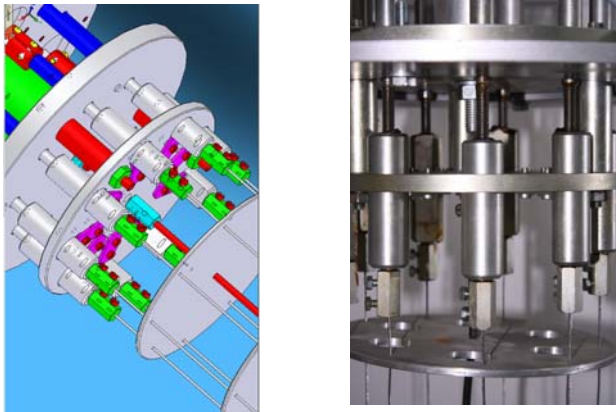


Fig. 3. The actuation unit (virtual model and real model)

The actuating elements are disposed on three levels, each level containing three actuators. This disposal was preferred because of the dimensions of the stepper motors. The motors from each level are corresponding to the actuation of one module from the robot's flexible unit.

The elements that are generating the movement are represented by Berger-Lahr triphase stepper motors.

The command and control system is composed from one computer and a number of nine controllers, each for every motor. The PC has an Intel E6300 dual core processor, an available Ram memory of 512 Mb and the operating system is Windows XP. The main elements forming the software control architecture are the Visual C++ programming environment and the command functions included in the .DLL libraries provided by AD-Link. A program was developed in order to generate the displacement commands; this program computes the cable length variation in order to obtain a desired curvature and orientation, determines the equivalent number of motor steps, and transmits the command to the controllers, using the functions provided by Berger Lahr DLL's.

Using the parameterized model the cinematic and the dynamic of the robotic structure were simulated.

Has been elaborated a method for measuring the positions, the speeds and the accelerations based on the information provided from the images obtained during the actuation of the robotic structure.

Using the equipment described in the next chapter of this paper, we were able to compare the real determinations with the ones obtained from the simulation.

II. DESCRIPTION OF ROBOT ANALYSIS EQUIPMENT

For the experiment described in this paper, we have used an equipment dedicated for motion analysis called CONTEMPLAS. This equipment is composed from the TEMPLO motion analysis software, two ultra-speed cameras and a computer.

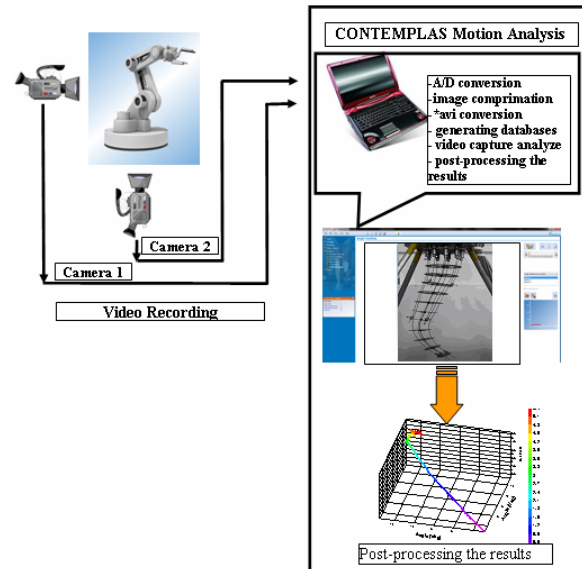


Fig. 4. The experiment's structure

The experiment follows the next steps:

1. Creating the database: by creating the database the video recordings can be sorted and filtered; some analyzing criteria can be selected by redefining the quality of the video recordings.
2. The video recordings: on the model that is to be studied will be attached markers on points (situated on the mechanical structure) that are considered to be important for the cinematic analysis.
3. Analyzing: during this step are measured angular variations by identifying the position of every marker attached to the mechanical system during the structure's motion. The software allows the defining of analyzing schemes, marking or unmarking characteristic points situated on a system.
4. Result's post-processing: the resulted data can be exported in .xls folders that contain values of the analysis that can be used for generating diagrams representing the trajectory of the critical points.



Fig. 5. The arrangement of the hardware components for the experiment

III. RESULTS OF THE EXPERIMENT

A. The Importance Of The Angular Displacements

In order to control a hyper-redundant robot we have to develop a method to compute the positions for each one of its segments [4]. By consequence, given a desired curvature $S^*(x, t_f)$ as sequence of semi circles, the goal is to identify how to move the structure in order to obtain $s(x, t)$ such that

$$\lim_{t \rightarrow t_f} s(x, t) = S^*(x, t_f)$$

where x is the column vector of the shape description and t_f is the final time. The shape of each of the three segments that compose the robotic structure can be described by using a circular arc. R_1 represents the radius of the circle that corresponds to the circular arc used to describe the shape of the first module, R_2 corresponds to the second module and R_3 is used for the third module.

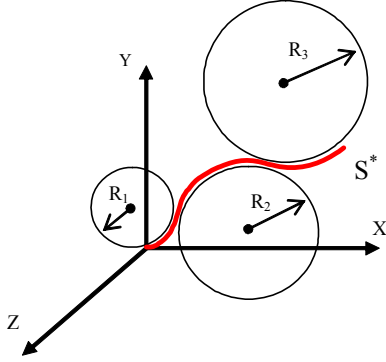


Fig. 6. The description of the desired shape

In [5], [6] and [7] was presented a method that is based on the particular structure of the system defined as a “backbone with two continuous angles $\theta(s)$ and $q(s)$ ”. The control of the system is based on the control of the two angles $\theta(s)$ and $q(s)$. These angles are measured directly or indirectly. The angle $\theta(s)$ is measured directly by the projection on the image plane $Z_{S_1}O_{S_1}Y_{S_1}$ (3) and $q(s)$ is computed from the projection on the image plane $Z_{S_2}O_{S_2}Y_{S_2}$ in (4).

The essence of the model is a 3 – dimensional backbone curve C that is parametrically described by a vector $r(s) \in \mathbb{R}^3$ and an associate frame $\Phi(s) \in \mathbb{R}^3$ whose columns create the frame base (Fig. 6). The independent parameter s is related to the arc length from the origin of the curve C . The position of a point s on the curve C is defined by the position vector,

$$\bar{r} = \bar{r}(s) \quad (2)$$

where $s \in [0, l]$.

$$\theta_s(s') = \theta(s), \quad s \in [0, l], \quad s' \in [0, l'] \quad (3)$$

for the plane $Z_{S_1}O_{S_1}Y_{S_1}$ and $tgq_s = \frac{\Delta z_{s2}}{\Delta y_{s2}} = \frac{\Delta z}{\Delta y}$.

This relation allows the computation of the orientation angle q_s in the plane $Z_{S_2}O_{S_2}Y_{S_2}$

$$tgq_s(s'') = tgq(s) \cdot \frac{1}{\cos \theta(s)} \quad (4)$$

where, s', s'' and l', l'' represent the projections of the variable s and the length l in the two planes, respectively.

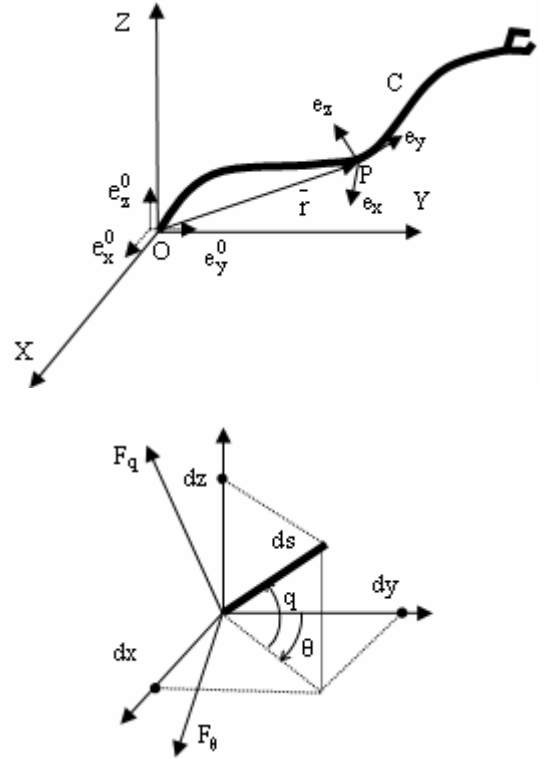


Fig. 7. The structure of the backbone and its parameters

The conclusion of the method presented above is that if the values of the two angles $\theta(s)$ and $q(s)$ are known, the position of the robot can be controlled.

B. Data Processing

The software and hardware equipment described in the previous chapter was used in order to obtain experimental data from the motion of the robotic structure presented in chapter 1. Two ultra-speed video cameras were placed at a 1,5 meters from the robotic structure (facing the robot) and perpendicular on the robot.

Using markers, there were established n points of interest (in this particular case $n = 11$) on the mechanical

structure in order to study the motion. All those points were situated on the central column of the flexible unit.

In the following we will present the results from analyzing the displacements and the angular variations of the 11 points. The angular variations are computed referring to a fixed point established at the base of the robotic structure.

At first we actuated only the first module, after only the second one and then only the third one. Finally, all the three modules were actuated simultaneously.

The processing of the data obtained from the experiment was performed in the following order:

- the calculation of the displacements of the first module;
- the calculation of the displacements of the second module;
- the calculation of the displacements of the third module;
- the calculation of the displacements of the entire robotic structure;
- the calculation of the displacements of all the n selected points;
- the presentation of the angular variations for the first module of the robotic structure;
- the presentation of the angular variations for the second module of the robotic structure;
- the presentation of the angular variations for the third module of the robotic structure;
- the presentation of the angular variations for all the n selected points situated on the mechanical structure;

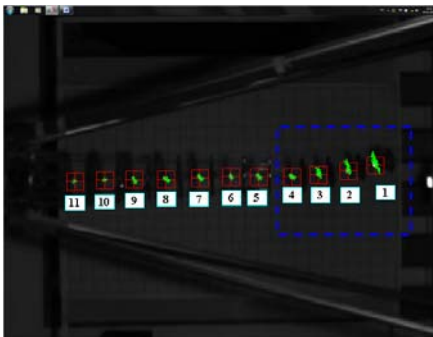


Fig. 8. The displacement analyse from the field of view of camera 1

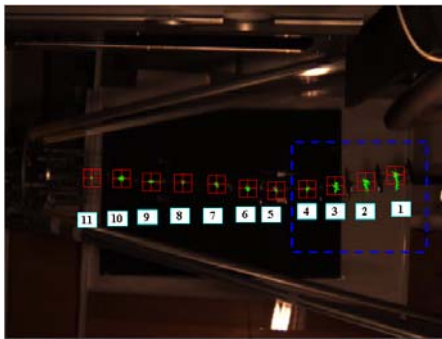


Fig. 9. The displacement analysis from the field of view of camera 2

The results of the displacement for the first module can be observed in the following figures:

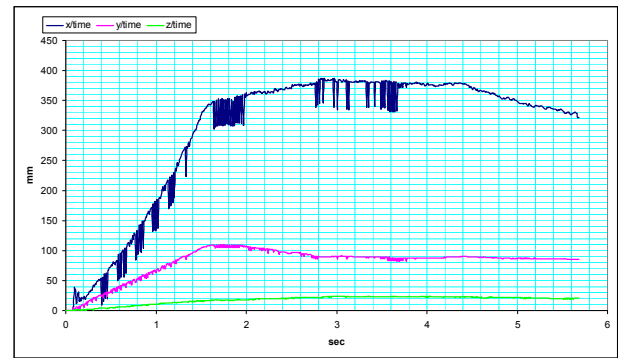


Fig. 10. The result of the displacement for the first point the field of view of camera 1

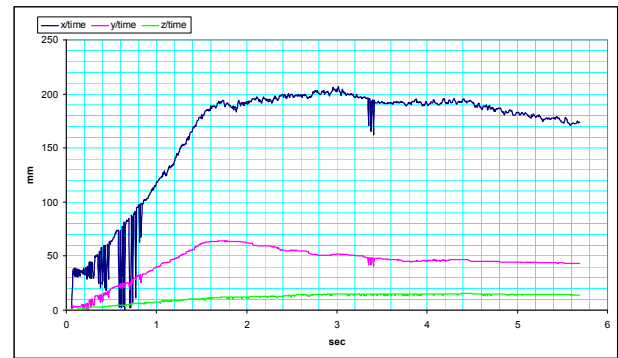


Fig. 11. The result of the displacement for the first point the field of view of camera 2

The angular displacements for the first module (both from the field of view of camera 1 and camera 2) can be observed in the following figures:

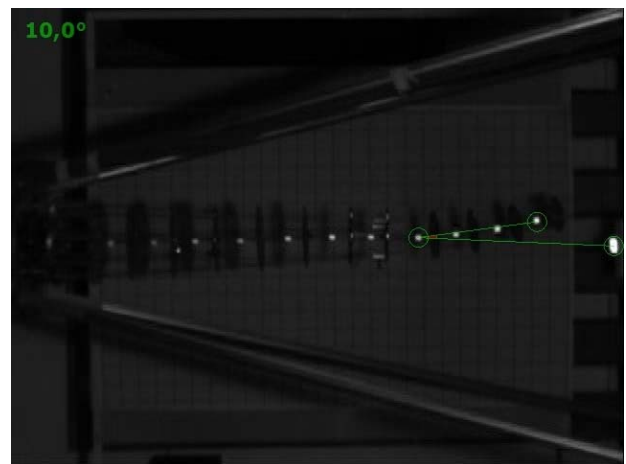


Fig. 12. Measurement of the angular displacement from the camera 1 point of view

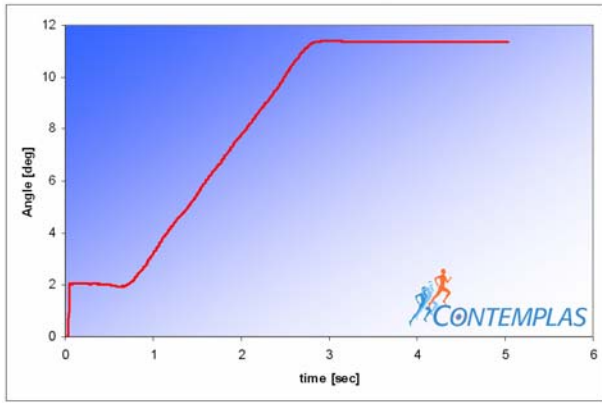


Fig. 13. The angular amplitude for the first point from the point of view of camera 1

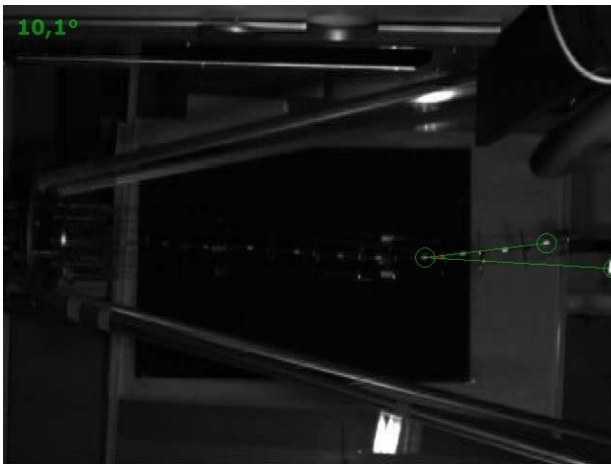


Fig. 14. Measurement of the angular displacement from the camera 2 point of view

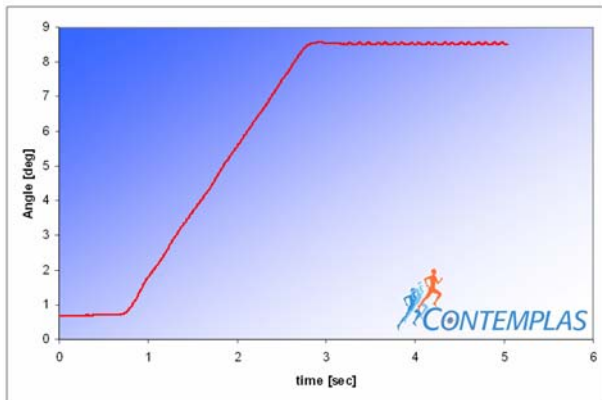


Fig. 15. The angular amplitude for the first point from the point of view of camera 2

Using the software ADAMS, we have developed a parameterized model of the robotic structure.

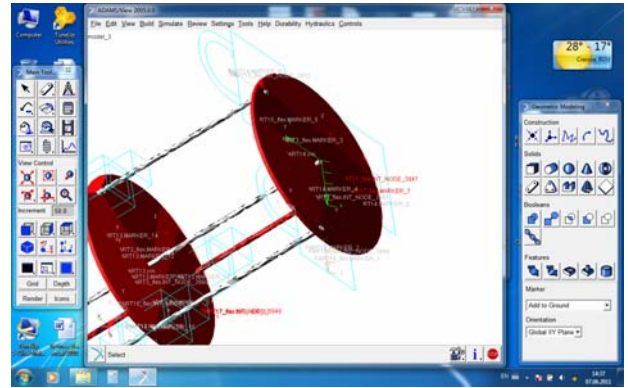


Fig. 16. The CAD model

We applied markers on the first and the second disk of the first module in order to observe the displacements and the velocity of this two points of interest during the actuation of the robotic structure.

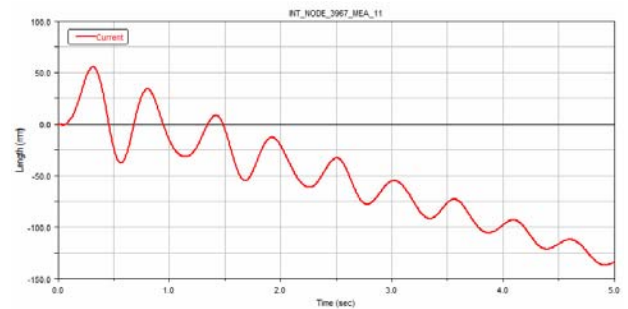


Fig. 17. The displacements of the first disk along the X axis

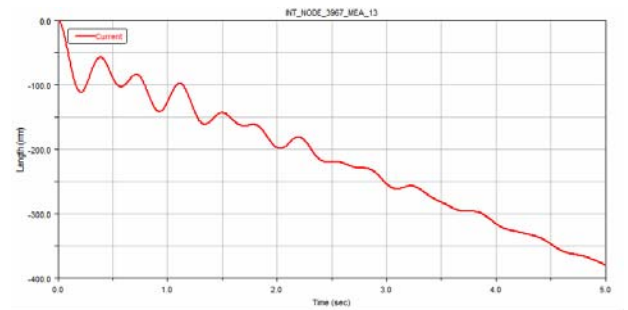


Fig. 18. The displacements of the first disk along the Y axis

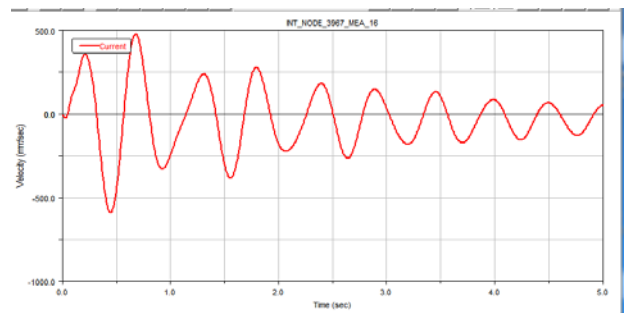


Fig. 19. The rate of travel of the first disk along the X axis

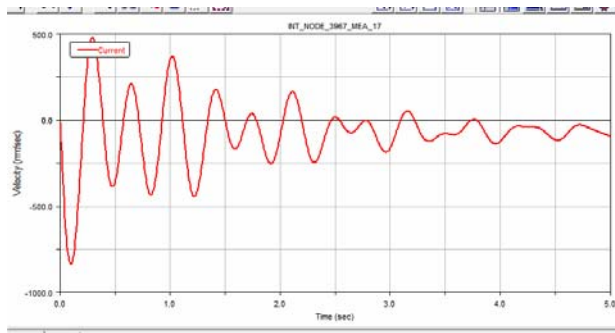


Fig. 20. The rate of travel of the first disk along the Y axis

IV. CONCLUSIONS

In this paper was studied, using image sensors and the Contemplas software, the motion of a hyper-redundant robotic structure.

In order to study the displacements and the angular amplitudes of the robotic system during its motion, were selected 11 points placed along the central column of the mechanical structure.

The experiment was performed by following the next steps:

- 2 ultra-speed cameras were positioned at a 1,5 m distance from the robotic unit; the camera's optical axes made an 90 degrees angle on the horizontal;
- the cameras were connected to a PC used for analysis;
- was actuated only the first module, after only the second one and finally only the third one (after every of each actuation, the robotic structure was brought in the initial position);
- was performed a combined motion of all the three modules in order to collect data from the motion of the entire robotic structure;
- was selected a fix point at the base of the robot in order to study the angular amplitude;
- the images acquired from the 2 video cameras were studied frame by frame; the obtained data were exported in Excel and used for graphical representation.

After the interpretation of the collected data, it was performed the analysis of the displacements for each module and the displacements for the entire robotic structure (following to the simultaneously actuation of all the three modules).

In the cases in which only one module was actuated, there were studied not only the displacements for the points corresponding to that module, but for all the n selected points.

Both the angular amplitudes and the displacements were calculated from the perspective oh both cameras, in order to compare the resulted data.

The data obtained from this experiment were used for the determination and the validation of the kinematical model of the robotic structure presented in this paper and compare the results with the ones obtained from the simulation of the virtual model.

In conclusion, we can submit that the cinematic of the robotic structure has errors similarly with the ones

obtained from the simulation. From the dynamically point of view, the robot's behavior is very different from the virtual simulation, because at the online actuation of the stepper motors large errors appear due to the fact that some parameters that were programmed at the beginning need to be varied .

REFERENCES

- [1] I.D. Walker, and C. Carreras, "Extension versus Bending for Continuum Robots", *International Journal of Advanced Robotic Systems*, Vol. 3, No.2, 2006, ISSN 1729-8806, pp. 171-178.
- [2] I.D. Walker, Darren M. Dawson, a.o. *Continuum Robot Arms Inspired by Cephalopods*, DARPA Contr. N66001-C-8043, 2005.
- [3] J.B.C. Davies, D.M. Lane, a.o. 1998, Subsea applications of continuum robots, Underwater Technology, *Proceedings of the 1998 International Symp. on Volume*, Issue 15-17 Apr 1998, Page(s):363 – 369.
- [4] G. Boccolato, D. Cojocaru, a.o., 2009, 3D Control for a Tentacle Robot, ISBN 978-960-474-147-2, ISSN 1790-5117, *Selected Topics on Applied Mathematics, Circuits, and Signals 2009*, p100-106, *Proceedings of NAUN the 3rd International Conference on Applied Mathematics, Simulation, Modelling (ASM'09)* ISBN 978-960-474-147-2, ISSN 1790-5117, Vouliagmeni, Athens, Greece, December 29-31 [ISI Proc., INSPEC].
- [5] D. Cojocaru, M. Ivanescu, R.T. Tanasie, S. Dumitru, F. Manta, 2010, Vision control for hyperredundant robots, *IJAA International Journal Automation Austria*, ISSN 1562 – 2703, p66-80. [IFAC-Beirat Österreich]
- [6] M. Ivanescu, D. Cojocaru, a.o. 2007, Visual Servoing By Artificial Potential Methods Of A Hyperredundant Robot, *Proceedings of 16th International Workshop on Robotics in Alpe-Adria-Danube Region - RAAD 2007*, June 7-9, Ljubljana, Slovenia.
- [7] M. Ivanescu, 2002, Position Dynamic Control for a Tentacle Manipulator, *Proceedings IEEE International Conference on Robotics and Automation, Washington*, A1-15, 1531-1539.
- [8] Dumitru, S., Cojocaru, D., ş.a., 2009, Finite Element Modeling of a Polyarticulated Robotic System, *The 20th DAAAM International Symposium Intelligent Manufacturing & Automation: Theory, Practice & Education*, ISSN 1726-9679, ISBN 978-3-901509-70-4, p3331, Vienna, 2009-11-25/28 [ISI Proc.].
- [9] D. Cojocaru, S. Dumitru, F. Manta, G. Boccolato, I. Manea, Experiments with a Continuum Robot Structure, *7th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2010*, ISBN: 978-989-8425-01-0, vol. 2, 15-18 June 2010, p198-205, Funchal, Madeira, Portugal [Thomson Conference Proceedings Citation Index, INSPEC, dblp, Awaitingconfirmation of indexation].

Robust predictive control of a wet-clutch using evolutionary algorithm optimized engagement profile

Abhishek Dutta, Robain De Keyser, Yu Zhong, Bart Wyns
Electrical Energy, Systems and Automation,
Ghent University, 9052 Gent, Belgium

Gregory Pinte and Julian Stoev
Flanders Mechatronics Technology Center,
3001 Heverlee, Belgium

Abstract—Modeling of hydraulic clutch transmissions are not straightforward because of their hard nonlinearities. Therefore a model-free optimizer is used to tune the input parameters for desired operation, thereby generating optimal reference trajectory. However these machines suffer from time varying dynamics, hence a feedback predictive control is used to track this reference against identified linear sub-models to ensure robust performance. This two step strategy is called Learning Predictive Control.

I. INTRODUCTION

Modeling and control of a hydraulic clutch transmission is by far a complicated process. This is due to the interplay of various subsystems introducing hard nonlinearities and discontinuities with added time variance of its constituent parameters.

In general, such a system can be decomposed into the clutch itself and the source of actuation which in this case is a hydraulic mechanism (Fig. 1). The relationship between flow and pressure in the hydraulics and coupling of hydraulics to the clutch itself exhibit stiff nonlinearity. In principle there are two other dominant reasons which make the full identification of such systems very complex. First is the lack of sensors. Second is the variability in the actuation systems which could be pneumatic or even electric and then the clutch itself can be wet or dry. Therefore the identification would be far from general and too specific for the given setup.

The modeling and control of slip and stick for dry clutches to maintain a certain torque has received considerable attention in literature, more recently using hybrid optimal control [1]. Position based control of the clutch has been developed in [2] with predictive and in [3] with flatness based control.

In this paper a novel two step strategy of learning and control is introduced to achieve a fast engagement with minimum torque loss (i.e. deviations from steady state torque output) for an electro hydraulic wet clutch ensuring driver comfort. Note that, these two objectives are conflicting. Any model based control would require an accurate hard nonlinear hybrid model of the system which as argued is too expensive a task. Therefore a genetic algorithm (GA) based optimization would be used alternatively to evolve a suitable input current trajectory satisfying the desired output (slip, torque) bounds. The corresponding measured pressure signal would be optimal too.

However, such a feedforward scheme is not robust to un-

known disturbances and model uncertainties. This could for example arise due to the change in the flow rate due to temperature variation causing a change in viscosity of the oil. In [4], robust control of wet clutches has been identified as a challenging industrial control problem. Therefore a model based predictive control (MPC) scheme would be used to track this intermediate pressure reference which also corresponds to the optimal output. This is because the input current to pressure subsystem can be identified with a linear approximation.

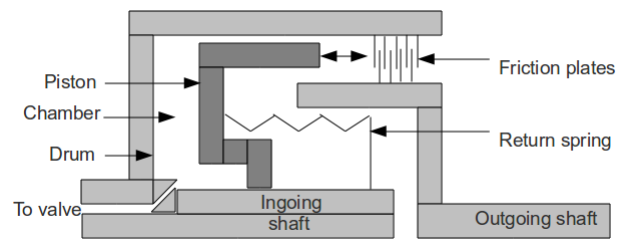


Fig. 1. Wet clutch design

II. INSIGHTS INTO CLUTCH MODEL

The control process can be divided into three phases, filling, slipping and engagement. Filling refers to building up of the pressure in the clutch till the first contact of the piston with the friction plates. In general it's considered that there's no torque transferred in this phase, however this assumption for this setup doesn't hold and hence makes the problem even more challenging. Due to high compression of the fluid, output torque starts building just before any contact is made. The physical contact triggers the slipping phase where torque loss is due to kinetic friction, causing a rotational speed difference between input, output axes. Once both these speeds are the same, the clutch is in the engagement phase with sufficient static friction.

The overall block diagram of a clutch to highlight the flow of physical variables is illustrated in Fig. 2. A 'Current' is input to the 'Hydraulics' subsystem which drives a 'Hybrid Clutch' subsystem. The 'Engine Torque' can be treated as a known disturbance to the 'Hybrid Clutch'. The measured outputs are 'Engine and Vehicle speeds, Torque outputs' and 'Pressure' from 'Hydraulics'.

A. Hybrid Clutch

A Coulomb friction model can be used to get an idea on the relationship between the input normal force F_n , engine and vehicle angular speeds ω_e, ω_v respectively and clutch torque τ_{cl} [5].

Ideally there's not much impact on the output in the filling phase. The slipping phase behavior can be summarized by the following equation:

$$\tau_{cl} = \text{sign}(\omega_e - \omega_v) \cdot \frac{2}{3} R F_n \mu_k \quad (1)$$

with the torque capacity as a function of its size R , kinetic friction characteristic μ_k and force due to the pressure.

The system loses a degree of freedom upon engagement, the transmitted torque goes through a step discontinuity. The slip to engagement transition is triggered when:

$$\text{Slip} = \omega_e - \omega_v = 0 \quad (2)$$

The torque transmitted by clutch in engagement phase is [6]:

$$\tau_{cl} = \frac{J_v \tau_{en} - (J_v b_e - J_e b_v) \omega}{J_v + J_e} \quad (3)$$

where J_e, b_e, J_v, b_v are the moments of inertia and damping rates for engine and transmission respectively, τ_{en} is the engine torque, and ω the shaft speed.

The factor μ_k can depend on angular velocities and contact time with friction plates.

B. Hydraulics

This system can be split into three parts which are the valve, hydrodynamics and mechanical subsystems [7]. A solenoid driven proportional pressure reducing valve is the input device of the system, which controls pressure on its application side. The turbulent flow through an orifice has the following key equation:

$$|\phi| = \gamma(\theta_{oil}) A_{orifice}(|x_{stem}|) \sqrt{|p_{clutch} - p_{pump/tank}|} \quad (4)$$

where θ_{oil} depends on the oil used and its temperature to take into account its viscosity. The clutch, pump and tank pressures are $p_{clutch}, p_{pump}, p_{tank}$ respectively. The area of orifice $A_{orifice}$ is a function of the stem position x_{stem} .

The hydrodynamics describes relation between oil flow and oil pressure. By the law of conservation of mass, it follows:

$$\dot{p}_{clutch}(t) = \frac{k_s}{V_0 + A_{piston} x_{piston}(t)} \phi(t) \quad (5)$$

where V_0 is the chamber volume, k_s the oil compressibility, x_{piston}, A_{piston} are the piston position and area respectively. The relation holds for all the three phases of filling, slipping and engagement.

The mechanical part gives the force:

$$F_n = p_{clutch} * A_{piston} - k x_{piston} \quad (6)$$

where k is the return spring constant.

C. Hypothesis

Based on the above physical interpretation, the following conclusions can be drawn. Since the 'Hybrid Clutch' subsystem undergoes step discontinuity, the identification and control of such systems over a real plant would be unrealistic. However, the 'Hydraulics' subsystem clearly suggests that the only time-variance in this model is caused by the piston position, which is under bounds, and does not pose discontinuity. This is also the part where the temperature plays a significant role in affecting the flow dynamics. Therefore it makes more sense to identify this subsystem and apply robust control against this bounded uncertainty, and also reject faster the disturbance caused by temperature, in this inner control loop.

An offline learning optimizer can then be used as a first step to generate the suitable reference for this inner loop based on overall torque and slip specifications.

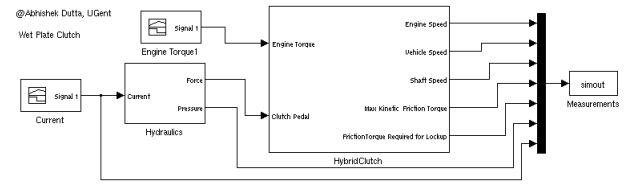


Fig. 2. Wet clutch model

III. WET CLUTCH PLANT AND PARTIAL IDENTIFICATION

The real wet clutch setup is a mechanical device that transmits torque from its input axis to its output by means of friction, often used in power transmissions of agricultural machines and off-road vehicles to selectively engage gear elements [8].

Fig. 1 shows the mechanical design of a wet clutch. The input axis of the clutch is connected to a drum with a first set of friction plates (clutch plates), while a second set of friction plates (clutch discs) are connected to the output axis. An electro-hydraulic pressure-regulated proportional valve regulates the pressure inside the clutch against a return-spring to keep the piston away from plates when not actuated.

An electromotor drives a flywheel via a torque converter and two mechanical transmissions (Fig. 3). The purpose is to engage one of the five clutches of the first transmission in the test setup, which is equipped with different sensors, measuring the speed of the ingoing and outgoing shaft, the temperature and pressure of the oil to the clutch and the output torque loss. The second transmission is used to apply an adjustable breaking torque and vary the actual load observed by the first transmission.

Modeling and control of hard nonlinear transitions together with time variance as stated before, is very complicated. Therefore a system identification of the filling phase alone, with a linear working range is performed. The dynamics between the current to the valve and generated oil pressure in the chamber of the clutch, while the piston is located between

completely open clutch position and the friction plates, is found to be [8]:

$$\frac{P-1}{I-0.04} = \frac{z^{-N}}{1-2.736z^{-1}+2.515z^{-2}-0.7764z^{-3}} \quad (7)$$

Depending on the amount of oil, that is present in the pipeline, and the oil temperature, the delay in this transfer function changes. When a pressure of approximately 10% of the maximum value is generated in the pipeline, the pipeline gets completely filled with oil and time delay gradually decreases to $N = 2$ samples, thereafter is supposed to remain constant. Note that, the units are concealed and results scaled as per the confidentiality agreement of the LeCoPro project.



Fig. 3. Transmission test bench

IV. LEARNING PREDICTIVE CONTROL (LPC)

When the clutch is commanded for engagement, the objective is to decrease the distance between the piston and the plates as fast as possible, without the piston and plates making brutal contact. This essentially means a short filling phase followed by a smooth transition into slipping. Therefore the controller can ensure a good overall engagement primarily based on the filling phase model presented above. However, under such arrangement the controller would need a pressure reference that guarantees fast filling and minimum torque loss, this restriction being imposed by the identified linear model. In order to bypass the cumbersome modelling as highlighted before of the entire clutch, a model-free scheme with a typical parameterized feedforward current signal (Fig. 4), achieving the output objectives iteratively is proposed to be used as the first step. After such an optimization, the various intermediate sensor profiles are read out and used for the feedback control.

These machines are operated through several years such that clutches undergo significant amount of wear and tear due to which their dynamics change over time. Moreover, the vehicles are used under varying environmental conditions like different temperatures in summer and winter. As a consequence of the varying conditions and system dynamics, the control signals for wet clutches leading to an optimal engagement need to be adapted during the transmission's

life cycle. This is the reason a real-time feedback controller is needed to track the feedforward optimised profiles.

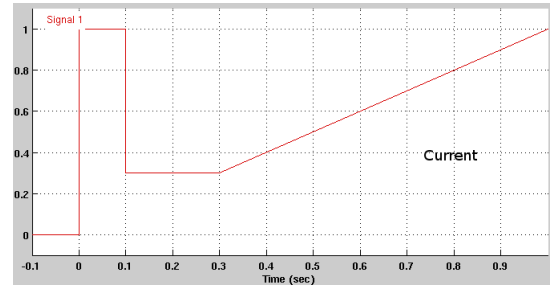


Fig. 4. Typical feedforward signal to engage clutch

V. GENETIC ALGORITHM (GA) OPTIMIZED ENGAGEMENT

The GA is a search heuristic that mimics the process of natural evolution. The heuristic is often used to generate useful solutions for optimization and search problems. GAs use techniques inspired by natural evolution, such as selection, mutation, and crossover to explore the strategy space. In the exploration phase, the fittest individual has large chance to produce offspring and thus to achieve the optimization. GA has received considerable attention as a novel approach to multi-objective optimization problems, resulting in a fresh body of research and applications. Typically, there are two ways in GA for handling the multi-objective optimization; one is to use the weighted sum method to transfer the multi-objective problem to a single objective problem; the other is based on a nondominated sorting method (NSGA) to find the Pareto front solution. The later appears to have got the most attention in the evolutionary algorithm literature and has been used in various studies. The key characteristics of this method are the nondominated sorting and the fitness assignment.

The objectives are to minimize engagement time (i.e. the time span between the application of control signal to slip going to zero) and torque loss. Before sending the individuals to test on the real setup, all individuals are first tested under no external load and no breaking inertia condition, for safety reasons. If the reading of the torque loss under such working conditions is larger than a threshold for any individual, then this one is considered as 'unsafe', and is not tested further. The fitness values for such individuals will be assigned for rejection. If the individual is checked as 'safe', it is then shifted to the low external load and low breaking inertial condition for testing, the readings of the corresponding objectives would be treated as fitness value for further operations. The working temperature is controlled at 40°C with a $\pm 1^{\circ}\text{C}$ range.

Then the Pareto ranking method is used instead of the random weight method. The initial population is randomly generated within the given range, normally distributed.

The results after optimization are plotted in Fig. 5. The middle of the Pareto front corresponds to the best possible

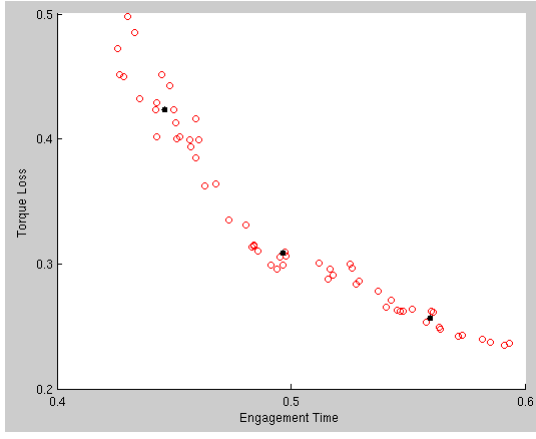


Fig. 5. Pareto front after fuzzy clustering

simultaneous minimization of torque loss and engagement time at 40°C . The optimized input current along with the sensed pressure are recorded for future use with the predictive controller. The corresponding results together with the slip and torque are plotted in Fig. 6. The engagement time is 0.5 and torque loss 0.3, which can be verified from the Pareto front.

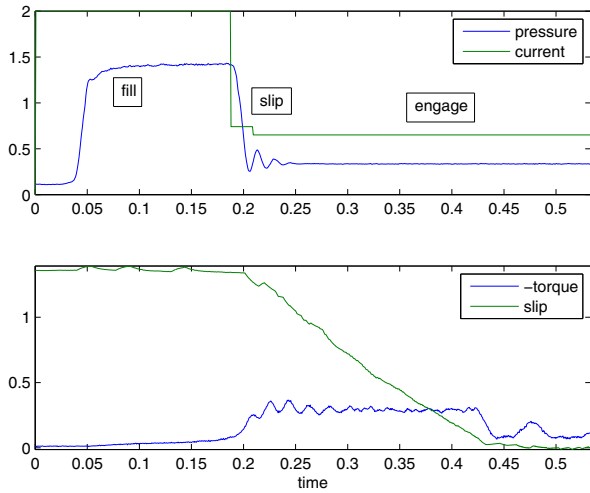


Fig. 6. Pareto front center individual

A. Under Varying Experimental Conditions

The experiment is repeated after several months. The above optimized current profile is fed forward into the real clutch plant at 40°C and the results are plotted in Fig. 7. Throughout most of the filling phase, the current is at its maximum value, with the pressure following the input as well, with zero output torque and zero output speed. This would mean the filling time is being minimized. Just before the onset of slipping, there's some torque output with marginal decrease in slip. At this point, the current drops down leading a gradual decrease in pressure. This is to ensure that the piston doesn't hit the friction plates thus avoiding

sharp torque dips. The current drops further as the clutch enters the slip phase, thereby ensuring a gentle pressure gradually closing the friction plates. This ensures smooth transition to the engagement phase with the slip now going to zero and torque jerks fading away. As mentioned earlier these are scaled values of actual experiments and the torque loss in this case is 0.8 and engagement time is 0.25. This shift in the values compared to before can be attributed to the possible variations in the experimental conditions.

The clutch is now operated continuously with the cooling turned off in order to ramp up its operating temperature to 50°C which is to be expected in a realistic scenario as it would be used for longer periods of time. Under such conditions, the same GA optimized current profile for 40°C is tested once again, and the results in Fig. 8 indicate a dramatic increase in torque loss to 2. This has to be expected, since from the explanation given in section II, it follows that the change in temperature causes a subsequent change in oil properties thereby producing different flow for the same input current in the hydraulics section. This ofcourse causes a drift in the intermediate pressure profile thereby affecting the pressure to torque and slip relation.

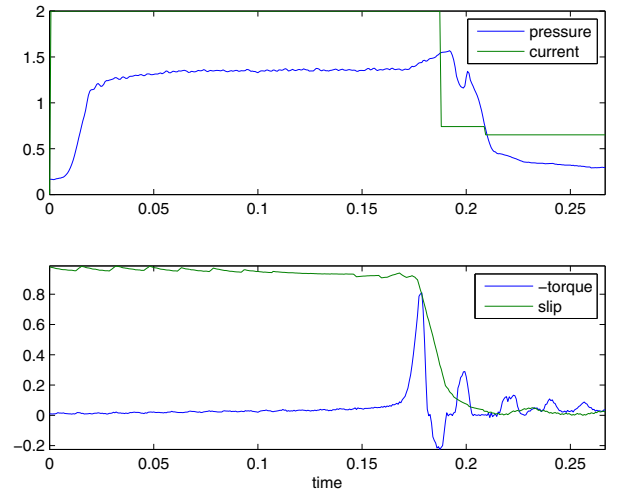


Fig. 7. GA feedforward 40°C

VI. ROBUST PREDICTIVE CONTROL OF ENGAGEMENT

The bandwidth of the system between the input current to the valve and the pressure to the clutch is much higher in the slipping phase then in the filling phase. There can be added disturbances on the pressure due to variation in temperature causing a change in flow. The model of the clutch changes nonlinearly between the various phases of its operation, and since the controller is based on the model of the filling phase, it must be also robust to such plant variations. This translates to robust performance requirements as per control theory literature. Robustness and optimality are inherent to MPC like in the Extended Prediction Self-Adaptive Control (EPSAC) strategy [9].

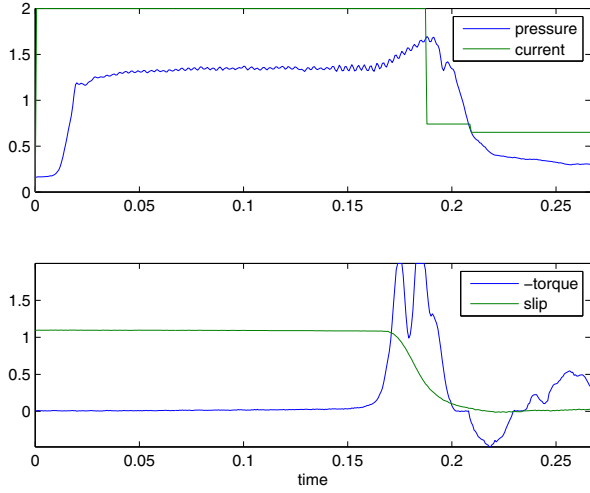


Fig. 8. GA feedforward 50°C

A. EPSAC

The process is modelled as:

$$y(t) = x(t) + n(t) \quad (8)$$

with $y(t)$, $x(t)$, $n(t)$ as process output, model output, process/model disturbance respectively. The fundamental step is based on the prediction over the basic process model given by:

$$y(t+k|t) = x(t+k|t) + n(t+k|t) \quad (9)$$

where $y(t+k|t)$ is the prediction of process output at time instant t , over prediction horizon $k = N_2$, based on prior measurements and postulated values of inputs. Prediction of model output $x(t+k|t)$ and of colored noise process $n(t+k|t)$ can be obtained by the recursion of process model and filtering techniques respectively.

The future response can then be expressed as:

$$y(t+k|t) = y_{base}(t+k|t) + y_{optimize}(t+k|t) \quad (10)$$

The two contributing factors have the following origins:

- $y_{base}(t+k|t)$ is the cumulative effect of past control inputs $u(t-1)$, $u(t-2)$..., base future control scenario $u_{base}(t+k|t)$ which is defined a priori and predicted disturbances $n(t+k|t)$.
- $y_{optimize}(t+k|t)$ is the effect of optimizing future control actions $\delta u(t|t)$, ..., $\delta u(t+N_u-1|t)$ with $\delta u(t+k|t) = u(t+k|t) - u_{base}(t+k|t)$.

The design parameter N_u is the control horizon. Further, the colored noise can be modeled as:

$$n(t) = \frac{C(q^{-1})}{D(q^{-1})}e(t) \quad (11)$$

This disturbance model must be designed to achieve robustness of the control loop against unmeasured disturbances and modelling errors. These disturbances have a stochastic nature with non-zero bias. $e(t)$ is white noise, C and D are

polynomials in backward shift operator q^{-1} .

The optimizing signal can be expressed as a discrete time convolution of impulse response coefficients $h_1, h_2, \dots$ and step response coefficients $g_1, g_2, \dots$ of the system as follows:

$$y_{optimize}(t+k|t) = h_k \delta u(t|t) + h_{k-1} \delta u(t+1|t) + \dots + g_{k-N_u+1} \delta u(t+N_u-1|t) \quad (12)$$

Using (10) and (12), the key EPSAC-MPC formulation becomes:

$$Y = \bar{Y} + GU \quad (13)$$

where Y , $\bar{Y}$ are the vector of outputs, base responses through N_2 respectively. U is the vector of future controls upto N_u-1 . G is a special matrix with impulse and step response coefficients with dimension upto $(N_2 - N_1 + 1) \times N_u$. The optimal control is then obtained by minimizing the following cost function:

$$V(U) = \sum_{k=N_1}^{N_2} [r(t+k|t) - y(t+k|t)]^2 + \lambda \sum_{k=0}^{N_u-1} [u(t+k|t)]^2 \quad (14)$$

with λ being the control penalty and $r(t+k|t)$ the desired reference trajectory. N_1 is the process dead time. A closed form solution can be obtained for the unconstrained case to be:

$$U^* = [G^T G + \lambda I]^{-1} [G^T (R - \bar{Y})] \quad (15)$$

In case of input and output linear inequality constraints, an active set strategy for the quadratic programming formulation can be used. It maintains a set of active equality constraints and then at each iteration solves for an explicit solution using Lagrange multipliers and updates the active set again and continues till convergence [10].

Only the first optimal control input is applied to the plant and the whole procedure is repeated at the next sampling instant $t+1$.

VII. RESULTS

The robustness can be guaranteed by designing the disturbance model $n(t)$, or by just considering it as an integrated white noise, which works well in most of the cases, including here. However, in principle higher order filters may be used, when explicit knowledge of uncertainty is available. The following constraints are imposed:

$$0 \leq \text{Current}(\text{Input}) \leq 2 \quad (16)$$

$$0 \leq \text{Pressure}(\text{Output}) \leq 1.4 \quad (17)$$

The desired performance under this setting is obtained by choosing prediction horizon to be $N_2 = 8$ and control horizon $N_u = 1$ with control penalty $\lambda = 0$. Such a controller gives an ideal tradeoff between bandwidth and gain/phase margin.

The controller would now be tested under exactly the same conditions as in section V, i.e. at two different temperature settings. As pointed out in section IV of the LPC scheme, the controller would track the pressure profile, which was recorded in section V, as a by product of the GA optimized

current profile at 40°C for minimal torque loss and engagement time specifications.

The first experiment is performed at 40°C and the results plotted in Fig. 9 already show an improvement over the feedforward engagement of Fig. 7. Though the engagement time is a little over 0.25, which can be explained by the phase lag of the controller in closed loop, the torque loss stands at 0.2, which is 4 times less than before. This could be explained by the fact that even if the temperature of the experiment and optimization are same, there are several other factors in the setup which could have changed like the friction parameters and other unknown disturbances impacting the inner hydraulics loop. These are rejected faster by the controller in place.

The next experiment is conducted by again making the clutch work to reach 50°C , however the pressure profile optimized at 40°C would be continued to be used as the reference. The results are shown in Fig. 10. Notice that the pressure peak which existed in the feedforward results of Fig. 8 are no longer present between the end of filling and beginning of slip phase i.e. time 0.2 in the plot. This would of course ensure a smoother engagement, which is quiet evident from the torque loss which is again maintained at its previous value of 0.2, i.e. a 10 times improvement over feedforward scheme. The engagement time is again just over 0.25.

This proves that the learning predictive controller is robust against temperature variations in the plant. This is because the same current profile optimized at 40°C could no longer reproduce the optimal pressure profile, due to temperature changing the flow dynamics, leading to higher torque loss. However, the robust controller could modify the control input current in order to track the optimal pressure profile under temperature parameter variations.

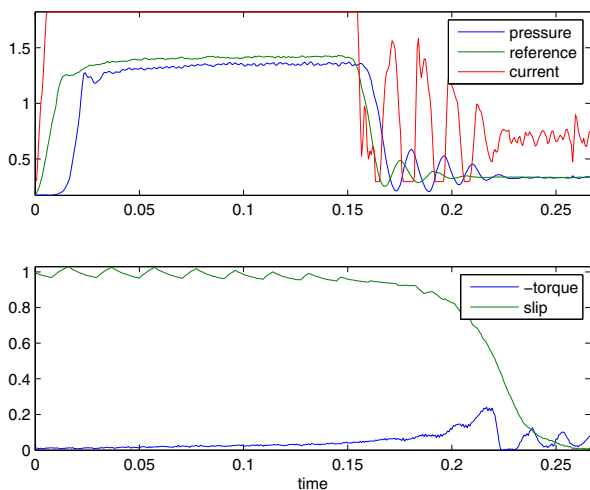


Fig. 9. MPC 40°C

VIII. CONCLUSIONS

Hydraulic clutch transmissions pose huge modelling challenge. Therefore, as a first step an evolutionary algorithm is

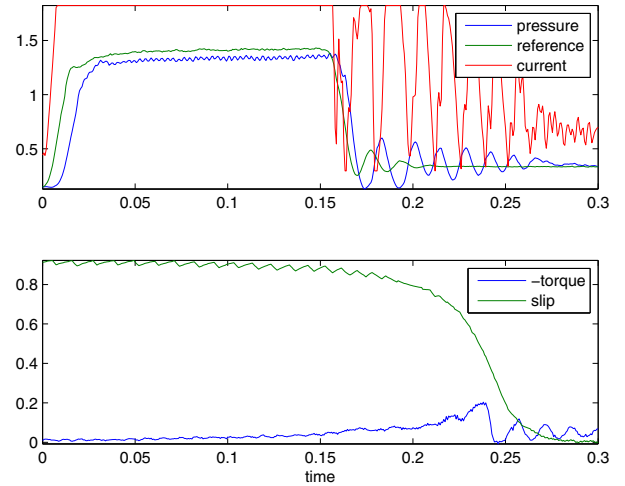


Fig. 10. MPC 50°C

used to find an optimal reference. However, such machines show high time variance in dynamics, therefore a predictive controller is used as a second step to track this reference over an identified linear sub-model with robust performance guarantees. This combination is introduced as Learning Predictive Control and shown to work successfully on a real setup.

IX. ACKNOWLEDGEMENTS

Abhishek Dutta and Yu Zhong are supported within the framework of the LeCoPro project (grant nr. 80032) of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen).

REFERENCES

- [1] A.C.Van der Heijden et.al., 'Hybrid optimal control of dry clutch engagement. Int. Journal of Control, pp 1717-1728, 2007.
- [2] C.Lazar et.al. Modelling and predictive control of an electro-hydraulic actuated wet clutch for automatic transmission. IEEE Symposium on Industrial Electronics, 2010.
- [3] J.Horn et.al. Flatness-based clutch control for automated manual transmissions. Control Engineering Practice, pp 1353-1359, 2003.
- [4] Z. Sun, K. Hebbale, Challenges and opportunities in automotive transmission control. American control conference. 2005, 3284-3289.
- [5] M.H.M.Dassen. Modelling and control of automotive clutch systems. Report number 2003.73, Eindhoven 2003.
- [6] The Mathworks Inc. Building a clutch lock-up model. Webinar, 2011.
- [7] M.J.W.H.Edelaar. Modeling of a wet plate clutch in a driveline. WFW Report 96.071, Eindhoven May 1996.
- [8] G. Pinte, B. Depraetere, W. Symens, J. Swevers, P. Sas, Iterative learning control for the filling of wet clutches. Mechanical systems and signal processing. 2010, 1924-1937
- [9] De Keyser, R. Model based predictive control. Invited chapter in UNESCO EoLSS, Article contribution 6.43.16.1, Oxford, 2003.
- [10] L. Wang. Model predictive control system design and implementation using matlab. Springer-Verlag, 2009.

Leakage-enhanced 3D-Stacked NEMFET-based Power Management Architecture for Autonomous Sensors Systems

Marius Enachescu, George R. Voicu, and Sorin D. Cotofana

Abstract—With the technology moving into the deep sub-100 nm region, the increase of leakage power consumption necessitates more aggressive power reduction techniques using emerging devices. Power gating with Nano-Electro-Mechanical Field Effect Transistors (NEMFET) is a promising avenue to reduce energy consumption of embedded autonomous sensor systems. Our research emphasizes that 3D Stacked hybrid circuits with NEMFET sleep transistors can be further enhanced to reduce leakage power by redesigning the entire power management circuitry with NEMFETs. To evaluate the practical implications of such an approach we implement NEMFET based power gating, which makes use of NEMFETs as sleep transistors, isolation cells, and components for power management controller design, on an embedded SoC platform running a bio-medical sensing application. Preliminary energy evaluations with Cadence EDI flow indicate that the enhanced architecture provides a reduction of 7% over the 3D Hybrid architecture at the expense of 4.7% area increase and of about 15% energy reduction with respect to the "classic" 2D CMOS counterpart. Furthermore, for applications with lower activity, the potential energy improvement the enhanced architecture could provide can reach up to 90% with respect to the 2D CMOS reference design.

I. INTRODUCTION

CMOS technology is approaching its physical limits; hence emerging technologies have been investigated and proposed to supersede the basic CMOS device, i.e., the MOSFET. Examples of such devices are Nano-Electro-Mechanical FETs (NEMFETs), carbon nano tubes, quantum-dot cellular automata, single-electron devices [1], [2], [3], [4]. However, at this time, only low complexity circuits were successfully designed with the above technologies.

On the other hand, scaling CMOS technology could still provide 30% improvement in performance at 28nm node with respect to 40nm node according to [5]. The price to be paid reflects also into the tremendously increase of power dissipation of digital CMOS. Hence, the life of the battery operated embedded systems with low activity, e.g., autonomous sensor systems, which is mostly determined by the standby leakage power dissipated by the circuit in sleep mode, is directly affected. Thus, the use of power gating

technique in portable devices is vital in achieving long life, low energy implementation solutions.

Power gating works on the principle of using Sleep Transistors (STs) to cut off supply voltage to the circuit in standby mode thus cutting off leakage current path to ground. Various enhancements based on Multi-Threshold CMOS (MTCMOS) technologies have been proposed to suppress the sleep transistor leakage current [6], however the NEMFET based ST proved to be more effective when compared with High- V_T counterpart [7].

One solution that partially address the leakage issue by exclusively minimizing the leakage of the ST is a hybrid CMOS-NEMS 3D stacked chip architecture. Benefitting from the NEMS-based power management, the architecture proposed in [8], emphasizes the outstanding characteristics of NEMFET devices, i.e., "abrupt" switching and ultra low leakage. To validate this proposal and evaluate its performance in a real-life scenario a careful assessment of the implications of this hybrid power management architecture on the rest of the system was described in [9].

Since power gated circuits spend very long times in standby mode, one can essentially infer that the overall energy can be further substantially reduced when the overall OFF state power consumption mostly generated by the AO cells is mitigated. In this paper we propose a novel 3D power management approach that attempts to alleviate some leakage overhead associated with the use of CMOS devices as Always-On (AO) cells, in isolation (ISO) cells and in the Power Management (PM) controller. Our proposal relies on: (i) moving the ISO cells and the PM controller on the NEMS die, and (ii) redesigning them in the NEMS technology to take advantage of the NEMFET ultra low leakage power. Our goal is to explore these implications and evaluate the performance and the energy efficacy of this power management architecture in a real-life scenario.

Our experiments indicate that, due to the extreme low leakage and the "abrupt" switching of the NEMFET, the system idle energy is decreased by 6.35x at the expanse of a 4.75% area overhead on the NEMS tier. Moreover, due to the leakage-enhanced 3D hybrid approach the energy-delay product of the embedded SoC platform is reduced by 7% with a potential improvement of up to 90% with respect to the 2D CMOS reference design.

Even though the NEMFET based AO cells add overhead to the total area, this does not affect the chip footprint since it is displaced to another tier (which contains NEMFET sleep transistors). Moreover, since the AO cells are not on the critical path, their replacement with the NEMFET-based cells,

This work has been jointly supported by the EU commission via the 7th Framework Programme project NEMSIC, reference 224525, and by Catrene and Agentschap NL via CT105 project 3DIM³.

M. Enachescu is with Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2628 CD Delft, The Netherlands M.Enachescu@tudelft.nl

G. R. Voicu is with Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2628 CD Delft, The Netherlands G.R.Voicu@tudelft.nl

S. D. Cotofana is with Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2628 CD Delft, The Netherlands S.D.Cotofana@tudelft.nl

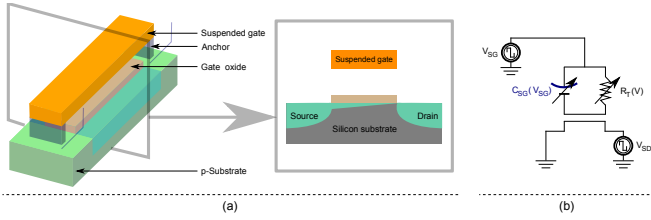


Fig. 1 (a) Schematic diagram of NEMFET. (b) Equivalent circuit model for the NEMFET.

which for the time being have a limited operation frequency of about 100 MHz, do not change the overall frequency of our approach. Exploiting 3D stacked hybrid integration, our approach can be applied directly on an industrial low power design which supports NEMS and CMOS technology.

The rest of the paper is organized as follows. First, in Section II, we give an overview of the 3D hybrid power management architecture and some relevant NEMFET background. Section III describes the enhanced-leakage architecture, which makes use of NEMFETs for the implementation of the entire power gating related infrastructure, i.e., sleep transistors, isolation cells, and power management controller. Section IV discusses the experimental results and finally, Section V presents the conclusions.

II. NEMFET BASED POWER MANAGEMENT ARCHITECTURE

A. NEMFET Background

The Nano-Electro-Mechanical FET (NEMFET) described in [10] is a rather complex device with a 3D geometry and cross-section as presented in Figure 1a.

Essentially speaking the device behaves like an electromechanical switch which responds to gate bias changes as follows. When the gate voltage V_G is low the gate-oxide capacitance is in series with the air-gap capacitance Figure 1b resulting in low electrostatic coupling of the gate to the channel thus in a negligible drain current I_D . If V_G increases the situation remains unchanged until it reaches a certain "ON" voltage case in which the electrostatic force cannot be compensated anymore by the mechanical restoring force and the suspended gate (beam) snaps onto the gate oxide, thus turning on the device. This is called pull-in effect and corresponds to a sudden I_D increase. After the pull-in, the I_D increase with V_G is comparable with the one of a standard MOSFET. On the other way around when V_G is decreased from some high value I_D starts decreasing until at a certain V_G value the system becomes unstable due to combined electro-mechanical force and the beam is pulled-out. This causes an abrupt I_D decrease due to a large decrease in capacitance and it is called pull-out effect.

As indicated by [7] NEMFET devices can potentially replace High- V_T sleep transistors due to their ultra low leakage characteristics. Note that for power gated designs two major features are desirable for the sleep transistors: (i) low sub-threshold leakage current to minimize the static power consumption, and (ii) low "ON" state resistance to

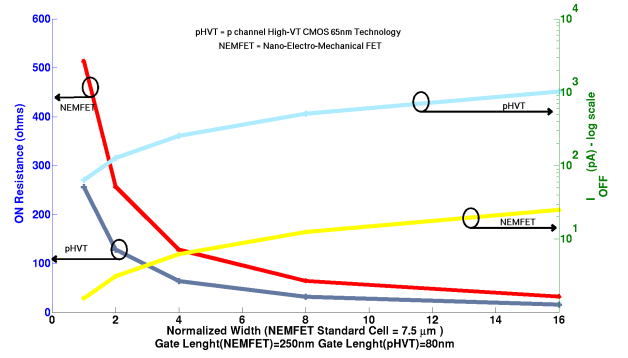


Fig. 2 R_{ON} and I_{OFF} for NEMFET and 65 nm High- V_T CMOS Switch Transistors.

minimize the voltage difference between the virtual and the real power supply nodes.

A comparison between the "ON" state resistance R_{ON} and "OFF" state leakage current I_{OFF} of NEMFET and 65 nm High- V_T CMOS based Sleep Transistors (STs) is presented in Figure 2 for different area in terms of standard cells. The R_{ON} values are computed for an IR-drop of 10 mV over the ST, and the I_{OFF} values consider a 1.2 V power supply. One can observe that when the ST area is increasing the NEMFET ST R_{ON} tends to become equal with the CMOS ST R_{ON} , while NEMFET ST I_{OFF} (leakage) is about 2 orders of magnitude smaller than the one of the CMOS counterpart.

B. 3D-Stacked Hybrid Power Management Architecture

In [9] the integration impact that a recent proposed 3D stacked power management architecture based on Nano-Electro-Mechanical FET power switches may have on a real-life embedded SoC design was described. The architecture relies on NEMS technology dies containing NEMFET STs placed on top of the die containing the actual power gated circuits, as in the two-tier stack test vehicle from Figure 3. The bottom CMOS tier comprises the active logic circuit, while the top tier incorporates the NEMFET STs. The dies are stacked using Through-Silicon Vias (TSVs), allowing for large density, high speed, and low power interconnect [11].

The approach has a number of advantages as follows: (i) it combines the appealing extremely low leakage currents of the NEMFETs with the versatility of CMOS technology by allowing for the power switches to be fabricated on a separate die, (ii) it simplifies the floorplanning in general, and can increase the computation platform performance because of extra area cleared by the switches on the CMOS tier, and (iii) it leverages the integration of other NEMS/MEMS devices, e.g., energy harvesters, sensors, on the same tier with the power management circuitry.

The test vehicle has three power domains (PD Core, PD DMem and PD PMem), which can be independently powered-off by the corresponding group of sleep transistor on top of the power domain. The Always-On (AO) power gating related blocks, contained in the PCM power domain are part of the bottom die. The next section presents a way to reduce the leakage power by the relocation of the

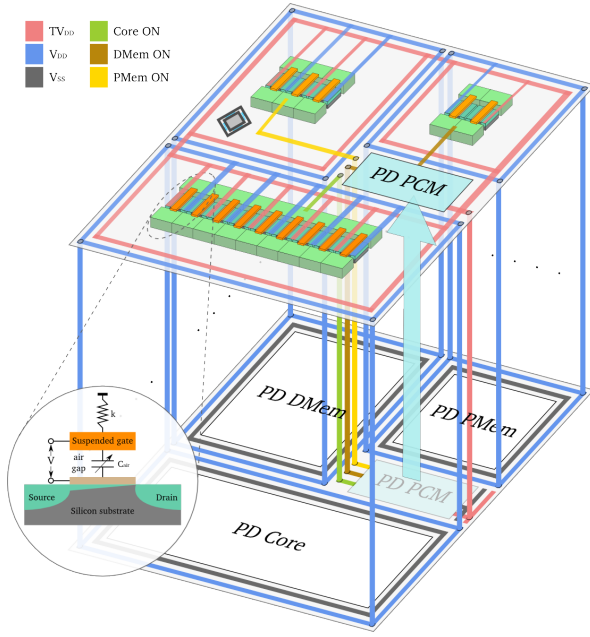


Fig. 3 Leakage-enhanced 3D Stacked NEMFET Power Management Architecture.

PCM power domain, as displayed by the light blue arrow in Figure 3.

III. LEAKAGE-ENHANCED 3D-STACKED NEMFET-BASED POWER MANAGEMENT ARCHITECTURE

We propose to enhance the above presented PM architecture by moving the Always-On (AO) power domain location to the NEMS die. This entails redesigning the isolation (ISO) cells and the PM controller in the NEMS technology. This translation alleviates part of the leakage overhead associated with the use of CMOS devices as AO cells by taking advantage of the NEMFET ultra low leakage power. The bottom CMOS tier comprises the active logic circuit, while the top tier incorporates the STs, ISO cells, and PM controller implemented with NEMFETs.

The Always ON cells include the following elements:

- 1) Isolation cells which controls the outputs of powered down blocks and clamp the outputs to a specific, legal value;
- 2) The Power Management Controller, which generates the control signals for the STs and the ISO cells in the appropriate sequence for shut-down and power-up transitions.

A. Isolation cells

Floating outputs of the power-down blocks altogether with crowbar currents during powering down may result in spurious behavior in the power-up blocks inputs [6]. Isolation cells are practically basic NAND/NOR gates, or even single pull-up/pull-down transistors with the gate input acting as the isolation signal generated by the PM controller that can hold the output to logic high/low.

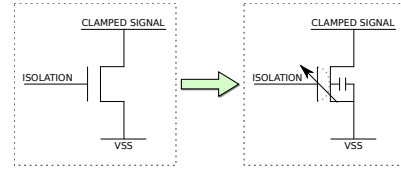


Fig. 4 Substitution of nMOS Pull-down Isolation cell with NEMFET cell.

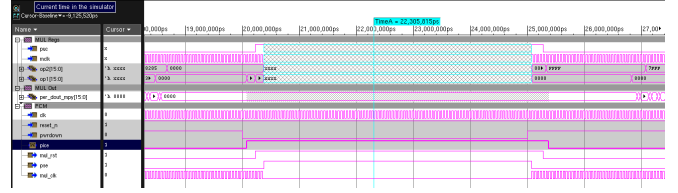


Fig. 5 Simulated waveform of Power Management Controller signals.

By replacing the isolation cells with NEMFET pull-up/pull-down transistors as indicated in Figure 4 we take advantage of the ultra-low leakage characteristic of the NEMFET to reduce with approximately two orders of magnitude the additional energy cost incurred by isolation. The twofold trade-offs are area and delay penalties. Taking into account that the operating frequency of a NEMFET based logic is currently limited to about 100 MHz and the parasitic TSV interconnect capacitance we can expect that the isolation sequence will increase from one to two clock cycles.

B. Power Management Controller

This module performs the following sequence of operations to power-down:

- 1) Stop the clock to minimize leakage into the power-gated region;
- 2) Assert the isolation control signal to park all outputs in a safe condition;
- 3) Assert reset to the block, so that it powers up in the reset condition;
- 4) Assert the power gating control signal to power down the block.

To restore power the same sequence is performed in the reversed order. The PM control signals for a complete power-down/power-up cycle simulated in Cadence LP-NCSim are depicted in Figure 5.

IV. PERFORMANCE EVALUATION

To evaluate the energy savings of the 3D stacked NEMFET based Power Management (PM) architecture an energy analysis was first performed on the Reference single-tier design with classic High- V_T sleep transistors. Subsequently, by moving the bulky STs to a different die, the Stacked design, precious area surrounding/inside the gated blocks previously allocated to them in the planar case can be reclaimed. Finally, by substituting the High- V_T STs with the equivalent area of NEMFET STs, the Hybrid implementation was analyzed.

Figure 6 presents the considered experimental platform. It consists of a typical SoC for low power embedded devices, based on the open-source 16-bit synthesizable processor core openMSP430 [12], a clone of the commercial Texas

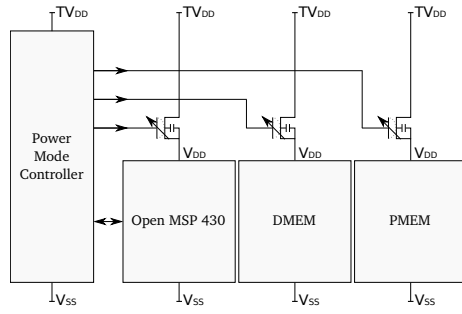


Fig. 6 System on Chip platform for autonomous sensors.

Instruments MSP430 microcontroller. The program memory (PMEM) and data memory (DMEM) sizes are 4 kB and 2 kB, respectively. Although not used by the reference application, all the peripherals of the openMSP430 core (16-bit multiplier, timer, 6 8-bit input/output ports) are part of the platform in order to account for their leakage power.

The reference application is a heart beat rate monitor which detects the QRS complex in an digitized electrocardiogram (ECG) signal. The QRS event corresponds to the depolarization of the ventricles, comprising a series of three deflections seen on a typical ECG signal. The middle one, i.e. the R wave, is the well-known “peak” of the ECG. The application program identifies the R peaks in the ECG signal and measures the interval between two consecutive R peaks. The algorithm we use is based on the open source arrhythmia detection software from EP Limited [13], which uses the filtering based Pan-Tomkins [14] method for R peak detection. For this method the input data are channeled through a five steps filtering process consisting of a low pass filter, a high pass filter, a derivative, an absolute value, and an integrator function, in this order. The utilized filter equations are valid for an ECG sample frequency of 200 Hz.

A. 3D-Stacked NEMFET-based Power Management Architecture evaluation

The reference platform was implemented using Cadence Encounter Digital Implementation [15] in a 65 nm commercial Low Power CMOS technology. A high operating frequency of the SoC means that it has to be synthesized with tighter timing constraints, which results in a design with faster and/or bigger cells. As a consequence leakage and dynamic power increase. For the presented SoC this effect limits the operating frequency to 150 MHz without causing a significant increase in the leakage power.

The obtained energy consumption values for the three designs mentioned at the beginning of this section are presented in Table I. One can observe that in spite of the reduced OFF leakage power the total improvement on the energy-delay product figure is only 9%. This happens because of two reasons: (i) the test application has a too high duty cycle, and (ii) the ON power term is dominant in the total energy equation.

The power consumption could be further reduced in two ways:

TABLE I Power and energy results

Implementation	ON power		OFF power [nW]		Energy
	[mW]	STs	Always-on cells*	Total	[μJ]
<i>Reference</i>	5.0340	56.28	26.02	81.28	9.05
<i>Stacked</i>	5.0343	900.44	26.02	925.44	9.46
<i>Hybrid</i>	5.0339	4.52	26.02	30.54	8.61

* Always-on cells = Power management controller and isolation cells

- addressing the OFF state power through implementing the PM controller and the isolation cells with NEMFET devices as discussed in Section III,
- optimizing the design in terms of ON state power and the software application to optimally utilize the hardware resources, hence reducing the run time.

B. Leakage-enhanced 3D-Stacked NEMFET-based Power Management Architecture evaluation

The proposed enhanced 3D power management hybrid approach makes use of the first option and redesigns the entire extra power gating logic using low-power NEMFET technology and place it on the different tier, taking advantage of the 3D stacking technology. We compare the enhanced architecture with the hybrid one and evaluate the overall energy gain from the embedded system running the biomedical application perspective. Using the same power analysis methodology we extracted the power and area values presented in Table II.

Even though the total area overhead due to logic implementation using NEMFETs is 4.75% of the design size, this does not affect the chip footprint since it is displaced to another tier (which contains NEMFET switch transistors). The prominent reduction of about two orders of magnitude in the OFF state power is due to the practical “zero-leakage” characteristic of NEMFET. Furthermore, the abrupt switching effect of the NEMFET causes a reduction in the dynamic power of PM controller and ISO cells, reducing the total ON state power. The changes in active and idle power are directly reflected in the energy figure, improving the energy consumption by 7% with respect to the hybrid design. Compared with the classic one-tier power-gated CMOS implementation Table I the overall energy saving could be up to 15%.

Figure 7 presents a comparison of the system idle power for the classic High- V_T Reference, i.e., the 3D Hybrid, and the proposed Leakage-enhanced architectures. The idle power is equal with the leakage power of the only active components in the idle state, the STs and the AO cells. A two-step approach to reduce the leakage power can be observed. Hybrid design implements in the first step only the STs in the NEMFET die, which results in a substantial reduction of the ST leakage but doesn’t affect the AO cell figure of merit. In the second step, the Leakage-enhanced design contains all STs, AO cells and the complete PM circuit on the top die, implemented with NEMFETs. In total, by using the Leakage-enhanced architecture, we reduce the

TABLE II Leakage-enhanced architecture area, power and energy results

Implementation type	Area [% of die size]		ON power [mW]	STs	OFF power [nW]		Total energy* [μ J]
	ISO cells	PM Controller			ISO cells	PM Controller	
Hybrid [9]	0.15	0.02	5.0339	4.52	18.01	8.01	8.61
Leakage-enhanced	3.75	1.17	4.7	4.52	0.289	0.075	8.00

* Active time=1.73 ms, Idle time=993.35 ms, Transition time=5.32 ms @ 150 MHz.

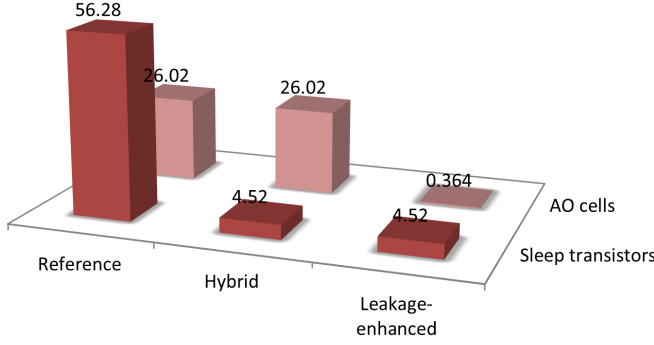


Fig. 7 Breakdown of leakage power (in nW) in an embedded processor for autonomous sensors.

leakage power in the idle state with 92%, from 82.3 nW in the 2D Reference design to 5.884 nW.

The ON state power consumption can be further reduced by optimizing the design in terms of power and the software application to optimally utilize the hardware resources, thus reducing the run time. To characterize the energy consumption efficiency versus various applications, we plot in Figure 8 the platform energy consumption for different operating duty cycles. We assume the same ON and OFF state power figures as for the bio-medical application. The duty-cycle is defined as: $T_{ON}/(T_{ON} + T_{OFF})$.

One can observe in the figure that the energy requirement has a steep decrease until the application duty-cycle reaches 0.0001. Optimizing and reducing the application duty-cycle lower than this does not bring much energy savings any longer. However, from this point further, for even lower activity applications the relative savings in energy consumption due to our power architecture when compared to High- V_T STs start to increase, reaching up to 90% lower energy consumption. Compared to the previous 3D-Stacked Hybrid PM architecture the overall energy improvement is about up to 50% higher.

V. CONCLUSIONS

In this paper, we diminished the leakage power of the NEMFET based 3D-stacked power management architecture by making use of NEMFETs also for the implementation of isolation cells, and power management controller design. We evaluated the practical implications of such an approach by implementing an embedded SoC platform running a bio-medical sensing application with NEMFET based power gating. We performed energy evaluation of the enhanced design and our results indicated a reduction of 7% over the

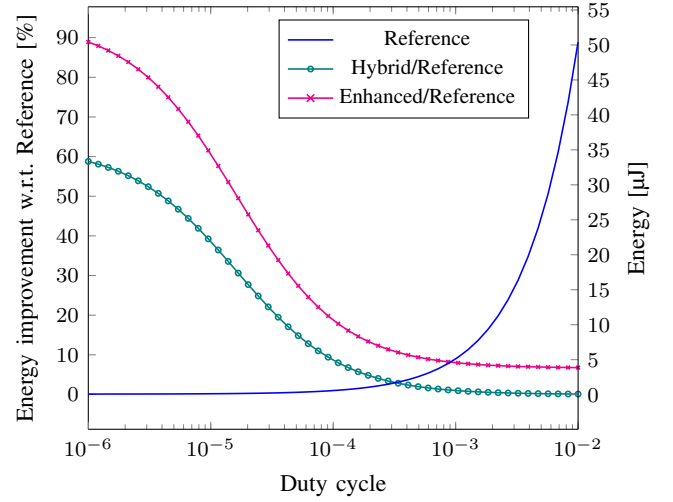


Fig. 8 Energy Consumption versus Duty-cycle.

3D Hybrid architecture with the price of 4.7% increase in area and of about 15% energy reduction with respect to the "classic" 2D CMOS counterpart. Furthermore, for applications with lower activity, the potential energy improvement of the Leakage-enhanced architecture could deliver can reach up to 90%, with respect to the 2D CMOS reference design. Our results suggest that when utilized in conjunction with CMOS the NEMFETs can induce significant performance improvements over CMOS only approaches in terms of energy efficiency. This justifies the need for further efforts in evaluating other aspects of the novel power management architectures, such as reliability and technology integration issues.

REFERENCES

- [1] D. Tsamados, Y. Singh Chauhan, C. Eggimann, K. Akarvardar, H. S. Philip Wong, and A. Mihai Ionescu, "Finite element analysis and analytical simulations of suspended Gate-FET for ultra-low power inverters," *Solid-State Electronics*, vol. 52, no. 9, pp. 1374–1381, 2008.
- [2] P. L. McEuen, M. S. Fuhrer, and H. Park, "Single-walled carbon nanotube electronics," *Nanotechnology*, *IEEE Transactions on*, vol. 1, no. 1, p. 78–85, 2002.
- [3] W. Lu and C. L. Lieber, "Nanoelectronics from the bottom up," *Nature Materials*, vol. 6, no. 11, p. 841–850, 2007.
- [4] A. Dehon, "Nanowire-based programmable architectures," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 1, no. 2, pp. 109–162, Jul. 2005.
- [5] J. Wei, "TSMC Technology Symposium - Leading Edge Technologies," TSMC, Amsterdam, NL, Tech. Rep., Jun. 2011.
- [6] *Low Power Methodology Manual*. Boston, MA: Springer US, 2007. [Online]. Available: <http://www.springerlink.com/index/10.1007/978-0-387-71819-4>

- [7] M. Enachescu, S. Cotofana, A. Genderen, D. Tsamados, and A. Ionescu, "Can SG-FET replace FET in sleep mode circuits?" *Nano-Net*, pp. 99–104, 2009.
- [8] M. Enachescu, G. Voicu, and S. Dan Cotofana, "Advanced NEMS-based power management for 3D stacked integrated circuits," in *2010 International Conference on Energy Aware Computing*, Cairo, Egypt, Dec. 2010.
- [9] G. Voicu, M. Enachescu, and S. Cotofana, "Towards "Zero-energy" using NEMFET-based power management for 3D hybrid stacked ICs," in *2011 IEEE/ACM International Symposium on Nanoscale Architectures*, San Diego, CA, USA, Jun. 2011, pp. 203–209.
- [10] A. M. Ionescu, V. Pott, R. Fritschi, K. Banerjee, M. J. Declercq, P. Renaud, C. Hibert, P. Fluckiger, and G. A. Racine, "Modeling and design of a low-voltage SOI suspended-gate MOSFET (SG-MOSFET) with a metal-over-gate architecture," in *Quality Electronic Design, 2002. Proceedings. International Symposium on*, 2002, pp. 496–501.
- [11] P. Garrou, *Handbook of 3D integration : technology and applications of 3D integrated circuits*. Weinheim: Wiley-VCH, 2008.
- [12] OpenCores, "openMSP430 softcore," 2011. [Online]. Available: <http://opencores.org/project,openmsp430>
- [13] EP Limited, "ECG analysis software," 2011. [Online]. Available: <http://www.eplimited.com/software.htm>
- [14] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Transactions on Bio-Medical Engineering*, vol. 32, no. 3, pp. 230–236, Mar. 1985, PMID: 3997178.
- [15] "Cadence design systems," 2011. [Online]. Available: <http://www.cadence.com/us/pages/default.aspx>

Relay Feedback Identification of SOPDT Transfer Functions with NMP Zero

Mona. Faraji-Niri, Mohammad Reza. Jahed-Motlagh, and Houman. Sadjadian

Abstract— In this paper a second order plus dead time transfer function with a NMP zero is identified. Through a single run of biased relay test exact analytical expressions are written for the unsymmetrical limit cycle and up to five parameters can be driven using fitting conditions simultaneously. By using both frequency and time domain features a clear algorithm is proposed which is applicable to identifying non-minimum phase systems including time delays and inverse responses.

I. INTRODUCTION

IN the recent years relay feedback has been used effectively for system identification. As a closed loop test it can provide a simple and fast way in which the process does not drift away from the set-point, which is very important specially in nonlinear processes. Additionally the relay feedback test can be performed automatically in online identification applications. Between different types of relay methods for identification, those with a single run are of high value because in these cases less time is needed to complete the test and find the parameters.

After the work of Astrom and Hagglund [1] on using the concepts of ultimate gain and frequency computed by the describing function approximation for system identification and autotuning, Luyben was the first who utilized the relay test to find the parameters of special classes of transfer functions [2].

Although relay test with describing function approximations for transfer function identification became a very popular tool, there was still need to make it a more accurate and general estimation method.

FOPDT models have always been one of the most popular ones in process control but they were applicable just for low or high order processes with over-damped responses. These transfer functions had negative time constant for some high order systems, so in some cases SOPDT models were better descriptions.

In order to improve precision of parameters and extend the

algorithms, Wang et.al. [3] proposed a way to derive parameters of a FOPDT and Majhi and Atherton [4] proposed methods in transfer function identification for both FOPDT and SOPDT [2], Majhi introduced a way for SOPDT identification based on state space [5].

In order to reduce the time of identification which is very important in online tuning, many efforts has been done to use a single run of relay test and improve it. Vivek and Chidambaram proposed a method for identification of FOPDT by Fourier analysis and Panda and Yu utilized time domain information of the limit cycle for a SOPDT model [2], [6], [7].

To solve the problem of computing process gain by relay feedback experiment, biased relay was used by Shen et. al, [8] also new algorithms were proposed by Chidambaram for FOPDT and Ramakrishnan and Chidambaram for SOPDT using the mentioned idea [2]. Liu and Gao gave a method for identification all of the parameters of a FOPDT, SOPDT and a more general form by a single run of biased relay test and fitting conditions in time domain [9], [10], [11]. In a recent paper Soltesz et. al. proposed a new identification method based on optimization for FOPDT [12].

In most of the cases, algorithms were useful just for transfer functions with up to four unknown parameters also minimum phase ones. A special class of non-minimum phase systems which have a NMP zero and repetitive poles is identified in [9] and a SOPDT with a time delay only, or NMP zero only, is discussed in [5], [9].

Already there is need to identifying a SOPDT with both time delay and NMP zero. This type of transfer function is useful for modeling a low or high order system with (over-damped, under-damped, or critically-damped) inverse response. The general form introduced in [9] can be considered as a SOPDT with similar poles and an inverse response, this is just applicable for systems having inverse and critically damped natures.

In this paper an algorithm for identification all the five parameters of a SOPDT with different poles and NMP zero is introduced, the parameters are derived simultaneously and the transfer function can prepare a precise model tuning and autotuning. In section two biased relay feedback test for identification is introduced, analytical expressions for limit cycle of mentioned transfer function in a biased relay test are derived in section three. Section four proposes an algorithm for identification, section five discusses model validation and finally illustrative examples showing effectiveness of the

M. Faraji-Niri is with Iran University of Science and Technology, Tehran, Iran (phone: +98-021-73225756; e-mail: faraji_mona@yahoo.com).

M. R. Jahed-Motlagh, is with Iran University of Science and Technology, Tehran, Iran (e-mail: jahedmr@iust.ac.ir).

H. Sadjadian is with Iran University of Science and Technology, Tehran, Iran (e-mail: hsadjadian@iust.ac.ir).

proposed method and a conclusion are given in sections six and seven respectively.

II. RELAY FEEDBACK TEST

As Astrom and Hagglund mentioned [1], a relay in the feedback path causes the closed loop system to oscillate around the setpoint when output lags behind input $-\pi$ radians. The information of the formed limit cycle can be used for identification and autotuning.

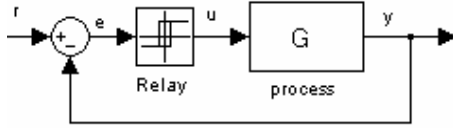


Fig. 1 Block diagram of relay feedback test

Biased relay test with hysteresis is shown in Fig. 1, in which r is the setpoint, y is the output of process and u is the output of relay.

Relay output is defined as (1).

$$u(t) = \begin{cases} u_+ & \text{if } e(t) > \varepsilon_+ \\ u_- & \text{if } e(t) < \varepsilon_- \end{cases} \quad (1)$$

Where $u_+ = \Delta\mu + \mu_0$ and $u_- = \Delta\mu - \mu_0$ are the amplitude of the relay, also ε_+ and ε_- are positive and negative switch hysteresis. μ_0 and $\Delta\mu$ are dependent on the admissible output range of the process and disturbance amplitude respectively, also the hysteresis is chosen as $\varepsilon_+ = -\varepsilon_-$ and such that incorrect switches be avoided in the presence of measurement noise. ε magnitudes are two or three times the noise band.

Transfer function of a SOPDT with NMP zero is written as

$$G(s) = \frac{K_p(-\tau_0 s + 1)e^{-\theta s}}{\tau_p^2 s^2 + 2\xi\tau_p s + 1} \quad (2)$$

where K_p is the process gain, θ is the time delay, $1/\tau_0$ is the zero, τ_p is the time constant, and ξ is the damping factor of the process.

In the presented work we focus on the case that the system has two complex poles and under-damped behavior, e.i. $0 < \xi < 1$. The outputs of the process and the relay are shown in Fig. 2. The relay test begins at $t=0$ and the output moves into steady oscillations after some switches, measurable parameters of the limit cycle are A_+ and A_- as the positive and negative amplitude of the process output, P_+ and P_- as the parts of the period corresponding to u_+ and u_- respectively, considering $P_u = P_+ + P_-$. P_u is defined as the oscillation period.

III. ANALYTICAL EXPRESSIONS

Analytical expressions for the output of (2) under biased relay test with a limit cycle such as fig. 2 can be derived in

steady oscillations where y_+ and y_- are parts of the output related to positive and negative switches of the relay proved in the appendix.

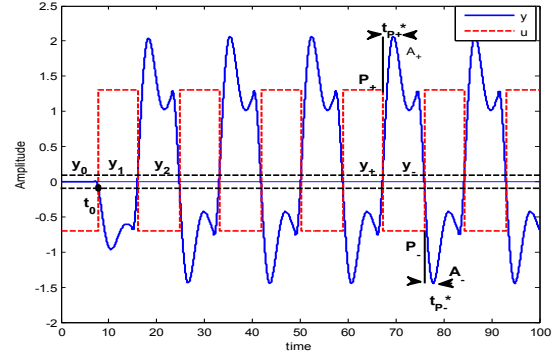


Fig. 2 Limit cycle of a SOPDT with a RHP zero transfer function

$$y_+(t) = K_p(\Delta\mu + \mu_0) - \frac{2K_p\mu_0\rho_1}{\eta\gamma} e^{-\frac{\xi}{\tau_p}t} \left\{ \left(\frac{\tau_p + \tau_0\xi}{\tau_p} \right) \times \right. \quad (3)$$

$$\left. \sin\left(\frac{\eta}{\tau_p}t + \psi_1\right) - \left(\frac{\tau_0}{\tau_p}\right) \cos\left(\frac{\eta}{\tau_p}t + \psi_1\right) \right\} \quad t \in [0, P_+]$$

$$y_-(t) = K_p(\Delta\mu - \mu_0) + \frac{2K_p\mu_0\rho_2}{\eta\gamma} e^{-\frac{\xi}{\tau_p}t} \left\{ \left(\frac{\tau_p + \tau_0\xi}{\tau_p} \right) \times \right. \quad (4)$$

$$\left. \sin\left(\frac{\eta}{\tau_p}t + \psi_2\right) - \left(\frac{\tau_0}{\tau_p}\right) \cos\left(\frac{\eta}{\tau_p}t + \psi_2\right) \right\} \quad t \in (P_+, P_u]$$

$$\text{Where } \eta = \sqrt{1 - \xi^2}, \quad \rho_1 = \sqrt{U_1^2 + V_1^2}, \quad \rho_2 = \sqrt{U_2^2 + V_2^2},$$

$$\psi_1 = \tan^{-1}(V_1/U_1), \quad \psi_2 = \tan^{-1}(V_2/U_2), \quad \varphi = \tan^{-1}$$

$$(\sqrt{1 - \xi^2}/\xi), \quad \gamma = 1 - 2e^{-P_u\xi/\tau_p} \cos(\eta P_u/\tau_p) + e^{-2P_u\xi/\tau_p}.$$

Also U and V are defined in (5) to (8).

$$U_1 = \cos\varphi - e^{-\frac{P_u\xi}{\tau_p}} \cos\left(\varphi - \frac{\eta P_u}{\tau_p}\right) - e^{-\frac{P_- \xi}{\tau_p}} \times \quad (5)$$

$$\left[\cos\left(\varphi + \frac{\eta P_-}{\tau_p}\right) - e^{-\frac{P_u\xi}{\tau_p}} \cos\left(\varphi - \frac{\eta P_+}{\tau_p}\right) \right]$$

$$V_1 = \sin\varphi - e^{-\frac{P_u\xi}{\tau_p}} \sin\left(\varphi - \frac{\eta P_u}{\tau_p}\right) - e^{-\frac{P_- \xi}{\tau_p}} \times \quad (6)$$

$$\left[\sin\left(\varphi + \frac{\eta P_-}{\tau_p}\right) - e^{-\frac{P_u\xi}{\tau_p}} \sin\left(\varphi - \frac{\eta P_+}{\tau_p}\right) \right]$$

$$U_2 = \cos \varphi - e^{-\frac{P_u \xi}{\tau_p}} \cos(\varphi - \frac{\eta P_u}{\tau_p}) - e^{-\frac{P_+ \xi}{\tau_p}} \times \quad (7)$$

$$\left[\cos(\varphi + \frac{\eta P_+}{\tau_p}) - e^{-\frac{P_u \xi}{\tau_p}} \cos(\varphi - \frac{\eta P_-}{\tau_p}) \right]$$

$$V_2 = \sin \varphi - e^{-\frac{P_u \xi}{\tau_p}} \sin(\varphi - \frac{\eta P_u}{\tau_p}) - e^{-\frac{P_+ \xi}{\tau_p}} \times \quad (8)$$

$$\left[\sin(\varphi + \frac{\eta P_+}{\tau_p}) - e^{-\frac{P_u \xi}{\tau_p}} \sin(\varphi - \frac{\eta P_-}{\tau_p}) \right]$$

IV. IDENTIFICATION ALGORITHM

Frequency response of a process can be estimated in any arbitrary point using a biased relay test. According to [9], (9) and (10) can be used in a single run of a biased relay test.

$$G(j\omega_u) = \frac{Y(j\omega_u)}{U(j\omega_u)} = \frac{\int_{t_{os}}^{t_{os}+P_u} y(t) e^{-j\omega_u t} dt}{\int_{t_{os}}^{t_{os}+P_u} u(t) e^{-j\omega_u t} dt} = A_u e^{j\varphi_u} \quad (9)$$

$$\begin{aligned} G(\alpha + j\omega) &= \frac{(1 - e^{-P_u(\alpha + j\omega)}) \int_{t_{os}}^{t_{os}+P_u} [y(t) e^{-\alpha t}] e^{-j\omega t} dt + \int_{t_{os}}^{t_{os}+P_u} [y(t) e^{-\alpha t}] e^{-j\omega t} dt}{(1 - e^{-P_u(\alpha + j\omega)}) \int_{t_{os}}^{t_{os}+P_u} [u(t) e^{-\alpha t}] e^{-j\omega t} dt + \int_{t_{os}}^{t_{os}+P_u} [u(t) e^{-\alpha t}] e^{-j\omega t} dt} \quad (10) \\ &= A_\alpha e^{j\varphi_\alpha} \end{aligned}$$

t_{os} is any time related to relay switches and σ is the estimation precision which is usually between 10^{-4} to 10^{-6} so α can be chosen such that the (11) is met where T_s is the sample time [6].

$$\alpha > \sigma, \quad \min \left\{ \left| u(t_{os} + P_u) e^{-\alpha(t_{os} + P_u) T_s} \right|, \left| y(t_{os} + P_u) e^{-\alpha(t_{os} + P_u) T_s} \right| \right\} > \sigma \quad (11)$$

Process gain can be computed by using multiple point frequency response estimation done through (12). Where M_0 is any even number for computing FFT, and $FFT\{ \cdot \}_{t=0}$ is the first element corresponding to ω_0 of FFT series for $\omega_k = k\omega_u / M_0, k = 0, 1, 2, \dots, N_0 - 1, N_0 = M_0 P_u / T_s$ [6].

$$G(0) = FFT \left\{ FFT^{-1} \left\{ G(\alpha + j\omega_k) \right\} e^{\alpha k T_s} \right\}_{t=0} \quad (12)$$

Equations (13) and (14) can be written by substituting (2)

into (9).

$$A_u = K_p \frac{\sqrt{\tau_0^2 \omega_u^2 + 1}}{\sqrt{(1 - \tau_p^2 \omega_u^2)^2 + 4\xi^2 \tau_p^2 \omega_u^2}} \quad (13)$$

$$\theta = -\frac{1}{\omega_u} [\varphi_u + \tan^{-1}(\tau_0 \omega_u) + \tan^{-1}(\frac{2\xi \tau_p \omega_u}{1 - \tau_p^2 \omega_u^2})] \quad (14)$$

Output derivatives can be written by (3) and (4) which are respectively as (15) and (16).

$$\begin{aligned} \frac{dy_+}{dt} &= \frac{-2K_p \mu_0}{\eta \gamma \tau_p} \rho_1 \left(\frac{\tau_p + \tau_0 \xi}{\tau_p} \right) e^{-\frac{\xi}{\tau_p} t} \left(\eta \cos \left(\frac{\eta}{\tau_p} t + \psi_1 \right) \right. \\ &\quad \left. - \xi \sin \left(\frac{\eta}{\tau_p} t + \psi_1 \right) \right) + \frac{2K_p \mu_0}{\eta \gamma \tau_p} \rho_1 \left(\frac{\tau_0}{\tau_p} \right) e^{-\frac{\xi}{\tau_p} t} \\ &\quad \left(-\eta \sin \left(\frac{\eta}{\tau_p} t + \psi_1 \right) - \xi \cos \left(\frac{\eta}{\tau_p} t + \psi_1 \right) \right) \quad (15) \end{aligned}$$

$$\begin{aligned} \frac{dy_-}{dt} &= \frac{2K_p \mu_0}{\eta \gamma \tau_p} \rho_2 \left(\frac{\tau_p + \tau_0 \xi}{\tau_p} \right) e^{-\frac{\xi}{\tau_p} t} \left(\eta \cos \left(\frac{\eta}{\tau_p} t + \psi_2 \right) \right. \\ &\quad \left. - \xi \sin \left(\frac{\eta}{\tau_p} t + \psi_2 \right) \right) - \frac{2K_p \mu_0}{\eta \gamma \tau_p} \rho_2 \left(\frac{\tau_0}{\tau_p} \right) e^{-\frac{\xi}{\tau_p} t} \\ &\quad \left(-\eta \sin \left(\frac{\eta}{\tau_p} t + \psi_2 \right) - \xi \cos \left(\frac{\eta}{\tau_p} t + \psi_2 \right) \right) \quad (16) \end{aligned}$$

If $\frac{dy_+}{dt} = 0$ and $\frac{dy_-}{dt} = 0$ the time to reach the peak of the limit cycle namely t_{P+}^* and t_{P-}^* can be derived.

$$t_{P+} = \frac{\tau_p}{\eta} (\varphi_1 - \psi_1 + k\pi) \quad k = 0, 1, \dots \quad (17)$$

$$t_{P-} = \frac{\tau_p}{\eta} (\varphi_2 - \psi_2 + k\pi) \quad k = 0, 1, \dots \quad (18)$$

$$\text{Where } \varphi_1 = \varphi_2 = \tan^{-1} \left(\frac{\tau_0 \xi + (\tau_p + \tau_0 \xi) \eta}{-\tau_0 \eta + (\tau_p + \tau_0 \xi) \xi} \right).$$

After reading t_{P+}^* and t_{P-}^* as the time to reach the peak of the limit cycle in steady oscillations as shown in fig. 2 we have (19), [9]. If there are several peaks the time to reach the largest one should be considered.

$$\begin{aligned} t_{P+} - t_{P-} &= \frac{\tau_p}{\eta} (\psi_2 - \psi_1) = P_+ - \theta + t_{P+}^* \\ &\quad - (P_- - \theta + t_{P-}^*) = P_+ - P_- + t_{P+}^* - t_{P-}^* \quad (19) \end{aligned}$$

If we use the point $G(\alpha+j0)=A_\alpha$ which is identified during the process of computing the gain, and substitute in (2) we can have (20). By computing k_p from (12) and substituting it into (13) τ_θ can be derived explicitly with respect to τ_p and ζ as is showed in (21).

$$\frac{-\tau_0 \alpha + 1}{\tau_p^2 \alpha^2 + 2\xi \tau_p \alpha + 1} e^{-\theta \alpha} = A_\alpha \quad (20)$$

$$\tau_0 = \frac{1}{\omega_u} \frac{A_u}{K_p} \sqrt{\left((1 - \tau_p^2 \omega_u^2)^2 + 4\xi^2 \tau_p^2 \omega_u^2 \right) - 1} \quad (21)$$

By substituting (14) and (21) into (20) and solving the resultant equation with (19) simultaneously τ_p and ζ can be computed. By substituting τ_p , ζ and k_p into (21) τ_0 can be derived, after all θ can be computed by (14).

In order to solve the nonlinear system of equations of two variables, numerical methods such as Newton-Raphson can be useful, initial conditions for τ_p and ζ can be roughly estimated using the relay response shapes of TABLE B.I. in the appendix

Finally (22) is computed as a measure of precision [11], where $\hat{y}(kT_s)$ is the output of the identified model, $y(kT_s)$ is the output of the system, $N=P_u/T_s$ and ε is a predefined precision, in order to change initial conditions if needed. The identification algorithm can be summarized as in Fig.3.

$$J = \sum_{k=1}^N |\hat{y}(kT_s) - y(kT_s)|^2 / N < \varepsilon \quad (22)$$

V. MODEL VALIDATION

For validating the identified model IAE index can be used. Where $\hat{G}(j\omega)$ is identified model and $G(j\omega)$ the real system. This index calculates multiplicative error in frequency domain in the frequency range of $[0, \omega_u]$ regarding that the ultimate frequency has been read from the limit cycle.

$$IAE_\omega = \int_0^{\omega_u} \left| \frac{\hat{G}(j\omega) - G(j\omega)}{G(j\omega)} \right| d\omega \quad (23)$$

VI. EXAMPLES

To show effectiveness of the proposed identification algorithm two examples are given.

Example 1: Consider system (24).

$$G(s) = \frac{(-0.2s+1)e^{-7s}}{s^2+0.6s+1} \quad (24)$$

by running a biased relay test with $u_+=1.3$, $u_-=-0.7$, and $\varepsilon_+=-\varepsilon_-=0.2$ a limit cycle forms as Fig. 4, if $\alpha=0.01$ and $M_0=4$

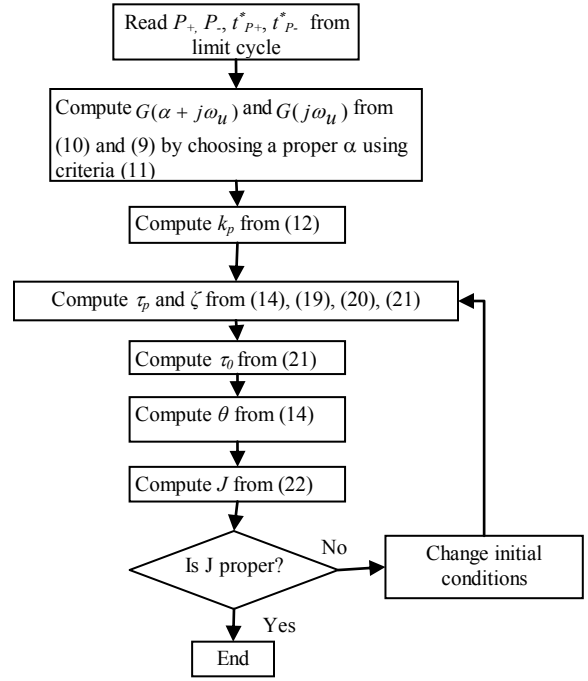


Fig. 3 Flowchart of the proposed algorithm

estimated parameters by the proposed algorithm are $k_p=1.0053$, $\tau_p=1.0026$, $\xi=0.2977$, $\tau_0=0.1556$ and $\theta=7.0472$ also IAE is 1.7241×10^{-5} . Results of the experiment are in TABLE I.

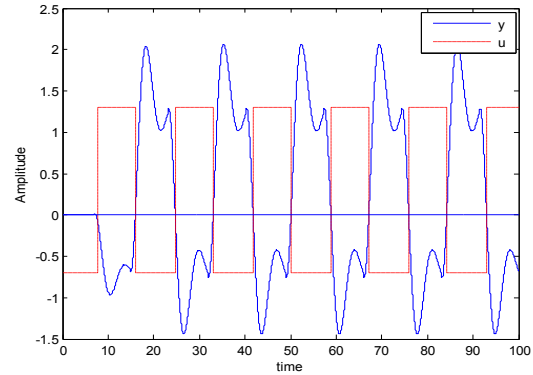


Fig. 4 Limit cycle of example 1.

Example 2: Consider the transfer function (25).

$$G(s) = \frac{1.5(-2s+1)e^{-2s}}{0.01s^2+0.14s+1} \quad (25)$$

under the relay test with $u_+=1.3$, $u_-=-0.7$ and $\varepsilon_+=-\varepsilon_-=0.1$ with limit cycle shown in Fig. 5 estimated parameters through the proposed algorithm with $\alpha=0.01$ and $M_0=4$, are $k_p=1.4997$, $\tau_p=0.1037$, $\xi=0.6461$, $\tau_0=1.9837$ and $\theta=1.0058$ also IAE is 6.052×10^{-4} . Results are in TABLE I.

TABLE II
Measured parameters of the limit cycle and estimated models for examples 1 and 2

system	P_+	P_-	$t_{P_+}^*$	$t_{P_-}^*$	φ_u	A_u	A_a	Estimated model	IAE
$G(s) = \frac{(-0.2s+1)e^{-7s}}{s^2+0.6s+1}$	8.3099	8.7551	2.2377	1.814	-2.9012	1.1238	0.9248	$G(s) = \frac{1.0053(-0.1556s+1)e^{-7.0472s}}{1.0052s^2+0.5971s+1}$	1.7241×10^{-5}
$G(s) = \frac{(-2s+1)e^{-2s}}{10s^2+11s+1}$	1.389	1.41	1.11	1.108	-3.916	6.8976	1.4535	$G(s) = \frac{1.4997(-1.9837s+1)e^{-1.0058s}}{0.0107s^2+0.134s+1}$	6.052×10^{-4}

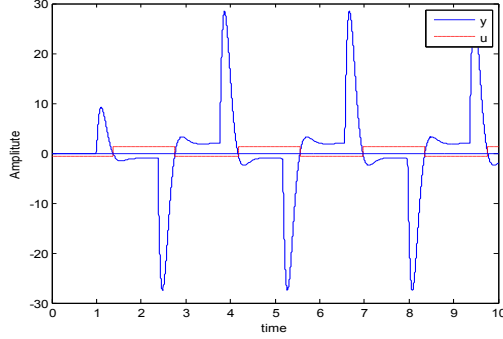


Fig. 5 Limit cycle of example 2.

VII. CONCLUSION

In this paper a new algorithm for estimation of five parameters of a SOPDT with a NMP zero is proposed. This is done through a single run of biased relay test which needs a short time to be performed and analyzed. Analytical expressions for the limit cycle formed under the relay test are derived exactly, these expressions and frequency domain estimations help to find the parameters. The algorithm prepares a simple way for modeling low or high order systems with inverse responses and under-damped behaviors. Solving the equations is done using numerical methods whose initial conditions can be chosen roughly from a table of the relay shapes of the transfer function with different parameters. A future work can be a try to eliminate the need to solve nonlinear equations and choosing initial conditions, by proposing some linear ones also extending the method for such systems with over-damped responses.

APPENDIX

A. Table of shapes for choosing initial conditions

For deriving the shapes, the ranges of the parameters for τ_p , ζ and τ_0 is selected such that it can cover reasonable possible forms for the responses in order to be used for the selection of the parameter for initial conditions.

B. Proof of the equations (3) and (4)

Relay feedback test can be considered as series of step tests [9] so the response to the first step of the relay output, $\Delta\mu - \mu_0$ is (B.1).

After the first switch of the relay, its output reaches to

$\Delta\mu + \mu_0$ and a step change of $2\mu_0$ is added to process input, the output of the process becomes as (B.2) by using linear superposition. Similarly for the second switch when the relay output changes to $\Delta\mu - \mu_0$, process output to a $-2\mu_0$ step change becomes as (B.3). By a time shift of $t_0 + \theta$ (B.2) is written as (B.4), also (B.3) changes to (B.5).

$$y_0(t) = K_p(\Delta\mu - \mu_0) \left(1 - \frac{\tau_p + \tau_0\zeta}{\eta\tau_p} e^{-\frac{\zeta(t-\theta)}{\tau_p}} \times \sin\left(\frac{\eta(t-\theta)}{\tau_p} + \varphi\right) + \frac{\tau_0}{\tau_p} e^{-\frac{\zeta(t-\theta)}{\tau_p}} \cos\left(\frac{\eta(t-\theta)}{\tau_p} + \varphi\right)\right) \quad (B.1)$$

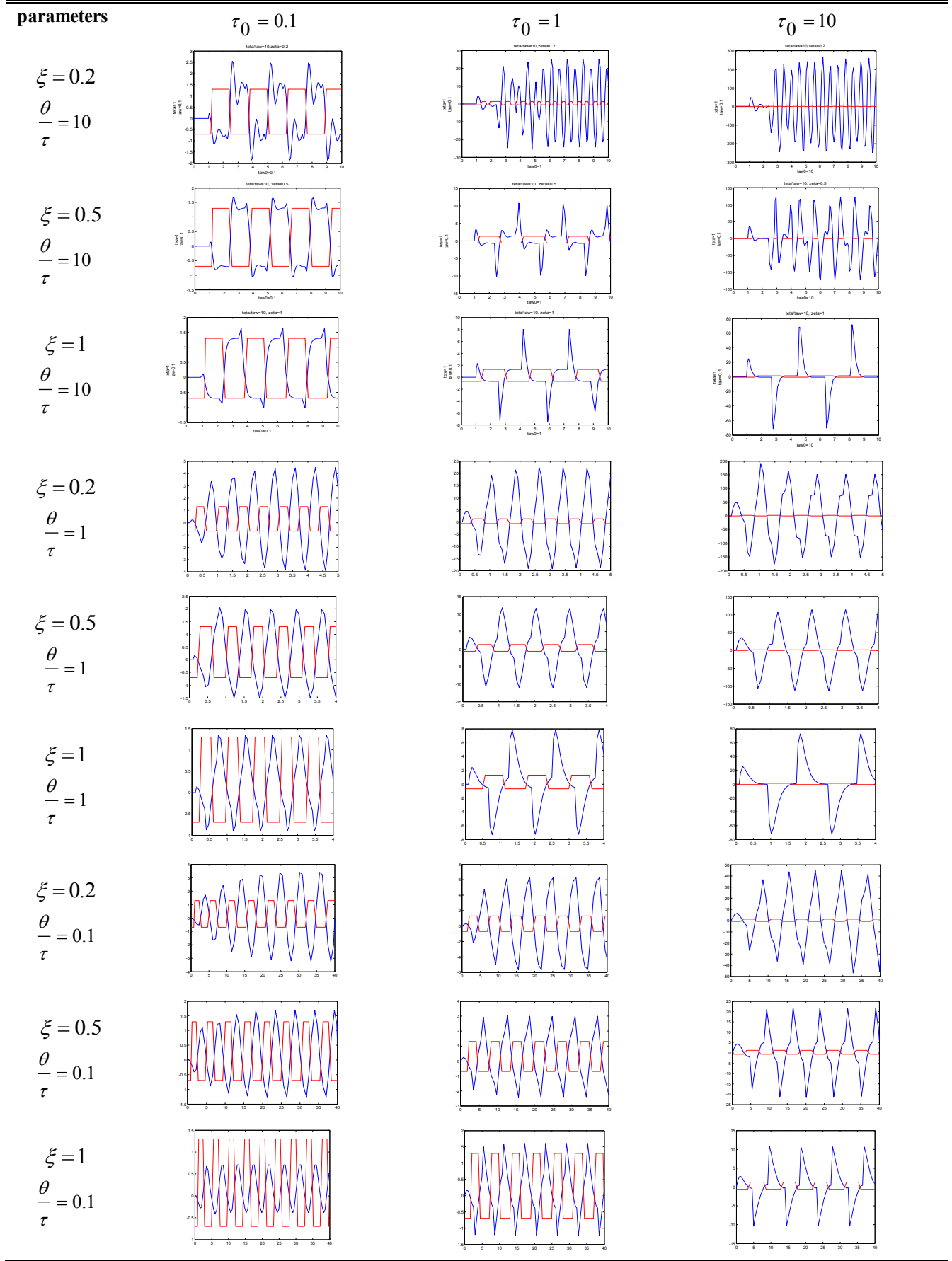
$$y_1(t) = y_0(t + t_0) + 2k_p\mu_0 \left(1 - \frac{\tau_p + \tau_0\zeta}{\eta\tau_p} e^{-\frac{\zeta(t-\theta)}{\tau_p}} \times \sin\left(\frac{\eta(t-\theta)}{\tau_p} + \varphi\right) + \frac{\tau_0}{\tau_p} e^{-\frac{\zeta(t-\theta)}{\tau_p}} \cos\left(\frac{\eta(t-\theta)}{\tau_p} + \varphi\right)\right) \quad (B.2)$$

$$y_2(t) = y_0(t + P_+) - 2k_p\mu_0 \left(1 - \frac{\tau_p + \tau_0\zeta}{\eta\tau_p} e^{-\frac{\zeta(t-\theta)}{\tau_p}} \times \sin\left(\frac{\eta(t-\theta)}{\tau_p} + \varphi\right) + \frac{\tau_0}{\tau_p} e^{-\frac{\zeta(t-\theta)}{\tau_p}} \cos\left(\frac{\eta(t-\theta)}{\tau_p} + \varphi\right)\right) \quad (B.3)$$

$$y_1|_{shift}(t) = y_0(t + t_0 + \theta) \left(1 - \frac{\tau_p + \tau_0\zeta}{\eta\tau_p} e^{-\frac{\zeta(t-\theta)}{\tau_p}} \times \sin\left(\frac{\eta(t-\theta)}{\tau_p} + \varphi\right) + \frac{\tau_0}{\tau_p} e^{-\frac{\zeta(t-\theta)}{\tau_p}} \cos\left(\frac{\eta(t-\theta)}{\tau_p} + \varphi\right)\right) \quad (B.4)$$

$$y_2|_{shift}(t) = y_0(t + t_0 + \theta + P_+) - \frac{2K_p\mu_0}{\eta} e^{-\frac{\zeta}{\tau_p}t} \times \left(e^{-\frac{\zeta}{\tau_p}P_+} \left\{ \frac{\tau_p + \tau_0\zeta}{\tau_p} \sin\left(\frac{\eta}{\tau_p}(t + P_+) + \varphi\right) - \frac{\tau_0\eta}{\tau_p} \cos\left(\frac{\eta}{\tau_p}(t + P_+) + \varphi\right) \right\} - \frac{\tau_p + \tau_0\zeta}{\tau_p} \sin\left(\frac{\eta}{\tau_p}t + \varphi\right) + \frac{\tau_0\eta}{\tau_p} \cos\left(\frac{\eta}{\tau_p}t + \varphi\right)\right) \quad (B.5)$$

TABLE A.1
RELAY FEEDBACK RESPONSES OF SOPDT WITH ZERO WITH DIFFERENT PARAMETERS



The procedure of finding step responses continues, and finally the process output corresponding to the $(2n+1)^{\text{th}}$ and $(2n+2)^{\text{th}}$ switching namely y_{2n+1} and y_{2n+2} become as (B.6) and (B.7). Where $n=0,1,2,\dots$

$$y_{2n+1} |_{\text{shift}}(t) = y_0(t + t_0 + \theta + nP_u) - \frac{2K_p \mu_0}{\eta} e^{-\frac{\xi}{\tau_p} t} \left(\frac{\tau_p + \tau_0 \xi}{\tau_p} E_1 - \frac{\tau_0}{\tau_p} E_2 \right) + 2K_p \mu_0 \quad (\text{B.6})$$

$$y_{2n+2} |_{\text{shift}}(t) = y_0(t + t_0 + \theta + nP_u + P_+) - \frac{2K_p \mu_0}{\eta} e^{-\frac{\xi}{\tau_p} t} \left(\frac{\tau_p + \tau_0 \xi}{\tau_p} F_1 - \frac{\tau_0}{\tau_p} F_2 \right) \quad (\text{B.7})$$

We define (B.8) to (B.11) in order to simplify presentation.

$$E_1 = \sin\left(\frac{\eta}{\tau_p} t + \varphi\right) + \sum_{k=1}^n e^{-\frac{kP_u \xi}{\tau_p}} \sin\left(\frac{\eta}{\tau_p} (t + kP_u) + \varphi\right) - \frac{[(k-1)P_u + P_-] \xi}{\tau_p} \sin\left(\frac{\eta}{\tau_p} (t + (k-1)P_u + P_-) + \varphi\right) \quad (\text{B.8})$$

$$E_2 = \cos\left(\frac{\eta}{\tau_p} t + \varphi\right) + \sum_{k=1}^n e^{-\frac{kP_u \xi}{\tau_p}} \cos\left(\frac{\eta}{\tau_p} (t + kP_u) + \varphi\right) - \frac{[(k-1)P_u + P_-] \xi}{\tau_p} \cos\left(\frac{\eta}{\tau_p} (t + (k-1)P_u + P_-) + \varphi\right) \quad (\text{B.9})$$

$$F_1 = \sum_{k=0}^n \left(e^{-\frac{(kP_u + P_+)}{\tau_p}} \sin\left(\frac{\eta}{\tau_p} (t + kP_u + P_+) + \varphi\right) - e^{-\frac{kP_u \xi}{\tau_p}} \times \sin\left(\frac{\eta}{\tau_p} (t + kP_u) + \varphi\right) \right) \quad (\text{B.10})$$

$$F_1 = \sum_{k=0}^n \left(e^{-\frac{(kP_u + P_+)}{\tau_p}} \cos\left(\frac{\eta}{\tau_p} (t + kP_u + P_+) + \varphi\right) - e^{-\frac{kP_u \xi}{\tau_p}} \times \cos\left(\frac{\eta}{\tau_p} (t + kP_u) + \varphi\right) \right) \quad (\text{B.11})$$

Regarding $0 < e^{-P_u / \tau_p} < 1$ when $n \rightarrow \infty$ and Euelr's equalities, we have (B.12), [11].

$$\sum_{k=0}^{\infty} e^{-\frac{kP_u \xi}{\tau_p}} = \frac{1}{1 - e^{-\frac{P_u \xi}{\tau_p}}} \quad (\text{B.12})$$

also we can write (B.13) to (B.16) for parts of the (B.8) - (B.11). By these simplifications (B.13) to (B.16) can be rewritten in the form of the (B.17) to (B.21).

$$\sum_{k=1}^n e^{-\frac{kP_u \xi}{\tau_p}} \sin\left(\frac{\eta(t + kP_u)}{\tau_p} + \varphi\right) = -\sin\left(\frac{\eta t}{\tau_p} + \varphi\right) + \frac{\sin\left(\frac{\eta t}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \sin\left(\frac{\eta(t - P_u)}{\tau_p} + \varphi\right)}{1 - 2e^{-\frac{P_u \xi}{\tau_p}} \cos\frac{\eta P_u}{\tau_p} + e^{-\frac{2P_u \xi}{\tau_p}}} \quad (\text{B.13})$$

$$\sum_{k=1}^n e^{-\frac{(kP_u + P_-) \xi}{\tau_p}} \sin\left(\frac{\eta(t + kP_u + P_-)}{\tau_p} + \varphi\right) = \frac{e^{-\frac{P_- \xi}{\tau_p}} \left[\sin\left(\frac{\eta(t + P_-)}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \sin\left(\frac{\eta(t - P_+)}{\tau_p} + \varphi\right) \right]}{1 - 2e^{-\frac{P_u \xi}{\tau_p}} \cos\frac{\eta P_u}{\tau_p} + e^{-\frac{2P_u \xi}{\tau_p}}} \quad (\text{B.14})$$

$$\sum_{k=1}^n e^{-\frac{kP_u \xi}{\tau_p}} \cos\left(\frac{\eta(t + kP_u)}{\tau_p} + \varphi\right) = -\cos\left(\frac{\eta t}{\tau_p} + \varphi\right) + \frac{\cos\left(\frac{\eta t}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta(t - P_u)}{\tau_p} + \varphi\right)}{1 - 2e^{-\frac{P_u \xi}{\tau_p}} \cos\frac{\eta P_u}{\tau_p} + e^{-\frac{2P_u \xi}{\tau_p}}} \quad (\text{B.15})$$

$$\sum_{k=1}^n e^{-\frac{(kP_u + P_-) \xi}{\tau_p}} \cos\left(\frac{\eta(t + kP_u + P_-)}{\tau_p} + \varphi\right) = \frac{e^{-\frac{P_- \xi}{\tau_p}} \left[\cos\left(\frac{\eta(t + P_-)}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta(t - P_+)}{\tau_p} + \varphi\right) \right]}{1 - 2e^{-\frac{P_u \xi}{\tau_p}} \cos\frac{\eta P_u}{\tau_p} + e^{-\frac{2P_u \xi}{\tau_p}}} \quad (\text{B.16})$$

$$\begin{aligned}
& \sin\left(\frac{\eta t}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \sin\left(\frac{\eta(t - P_u)}{\tau_p} + \varphi\right) - \\
& e^{-\frac{P_- \xi}{\tau_p}} \left[\sin\left(\frac{\eta(t + P_-)}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \sin\left(\frac{\eta(t - P_+)}{\tau_p} + \varphi\right) \right] \\
E_1 = & \frac{\tau_p}{1 - 2e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta P_u}{\tau_p} + e^{-\frac{2P_u \xi}{\tau_p}}\right)} \\
& = \frac{\rho_1 \sin\left(\frac{\eta t}{\tau_p} + \psi_1\right)}{\gamma}
\end{aligned} \quad (B.17)$$

$$\begin{aligned}
& \cos\left(\frac{\eta t}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta(t - P_u)}{\tau_p} + \varphi\right) - \\
& e^{-\frac{P_- \xi}{\tau_p}} \left[\cos\left(\frac{\eta(t + P_-)}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta(t - P_+)}{\tau_p} + \varphi\right) \right] \\
E_2 = & \frac{\tau_p}{1 - 2e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta P_u}{\tau_p} + e^{-\frac{2P_u \xi}{\tau_p}}\right)} \\
& = \frac{\rho_1 \cos\left(\frac{\eta t}{\tau_p} + \psi_1\right)}{\gamma}
\end{aligned} \quad (B.18)$$

$$\begin{aligned}
& \sin\left(\frac{\eta t}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \sin\left(\frac{\eta(t - P_u)}{\tau_p} + \varphi\right) - \\
& e^{-\frac{P_+ \xi}{\tau_p}} \left[\sin\left(\frac{\eta(t + P_+)}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \sin\left(\frac{\eta(t - P_-)}{\tau_p} + \varphi\right) \right] \\
F_1 = & \frac{\tau_p}{1 - 2e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta P_u}{\tau_p} + e^{-\frac{2P_u \xi}{\tau_p}}\right)} \\
& = -\frac{\rho_2 \sin\left(\frac{\eta t}{\tau_p} + \psi_2\right)}{\gamma}
\end{aligned} \quad (B.19)$$

$$\begin{aligned}
& \cos\left(\frac{\eta t}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta(t - P_u)}{\tau_p} + \varphi\right) - \\
& e^{-\frac{P_+ \xi}{\tau_p}} \left[\cos\left(\frac{\eta(t + P_+)}{\tau_p} + \varphi\right) - e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta(t - P_-)}{\tau_p} + \varphi\right) \right] \\
F_2 = & \frac{\tau_p}{1 - 2e^{-\frac{P_u \xi}{\tau_p}} \cos\left(\frac{\eta P_u}{\tau_p} + e^{-\frac{2P_u \xi}{\tau_p}}\right)} \\
& = -\frac{\rho_2 \cos\left(\frac{\eta t}{\tau_p} + \psi_2\right)}{\gamma}
\end{aligned} \quad (B.20)$$

All the parameters are defined in the text and so parts of the relay response of an under-damped SOPDT with a NMP zero is as (B.21) and (B.22).

$$\begin{aligned}
y_+(t) = \lim_{n \rightarrow \infty} y_{2n+1} |_{shift} (t) = & K_P (\Delta\mu + \mu_0) - \\
& \frac{2K_P \mu_0 \rho_1}{\eta\gamma} e^{-\frac{\xi}{\tau_p} t} \left\{ \left(\frac{\tau_p + \tau_0 \xi}{\tau_p} \right) \times \sin\left(\frac{\eta}{\tau_p} t + \psi_1\right) - \right. \\
& \left. \left(\frac{\tau_0}{\tau_p} \right) \cos\left(\frac{\eta}{\tau_p} t + \psi_1\right) \right\} \quad t \in [0, P_+]
\end{aligned} \quad (B.21)$$

$$\begin{aligned}
y_-(t) = \lim_{n \rightarrow \infty} y_{2n+2} |_{shift} (t) = & K_P (\Delta\mu - \mu_0) + \\
& \frac{2K_P \mu_0 \rho_2}{\eta\gamma} e^{-\frac{\xi}{\tau_p} t} \left\{ \left(\frac{\tau_p + \tau_0 \xi}{\tau_p} \right) \times \sin\left(\frac{\eta}{\tau_p} t + \psi_2\right) - \right. \\
& \left. \left(\frac{\tau_0}{\tau_p} \right) \cos\left(\frac{\eta}{\tau_p} t + \psi_2\right) \right\} \quad t \in (P_+, P_u]
\end{aligned} \quad (B.22)$$

REFERENCES

- [1] K. J. Astrom and T. Hagglund, "Automatic tuning of simole regulators with specifications on phase and amplitude margins," *Automatica. J.*, vol. 20, no. 5, pp. 645-651, Sep. 1984.
- [2] C. C. Yu, "Autotuning of PID controller: a relay feedback approach," 2nd ed. Springer, 2006, pp. 47-144.
- [3] Q. G. Wang, C. C. Hang and B. Zou, "Low-order modeling from relay feedback," *Ind. Eng. Chem. Res. J.*, vol. 36, no. 4, pp. 375-381, Feb. 1997.
- [4] S. Majhi, D. P. Atherton, "Autotuning and controller design for processes with small time delays," in *Proc. IEE Control Theory App.*, vol. 146, no.5, pp. 421-427 Sep. 1999.
- [5] S. Majhi, "Relay based identification of processes with time delay," *Process Control. J.*, vol. 17, no. 2, pp. 93-101, Feb. 2007.
- [6] R. C. Panda and C. C. Yu, "Analytical expressions for relay feedback responses," *Process Control. J.*, vol. 13, no. 6, pp. 489-501, Sep. 2003.
- [7] S. Vivek and M. Chidambaram, "Identification using single symmetrical relay feedback test," *Computers and Chemical Engineering*, vol. 29, no. 7, pp. 1625-1630, 2005.
- [8] S. H. Shen, J. S. Wu and C. C. Yu, "Use of biased-relay feedback for system identification," *AIChE J.*, vol. 42, no. 4, pp. 1174-1180, Apr. 1996.
- [9] T. Liu and F. Gao, "A generalized relay identification method for time delay and non-minimum phase processes," *Automatica. J.*, vol. 45, no. 4, pp. 1072-1079, Apr. 2009.
- [10] T. Liu and F. Gao, "Alternative identification algorithm for obtaining first order stable/unstable process model from a single relay feedback test," *Ind. Eng. Chem. Res. J.*, vol. 47, no. 4, pp. 1140-1149, Jan 2008.
- [11] T. Liu, F. Gao and Y. N. Wang, "A systematic approach for on-line identification of second order process model from relay feedback test," *AIChE J.*, vol. 54, no. 6, pp. 1560-1578, Apr. 2008.
- [12] K. Soltesz, T. Hagglund and K. J. Astrom, "Transfer function parameters identification by modified relay feedback," in *Proc. Ameriacn Control Conf.*, Baltimore, 2010, pp. 2164-2169.

Multiobjective Graph Genetic Programming with Encapsulation Applied to Neural System Identification

Lavinia Ferariu and Bogdan Burlacu

Abstract—This paper presents two new encapsulation operators compatible with graph genetic programming. The approach is used for the evolution of partially interconnected, feed-forward hybrid neural networks, within the framework of nonlinear system identification. The suggested encapsulations are targeted to protect valuable terminals and useful sub-graphs directly connected with the root node. To preserve a better balance between exploitation and exploration, the quality of the inner substructures is assessed in relation with the phenotypic properties of the individuals to whom they belong. The multiobjective optimization of accuracy and parsimony is adopted; for each generation, the requirements expressed by the decision block are progressively translated to the evolutionary algorithm, via a preliminary clustering of the individuals, performed before Pareto-ranking. The experimental results achieved on the identification of an industrial plant indicate that the proposed encapsulations are able to enforce the selection of accurate and simple models.

I. INTRODUCTION

THE quality of the model describing the dynamic behavior of the plant is a key issue to automatic control applications [1]. Usually, the model is obtained via data driven approaches, whilst facing system nonlinearities, noisy working environments and partial/missing information about the model structure and/or the minimal set of input variables. Although the design procedure exploits a finite set of representative training samples, the validity of the model must be extrapolated to as large as possible plant operation ranges. Frequently, the model is forced to retain the most relevant characteristics through over-fitting avoidance techniques. This inherently leads to another valuable outcome, namely an easier and faster management of the model during any future stage of control system design and exploitation.

In relation to these ideas, one of the most efficient alternatives consists in multiobjective collaborative neuro-evolutionary systems [1]. The strength of this symbiosis results from the characteristics of all involved components. Firstly, the neural networks can bring universal

approximation capabilities in terms of continuous bounded functions, as well as robustness and good generalization ability via the inductive learning of parallel, highly interconnected architectures [2], [3]. In addition, the multiobjective optimization carried out in terms of model accuracy and model parsimony represents a typical anti-over-fitting technique [1], [4], [5], [6]. Finally, the selection of Pareto-optimal solutions (accurate and simple) could be done by means of evolutionary algorithms which provide universality, flexibility and robustness whilst exploring in an unsupervised manner within large, multidimensional spaces [1], [4].

This paper employs graph genetic programming (GP) which is able to act simultaneously on the structure and the parameters of the model (including also variable reduction) [7]. This means the canonical GP [8] is tailored to manage a population of directed acyclic graphs (DAG) (not tree-based individuals), by making use of an enhanced initialization procedure and customized genetic operators [9]. The main benefit is a natural and compact direct encoding of the neural models. However, offspring production is confronted with increased difficulties whenever certain sub-graphs need to be removed or replaced within a hierarchical individual.

The success of an evolutionary algorithm is strongly dependent on the balance maintained between the capacity of exploiting available valuable genetic material and exploring through new search areas (by means of numerous and diverse offspring) [7]. This balance is tuned by selection and genetic operators, which usually act sequentially, yet in different spaces. The main challenge here is that the algorithm interprets the quality of an individual via the comparison with the other competitor solutions chosen in the current population, while, in nonlinear optimizations, there is no guarantee that the fitted individuals are also the ones placed closest to the Pareto-optimal solutions within the genotypic space. Additionally, the offspring could result less adapted than their parents, especially when confronted with competing conventions and bloat, as in the case of GP-based neuro-evolution [7].

Usually, the structural crossover makes no difference between the phenotypic usefulness of the inner structures belonging to the parents, so it randomly selects the blocks to be exchanged [8]. To prevent structural redundancy, similitude-based crossovers avoid the export of substructures which are alike with other blocks already

Manuscript received April 30, 2010. This work was supported by The National Centre for Programs Management from Romania under the research grant SICONA – 12100/2008.

L. Ferariu and B. Burlacu are with the Department of Automatic Control and Applied Informatics, “Gheorghe Asachi” Technical University of Iasi, Romania, Bd. D. Mangeron 27, IS 700 050 RO (e-mail: lferaru@ac.tuiasi.ro, bburlacu@ac.tuiasi.ro).

contained by the host solutions [7]. However, in the case of graph encoding, the genotypic similitude is hardly detected and involves high computational costs.

According to earlier suggestions of Koza [8], exploitation could be better bridged with exploration when the crossover is mixed with encapsulation. Basically, the encapsulation protects some inner substructures during recombination, such that they are passed unaltered to the offspring. These blocks could be stochastically chosen from the current population, as suggested by Dessi, or could be discovered inside the best adapted individuals [10]. In the later case, the encapsulation acts beyond the microscopic level, as it has to analyze the phenotypic significance of the substructures available in the recombination pool. In addition, the protected blocks could be inserted within the set of primitives, in order to enforce their reuse. In this case, different protection strategies can be employed for the nodes corresponding to the initial and newly introduced primitives, or for the nodes which were originally protected by encapsulation [10]. However, all the former approaches are targeted to simpler hierarchical architectures only, namely the tree-based ones.

Obviously, productive encapsulation increases the risk of premature convergence due to the potential loss of diversity. It should be noticed that this behavior is very dangerous, especially when the multiobjective optimization is solved by means of Pareto-ranking techniques [5], [11]. In order to ensure a better control on the macroscopic effects induced by the microscopic alterations of the genetic material, the probability of encapsulation could be tuned via global strategies – e.g. by using a fuzzy controller, as suggested in [12].

This paper introduces two type of encapsulations tailored for graph-based individuals. Each individual encrypts a **multiple input single output (MISO)**, partially interconnected feed-forward **hybrid neural network (HNN)** [9], [13]. It should be noticed that these neural structures accept any connectivity map between successive neural layers, different types of parameters associated to the neural links and any combination of global and local neurons within each hidden layer. One of the suggested encapsulations is sensitivity-based; it protects certain blocks inside the parents after analyzing the influence of the sub-graphs directly connected with the output nodes. On the other hand, the terminal encapsulation keeps the terminals which are more useful than average, yet tend to be eliminated via the inherent variable reduction performed during offspring production. The approach employs Pareto-ranking techniques with preliminary clustering, in order to ensure a progressive articulation between decision and search.

The paper is organized as follows. Section II presents the main characteristics of HNNs and summarizes the

techniques used for the multiobjective optimization of the neural models, given the framework of customized graph GP. The novel encapsulation operators are described in Section III, while Section IV is devoted to experimental analysis. Main conclusions are stated in Section V.

II. ALGORITHM OVERVIEW

A. The Hybrid Neural Model

The adopted neural model is MISO HNN [13]. It accepts feed-forward partial interconnectivity, as well as any combination of global and local neurons within the hidden layers, thus leading to increased approximation capabilities [13]. Having in mind a potential wide output range, the output neuron is explicitly set to be global. Without altering the universality of the approach, this exemplification manages a combination of global perceptrons with or without functional links [2] and local Gaussian neurons with real or complex weights [3]. If $\mathbf{z} = [z_i]_{i=1, \dots, no-i}$ denotes the neural inputs of a neuron, the input-output mappings of the former neural units can be written as follows:

$$f_{PS}(\mathbf{z}, \boldsymbol{\theta}_{PS}) = \tanh\left(\sum_{i=1}^{no-i} w_i z_i + b\right), \quad (1)$$

for the standard perceptron with weights w_i and bias b ($\boldsymbol{\theta}_{PS} = [w_1 \dots w_{no-1} b] \in \mathfrak{R}^{no-i+1}$);

$$f_{PF}(\mathbf{z}, \boldsymbol{\theta}_{PF}) = \tanh\left(\sum_{i=1}^{no-i} \left(\sum_{j=1}^P w_{ij}^c (\cos \mathcal{M}_j z_i) + \sum_{j=1}^P w_{ij}^s (\sin \mathcal{M}_j z_i) + w_i z_i + b\right)\right) \quad (2)$$

for the global neuron with orthogonal trigonometric functional links of maximum order P ($\boldsymbol{\theta}_{PF} \in \mathfrak{R}^{1+no-i \cdot (2P+1)}$) [14];

$$f_{GR}(\mathbf{z}, \boldsymbol{\theta}_{GR}) = e^{-\frac{1}{2\sigma^2} \sum_{i=1}^{no-i} (c_i - z_i)^2}, \quad (3)$$

for the Gaussian neuron with centers c_i and spread σ ($\boldsymbol{\theta}_{GR} = [c_1, \dots, c_{no-i}, \sigma] \in \mathfrak{R}^{no-i+1}$);

$$f_{GC}(\mathbf{z}, \boldsymbol{\theta}_{GC}) = e^{-w^2 \left[\left(\sum_{i=1}^{no-i} \cos \alpha_i (z_i - c_i) \right)^2 + \left(\sum_{i=1}^{no-i} \sin \alpha_i (z_i - c_i) \right)^2 \right]} \quad (4)$$

for the local neuron with centers c_i and complex weights $w e^{\sqrt{-1} \cdot \alpha_i}$ ($\theta_{GC} = [\alpha_1, \dots, \alpha_{no_i}, c_1, \dots, c_{no_i}, w]' \in \mathbb{R}^{2no_i+1}$) [3]. An individual processed by graph GP encodes a potential neural model built in compliance with input-output series-parallel schemes. Consequently, the neural input variables have to be chosen from the set

$$\mathbf{T} = [\mathbf{u}(k), \dots, \mathbf{u}(k - n_u), \mathbf{y}(k - 1), \dots, \mathbf{y}(k - n_y)], \quad (5)$$

where $\mathbf{u} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^n$ denote the inputs and the outputs of the system, k indicates the current sampling instant and n_u, n_y represent the maximum permitted input and output lags, respectively. $\mathbf{T}$ permits to illustrate potential interdependencies between system outputs, so, when dealing with multiple inputs multiple outputs systems, each output could be approximated by a distinct model.

B. Graph Encoding and Enhanced Genetic Operators

Considering the former, GP works on DAGs having various sizes and shapes, produced with a sufficient set of terminals $\mathbf{T}$ and the minimally sufficient set of operators, $\mathbf{O} = \{f_{PS}, f_{PF}, f_{GR}, f_{GC}\}$ [9]. A node of the graph denotes a primitive (the leaves correspond to the terminals) and a link of the graph indicates a neural connection (Fig. 1).

For an adequate management of graph-based individuals, GP needs a customized production of initial chromosomes and enhanced genetic operators. Within this context, the initial DAGs are created from canonical GP trees by adding extra-connections between the nodes belonging to successive layers [9]. The distinction between normal and extra links is used also by the genetic operators: whenever an operator enables a structure modification, the correction of links' types is performed for all involved nodes, such that each node should have a single normal link towards its parents and any number of extra-connections [9]. This trick allows a simpler management of pointers and data copying whilst dealing with nodes of variable arity and different number/type of parameters. It should be also noticed that unlike other graph GP-based methods [14], this approach employs genetic operators able to work on the inner structure of the graph nodes [9] and to accept various types and number of neural parameters for each neural unit/link, whilst guaranteeing the offspring validity. The customizations are targeted to allow both ample and small alterations of the encrypted neural structures and parameters. Hence, the structural crossover switches two randomly chosen sub-graphs between the parents; the structural mutation substitutes a sub-graph with a stochastically newly generated one; the node mutation changes the type of randomly selected nodes; the link mutation adds extra-connections from the function nodes to

the children of other neurons belonging to the same level the parametric mutation alters the parameters of a selected node, without changing its type or its incoming connectivity map [9].

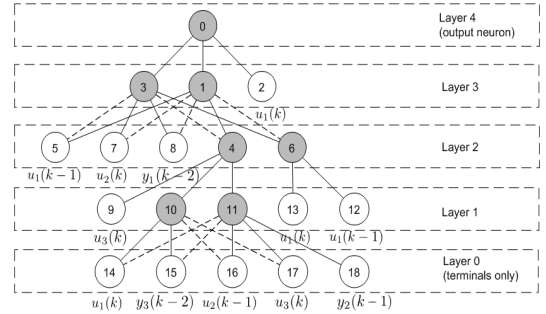


Fig. 1. DAG-based encoding of HNNs. The colored nodes are functional and the white ones denote the terminals.

C. Multiobjective Optimization Solved with Pareto-Techniques

The DAG-based individuals are evolved via a multiobjective optimization (MOO). Assuming a representative training data set, $\mathbf{S} = \{(\mathbf{u}(t), \mathbf{y}(t))\}$, which describes the dynamic behavior of the system at instants $t = 1, \dots, d$, the MOO aims to minimize the empirical risk (evaluated in terms of $\mathbf{S}$) and the complexity order of model, expressed as indicated below:

$$f_1(M) = \frac{\sum_{t=1}^d (y_i(t) - \hat{y}_i(t))^2}{d}, \quad f_2(M) = n_{nodes} + \frac{n_{links}}{MAXL} \quad (6)$$

Here, M denotes a potential model (encrypted as a graph-based individual), $y_i(t)$ and $\hat{y}_i(t)$ denote the i^{th} output of the plant at instant t and its approximation, respectively, n_{nodes} and n_{links} specifies the number of nodes and links included in M , and $MAXL$ represents the maximum arity of a graph node.

The former MOO is solved by means of Pareto-techniques (known to be very effective when dealing with few objectives). Every generation, before ranking, the population $P(gen)$ is separated in three different clusters: C_1 includes the individuals simpler and more accurate than average, C_2 has the individuals more accurate and more complex than average, and the rest of $P(t)$ belongs to C_3 . This preliminary clustering permits a customized ranking, tailored to the most common decisions involved in identification problems. More precisely, one can enforce the accuracy improvement while impeding the selection of the Pareto-optimal solutions which have no practical utility (namely the models very simple but too imprecise, or the

models accurate but too complex).

With this in mind, the Pareto-ranks are separately determined within each resulted group, keeping the selection probabilities from C_1 higher than those used in C_2 , and the selection probabilities from C_2 higher than those used in C_3 . The Pareto-fronts are depicted in sequence, as suggested by Deb: the first-order front includes the non-dominated solutions of the set; then, these solutions are excluded and the second-order front is filled with the non-dominated solutions of the remaining set, and so on [4]. The linearly assigned fitness values are used by stochastic universal sampling (for filling the recombination pool) and by insertion (for replacing the worst parents with fittest offspring). The most accurate model of the last population is picked as the final result of the algorithm.

D. Algorithm Outline

Summing up, the algorithm can be briefly portrayed as indicated below:

- 1) Create a stochastic initial population of trees (each one using all the elements of $\mathbf{T}$). Add random extra connections to form DAGs; $gen = 0$.
- 2) While ($gen < MAXGEN$)
 - i) Evaluate the population in terms of f_1 and f_2 .
 - ii) Compute the fitness values (depict the clusters; separate the fronts; assign the fitness values).
 - iii) Select the recombination pool.
 - iv) Apply the encapsulation. Produce the offspring (by means of crossover and mutations).
 - vi) Evaluate the offspring in terms of f_1 and f_2 .
 - vii) Compute the fitness values of the offspring.
 - viii) Insert the best 50% offspring into the population.
 - ix) $gen \rightarrow gen + 1$.
- 3) Display the result of the algorithm.

III. ENCAPSULATION OPERATORS

The success of GP is strongly dependent on the productivity and the exploration capacity of the genetic operators. Obviously, it is of great interest to use explorative crossovers and mutations able to produce a large variety of offspring. However, while the genetic operators act within the genotypic space, there is no guarantee that the resulted (various) offspring will be better adapted than their parents, especially when the genetic operators are dependent on the positions of the genes and the genetic encoding favors the epistasy [7]. In the case of HNN's encoding, epistasy is related to competing conventions [1], [13]. It occurs whenever the selected parents have distinct genetic material, yet they encrypt similar or identical HNNs (e.g. when some neurons are reordered within the hidden layers without altering the

overall functionality of the neural network).

Additionally, the unguided structural crossover can produce bloat, namely an increase of the mean complexity of the population unaccompanied by a significant improvement of the accuracy [7]. This happens especially because the substructures exchanged by the parents have different sizes, so the offspring could result of significantly different complexity orders. If the simplest children do not contain all necessary representative inner blocks, their unsuccessful lower accuracy conducts to low survival probabilities. However, the remaining offspring are not necessarily very accurate, even if they are very complex. Besides, the customized graph-based genetic operators can lead to the horizontal expansion of the individuals. This means the production of less interconnected graphs due to successive transformations of several extra links into normal ones with insertions of additional required nodes [9]. The undesired increasing of the complexity order (due to bloat or horizontal expansion) is controlled by means of MOO, which rejects the exceedingly large individuals. However, MOO eliminates the complex offspring after they have been already produced by the genetic operators. Obviously, a much more efficient alternative would be to guide crossover and the mutations while they act on the recombination pool.

Considering the former, the authors suggest two new encapsulations, tailored for graph GP. They make use of additional heuristics to disable the alteration of the most valuable inner blocks by means of structural crossover or mutation. The sensitivity-based encapsulation finds the most valuable sub-graphs of the parents, directly connected with the output (root) nodes. In Fig. 1, one can chose between the node 2 or the sub-graphs rooted by nodes 3 and 2, respectively. To fit this end, the algorithm computes the sensitivity scores of the root nodes' children:

$$s^q = |w_q| \cdot \sum_{t=1, if}^d \left| \frac{y_{hid}^q(t)}{y_i(t) - \hat{y}_i(t)} \right|_{|y_i(t) - \hat{y}_i(t)| > m_1} \quad (7)$$

and locks the sub-graphs rooted by the hidden neurons with a sensitivity score better than average. Here, $y_{hid}^q(t)$ represents the output of the q^{th} hidden neuron for the training sample t and w_q represents the weight of the link provided from the i^{th} hidden neuron to its parent.

Knowing that additional copies of helpful encapsulated blocks are inherently inserted within the population via well adapted offspring, the sensitivity-based encapsulation may lock a specific sub-graph for a reduced number (k) of consecutive generations, thus avoiding excessive protection and premature convergence. By protecting the destruction of the most relevant substructures, one can also reduce the

negative effect of competing conventions, as the structural crossover is limited to exchange sub-graphs which have less influence on the model behavior. It is also worth mentioning that the root node of an MISO graph cannot be connected with its children by means of extra links, so the sensitivity scores of the sub-graphs (directly) connected with output node can handled at low computational costs.

GP inherently performs the variable reduction via its genetic operators. Because the elimination of terminals is done in an unsupervised fashion, GP has to deal with two conflicting risks: the acceptance of unnecessary introns and the rejection of useful leaves. The MOO encourages the exclusion of unnecessary terms, via the minimization of f_2 . On the other hand, the unduly elimination of useful terminals is targeted by the suggested terminal encapsulation. Every generation, the algorithm counts the individuals to whom $t_i \in \mathbf{T}$, $i = 1, \dots, |\mathbf{T}|$ belongs and associates the mean fitness of these models as the fitness value of t_i (denoted f_{-t_i}). The valuable terminals which need to be protected are those having the highest fitness values. The empirical rules listed below are used for the best 10% terminals (having also $f_{-t_i} > \text{mean}_{i=1, \dots, |\mathbf{T}|} f_{-t_i}$):

If $z_i < 0.3 \text{mean}_{i=1, \dots, |\mathbf{T}|} z_i$ then $P_e = 0.8$.

If $z_i \in [0.3 \text{mean}_{i=1, \dots, |\mathbf{T}|} z_i, 0.7 \text{mean}_{i=1, \dots, |\mathbf{T}|} z_i]$, then $P_e = 0.5$. (8)

If $z_i > 0.7 \text{mean}_{i=1, \dots, |\mathbf{T}|} z_i$ then $P_e = 0.2$.

where P_e denotes the probability of encapsulation. Whenever a terminal should be encapsulated within a graph, a whole sub-graph should be protected, namely the one which contains the terminal and is directly connected with the root node. If the terminal has more instances in the graph, one of them, randomly selected determines the sub-graph to be encapsulated. It should be noticed that the protected sub-graph could also include undesired terminals. Hence, a refined terminal encapsulation can enforce the protection of the smallest sub-graph containing an instance of the terminal or can be applied on the simplest individuals compliant with (8). For example, according to Fig. 1, the terminal $u_1(k)$ can be protected via the node 2, or via the sub-graph rooted by 1, the latter case involving the inherent protection of all the other terminals used by the graph.

The encapsulations are enabled after several generations, when the population contains useful inherited models. Furthermore, the preliminary clustering performed by MOO can be exploited for implementing various customized encapsulation policies. For instance, the encapsulation can be solely applied to the most adapted DAGs included in C_1 , if C_1 is large enough ($|C_1| > \alpha |P(t)|$, with $\alpha < 1$).

IV. APPLICATION

All the facilities described in Sections II and III are implemented in C/C++ by the authors. The validity of the proposed design tool is verified on the identification of a nonlinear industrial plant, namely an evaporator subsystem (EV) from Lublin Sugar Factory of Poland. The plant admits three inputs (the steam flow and steam temperature at the input of evaporation station, as well as the juice temperature after heater) and one output (the juice temperature at the exit of the corresponding evaporation section). The training and the validation data sets contain about 300 measurements acquired during real-time operation, for different production shifts. They illustrate as large as possible ranges of plant variables. All missing or uncertain values have been replaced with polynomial interpolation and the noise has been filtered with a low-pass 4th order Butterworth system. To avoid ill conditioning and inconvenient aggregation of the neurons' outputs, the samples are scaled in $[-0.9, 0.9]$. It is worth mentioning that no sufficiently accurate analytical model is accessible for describing the dependencies between the plant variables.

The effects of the terminal and sensitivity-based encapsulations are separately analyzed, considering the configurations indicated in Table I.

TABLE I. CONFIGURATIONS USED FOR EXPERIMENTAL TRIALS

#	1	2	3	4	5	6
N	300	1000	300	1000	300	1000
N_{gen}	100	300	100	300	100	300
$n_u = n_v$	1	1	3	3	5	5

N_{gen} indicates the number of generations.

TABLE II. EXPERIMENTAL RESULTS

#	Without encapsulation (WE)		With SE		With TE	
	T	V	T	V	T	V
1	0.00388	0.07373	0.004639	0.06563	0.002618	0.04515
2	0.00161	0.183177	0.002337	0.02378	0.001496	0.02323
3	0.00289	0.017005	0.00316	0.00658	0.002514	0.01665
4	0.00148	0.004966	0.001981	0.01700	0.001406	0.007273
5	0.00359	0.035325	0.003028	0.008677	0.0057448	0.02208
6	0.00126	0.039990	0.001489	0.02409	0.0016464	0.019433

Table II shows the mean empirical risk obtained on the training and the validation data set (columns T and V , respectively) for the individuals which have been selected after five independent runs. The terminal encapsulation is very useful especially when the set of terminals is minimally sufficient or contains fewer terminals than minimally required (#1-4). In these cases, the protection of the most valuable leaves is critical, as each variable input has few lagged instances in the terminal set, which can be completely eliminated from the graph based individuals by a blind structural crossover. It is worth mentioning that the terminal encapsulation brings benefits when working with

small populations (#1, #3), by keeping unaltered the leaves belonging to the better adapted individuals, yet which tend to disappear. When **T** contains more lagged samples than necessary, the probability of eliminating all the leaves corresponding to a certain system variable is quite reduced and the terminal encapsulation has to be relaxed. However, enforcing the preservation of some input variables can be useful for the capacity of generalization of the designed neural model (#6). On the other hand, the sensitivity based encapsulation is helpful especially when dealing with large terminal sets. In this case, additional information concerning the usability of the inner substructures can be fruitfully for exploring through numerous possible neural architectures (#3-6). Details regarding the best models obtained for configurations #4-TE and #4-SE are given in the following. Both models provide better approximations on validation and training than an MLP/ RBF having comparable number of neurons (Fig. 2). However, being partially interconnected, the HNNs admit fewer parameters (#4-SE: 4 neurons, 23 parameters; #4-TE: 5 neurons, 34 parameters; MLP/RBF: 5 neurons, 86 parameters). Besides, the residuals analysis indicates that TE and SE were helpful for capturing the most important characteristic of the plant, as they lead to models which admit more autocorrelation coefficients within the 99% confidence interval than #4-WE and also better Theil score (TE: 0.6, SE: 0.8, WE: 0.9).

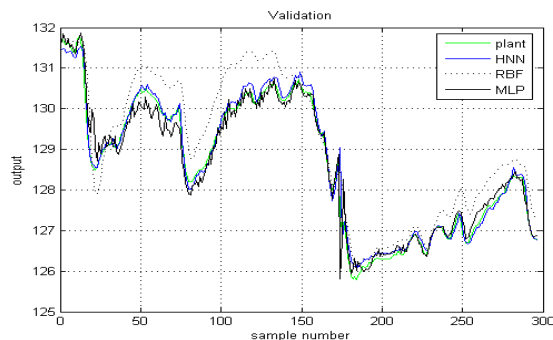


Fig. 2. Model #4-SE (validation data set) – comparison with MLP/RBF.

V. CONCLUSIONS

The proposed multiobjective graph GP-based algorithm allows an efficient design of accurate and compact HNNs, within the framework of data driven modeling. To ensure a better correlation between exploration and exploitation, the authors introduce two types of encapsulations which are targeted to protect valuable inner blocks within the DAG-based individuals.

The terminal encapsulation acts on the leaves of the graphs and enforces the preservation of certain terminals used in the best adapted individuals. The immediate effect lies in a more efficient variable reduction, which permits the selection of models featuring smaller lagged autocorrelation coefficients. The sensitivity-based encapsulation saves the best sub-graphs directly connected with the root

nodes and, hence, it guides the structural crossover to reuse of the most valuable genetic material of the parents for producing better adapted offspring.

The MOO graph GP with encapsulations can better deal with minimally/ non-minimally sufficient terminal sets and competing conventions. The approach is suitable for the design of dynamic or static nonlinear models, whenever the interdependency between the involved variables is not completely understood, large search spaces have to be scanned, or/and high model accuracy is required.

REFERENCES

- [1] P. J. Flemming, R. C. Purshouse "Evolutionary Algorithms in Control Systems Engineering: A Survey", in *Control Engineering Practice* 10, 2002, pp. 1223-1241.
- [2] J. Patra, R. Pal, B. Chatterji, G. Panda, "Identification of nonlinear dynamic systems using functional link artificial neural networks", in *IEEE Transactions on System, Man and Cybernetics, Part B: Cybernetics*, 29, 1999, pp. 254-262.
- [3] B. Igennik, M. Tabib – Azar, Le Clair S. R., "A net with complex weights", in *IEEE Transactions on Neural Networks*, 12, 2001, pp. 236-249.
- [4] K. Deb, *Multiobjective Optimization Using Evolutionary Algorithms*, Wiley&Sons, 2001.
- [5] J. Handl, J. Knowles, "Modes of Problem Solving with Multiple Objectives: Implications for Interpreting the Pareto Set and for Decision Making" in *Multiobjective Problem Solving from Nature*, J. Knowles · D. Corne, · K. Deb (Eds.), New York: Springer, 2008, pp.131-154.
- [6] K. Rodriguez-Vasquez., C. M Fonseca, P. J. Fleming, "Identifying the Structure of Nonlinear Dynamic Systems Using Multiobjective Genetic Programming" in *IEEE Transactions on Systems Man and Cybernetics, Part A – Systems and Humans*, 34, 2004, pp. 531-534.
- [7] Poli R., Langdon W. B., Mc Phee N. F., *A Field Guide to Genetic Programming*. Published via <http://lulu.com> (with contributions of J.R. Koza), [Online]. Available: <http://www.gp-field-guide.org.uk>, 2008.
- [8] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, Cambridge, MA: MIT Press, 1992.
- [9] L. Ferariu, B. Burlacu B., "Evolutionary Neural Networks with Heterogeneous Hidden Layers", in *Transactions on Automatic Control and Computer Science, Scientific Bulletin of "Politehnica" University of Timisoara*, 55 (69), 4, 2010, pp. 239-246.
- [10] A. Walker, J. F. Miller, "The Automatic Acquisition, Evolution and Reuse of Modules in Cartesian Genetic Programming", in. *IEEE Transactions on Evolutionary Computation*, 12 (4), 2008, pp. 397-417.
- [11] E. J. Hughes, "Fitness Assignment Methods for Many-Objective Problems", in *Multiobjective Problem Solving from Nature*, J. Knowles, · D. Corne, · K. Deb (Eds.), New York: Springer, 2008, pp. 307-330.
- [12] A. Patelli, L. Ferariu, "Elitist Multiobjective Nonlinear Systems Identification with Insular Evolution and Diversity Preservation" in *Proc. of 2010 IEEE World Congress on Computational Intelligence - Congress on Evolutionary Computation*, Barcelona, Spain, pp. 2076-2081, (ISI proceedings), IEEE catalog no. CFP10ICE-DVD.
- [13] L. Ferariu, M. Voicu, "Nonlinear System Identification Based on Evolutionary Dynamic Neural Networks with Hybrid Structure" in *Proc. of IFAC Congress*, Prague, Czech Republic, 2005.
- [14] A. J. Barton, J. J. Valdés, R. Orchard, "Neural networks with multiple general neuron models: A hybrid computational intelligence approach using Genetic Programming", in *Neural Networks*, 22, 2009, pp. 614-622.

Improving Rohn Approximation for the Real Parts of Interval Matrix Eigenvalues

Dorian Florescu, Mihaela Matcovschi, *Member, IEEE*, and Octavian Pastravanu, *Member, IEEE*

Abstract—The paper proposes a new approximation method for the upper and lower bounds of the real parts of interval matrix eigenvalues. As a theoretical support, we exploit the properties of the matrix measures corresponding to monotonic (or, equivalently, absolute) vector norms. The numerical approach focuses on the use of the results derived for the matrix measure defined by the Euclidean norm (or 2-norm). The computational procedure is formulated as a minimization problem of a linear objective function constrained by bilinear matrix inequalities. We prove (by mathematical arguments and, separately, by examples) that our estimations are better or, at least, as accurate as those provided by the method due to Rohn (see reference [6]).

Keywords: interval matrix, eigenvalue bounds, approximation methods, constrained minimization

I. INTRODUCTION

Let matrices $A^- = [a_{ij}^-]$, $A^+ = [a_{ij}^+] \in \mathbb{R}^{n \times n}$ satisfy the componentwise inequality $A^- \leq A^+$. The family (set) of real matrices

$$\mathcal{A} = \{A = [a_{ij}] \in \mathbb{R}^{n \times n} \mid A^- \leq A \leq A^+\}, \quad (1)$$

is called *interval matrix*. It is also denoted by

$$\mathcal{A} = [A^-, A^+] = A_0 + [-R, R], \quad (2)$$

where matrices $A_0 = [a_{ij}^0] = \frac{1}{2}(A^- + A^+)$ and $R = [r_{ij}] = \frac{1}{2}(A^+ - A^-) \geq 0$ represent the *center* and the *radius* of the interval matrix $\mathcal{A}$, respectively.

The *exact bounds* of the eigenvalue range of $\mathcal{A}$ (1) are defined for the real parts as follows:

$$\begin{aligned} R^+(\mathcal{A}) &= \max_{A \in \mathcal{A}} \max_{k=1, \dots, n} \operatorname{Re}\{\lambda_k(A)\}, \\ R^-(\mathcal{A}) &= \min_{A \in \mathcal{A}} \min_{k=1, \dots, n} \operatorname{Re}\{\lambda_k(A)\}, \end{aligned} \quad (3)$$

where $\lambda_k(A)$, $k=1, \dots, n$, denote the eigenvalues of matrix A . Generally speaking, these exact bounds are not available and, therefore, a series of works such as [1 – 7] proposed algorithms / methods for finding estimations. The above notations with a subscript „e”, refer to *estimated bounds* and

the following inequalities are obvious

$$R^+(\mathcal{A}) \leq R_e^+(\mathcal{A}), \quad R_e^-(\mathcal{A}) \leq R^-(\mathcal{A}). \quad (4)$$

An approximation procedure was issued by Rohn [6] giving the “practical bounds”

$$\begin{aligned} R_R^+(\mathcal{A}) &= \frac{1}{2} \lambda_{\max}(A_0 + A_0^T) + \frac{1}{2} \lambda_{\max}(R + R^T), \\ R_R^-(\mathcal{A}) &= \frac{1}{2} \lambda_{\min}(A_0 + A_0^T) - \frac{1}{2} \lambda_{\max}(R + R^T), \end{aligned} \quad (5)$$

with subscript „R” referring to Rohn.

Our work presents a new estimation method that may be regarded as a generalization of the results in [6]. The proposed estimation method exploits the connection between the eigenvalues of an arbitrary matrix A belonging to $\mathcal{A}$ and the values of different matrix measures [8] applied to A . For any *matrix measure*, we construct: • a family of *estimated upper bounds* for the eigenvalues of $\mathcal{A}$, from which we can determine the best estimation by *selecting the smallest value* in the family; • a family of *estimated lower bounds* for the eigenvalues of $\mathcal{A}$, from which we can determine the best estimation by *selecting the largest value* in the family.

For the *matrix measure corresponding to the 2-norm*, we develop a *robust numerical procedure* that computes the best estimations of the exact upper, and, respectively, lower bound. The procedure is formulated as a minimization problem of a linear objective function constrained by bilinear matrix inequalities. We prove (by mathematical arguments and, separately, by examples) that our estimations are better or, at least, as accurate as those proposed by [6].

The current paper is organized as follows. Section II presents the main theoretical result that, for any matrix measure, provides a family of estimated upper bounds and a family of estimated lower bounds for the eigenvalues of $\mathcal{A}$. Section III develops a numerical approach that utilizes the theoretical result of Section II corresponding to the matrix measure defined by the 2-nd norm; this approach computes the best estimations of the exact upper bound, as well as the best estimation of the exact lower bound. Section IV shows that the estimations computed by Section III are better or, at least, as accurate as those proposed by [6]. Section V illustrates the use of the computational procedure proposed in Section III for three relevant examples of interval matrices; comparisons with the estimations calculated in accordance with [6] are included.

The nomenclature and the notations are presented in Appendix I. The proofs of the key results are given in Appendix II.

Manuscript received September, 2001. This work was supported by CNMP, Romania, under Grant 12100/1.10.2008 - SICONA.

The authors are with the Department of Automatic Control and Applied Informatics from the Technical University “Gh. Asachi” of Iasi, Blvd. Mangeron 27, 700050 Iasi, Romania (e-mail: {mhanako, opastrav}@ac.tuiasi.ro).

II. MAIN RESULT

Let μ be the matrix measure on $\mathbb{R}^{n \times n}$ corresponding to a monotonic (or, equivalently, absolute) vector norm on $\mathbb{R}^n$. The following result provides estimations depending on n positive parameters for the exact bounds of the interval matrix eigenvalues. An adequate adjustment of these parameters can drive the estimations to values minimizing the distance from the exact bounds.

Theorem 1.

For any $A \in \mathcal{A}$ the following inequalities hold true:

$$\begin{aligned} \forall \mathbf{D} = \text{diag}\{d_1, \dots, d_n\}, d_i > 0, i = 1, \dots, n: \\ -b(\mathbf{D}) \leq \text{Re}\{\lambda_k(A)\} \leq B(\mathbf{D}), \quad \forall k = 1, \dots, n. \end{aligned} \quad (6)$$

where

$$B(\mathbf{D}) = \left\{ \mu(\mathbf{D}^{-1}A_0\mathbf{D}) + \mu(\mathbf{D}^{-1}R\mathbf{D}) \right\}, \quad (7-i)$$

$$b(\mathbf{D}) = \left\{ \mu(\mathbf{D}^{-1}(-A_0)\mathbf{D}) + \mu(\mathbf{D}^{-1}R\mathbf{D}) \right\}. \quad (7-ii)$$

Proof: See Appendix II. ■

Remark 1.

If the center matrix A_0 is *essentially nonnegative*, then for any $A = A_0 + \Delta \in \mathcal{A}$ the off-diagonal entries satisfy $|a_{ij}| = |a_{ij}^0 + \delta_{ij}| \leq a_{ij}^0 + r_{ij}$, $i \neq j$, whereas for the entries of the main diagonal we can write $a_{ii} \leq a_{ii}^0 + r_{ii}$. From Theorem 8.1.18 in [9] we get $\text{Re}\{\lambda_k(A)\} \leq \lambda_{\max}(A_0 + R)$, for all $k = 1, \dots, n$, or, in other words, we obtain the *exact right bound* $R^+(A) = \lambda_{\max}(A_0 + R)$. ■

Remark 2.

If the center matrix A_0 is *essentially nonpositive*, then we get the *exact left bound* $R^-(A) = -\lambda_{\max}(-A_0 + R)$. ■

Remark 3.

If the radius matrix R has equal entries, i.e. $r_{ij} = r > 0$, $i, j = 1, \dots, n$, then inequality (6) gets the simpler form:

$$\begin{aligned} \forall \mathbf{D} = \text{diag}\{d_1, \dots, d_n\}, d_i > 0, i = 1, \dots, n, \\ -\hat{b}(\mathbf{D}) - nr \leq \text{Re}\{\lambda_k(A)\} \leq \hat{B}(\mathbf{D}) + nr, \quad \forall k = 1, \dots, n. \end{aligned} \quad (8)$$

where

$$\hat{B}(\mathbf{D}) = \mu(\mathbf{D}^{-1}A_0\mathbf{D}), \quad (9-i)$$

$$\hat{b}(\mathbf{D}) = \mu(\mathbf{D}^{-1}(-A_0)\mathbf{D}). \quad \blacksquare \quad (9-ii)$$

Remark 4.

The search for the smallest values of $b(\mathbf{D})$, $B(\mathbf{D})$ in (6), and, respectively, of $\hat{b}(\mathbf{D})$, $\hat{B}(\mathbf{D})$ in (8) yields the best estimates of type $R_e^-(A)$, $R_e^+(A)$ that can be obtained by adjusting the positive diagonal entries of the generic matrix $\mathbf{D}$. This requires a rigorous numerical approach since $b(\mathbf{D})$

and $B(\mathbf{D})$ are nonlinear functions and their global minimization is not ensured by usual solvers. The next section deals with the numerical tractability in the case of the Euclidean norm. ■

III. NUMERICAL APPROACH

The numerical approach is convenient for the matrix measure μ_2 , corresponding to the 2-norm. The inequality $\mu_2(\mathbf{M}) \leq \frac{1}{2}[\mu_1(\mathbf{M}) + \mu_\infty(\mathbf{M})]$ shows that, for any $\mathbf{D}$, at least one of the measures μ_1 , μ_∞ yields larger values of $b(\mathbf{D})$ and $B(\mathbf{D})$ than resulting for μ_2 .

The current section exploits the general result expressed by Theorem 1 and transforms it into an effective computational tool. The key idea is to construct a well-defined minimization problem with adequate constraints, as a practical continuation of the theoretical development proposed by Theorem 1.

In the remainder of the paper, for brevity reasons, we use the following **writing convention** for functions depending on a diagonal matrix: a function $\Psi(\mathbf{D})$ with $\mathbf{D}$ a positive-definite diagonal matrix $\mathbf{D} = \text{diag}\{d_1, \dots, d_n\}$, $d_i > 0$, $i = 1, \dots, n$, is understood as a function defined on $(\mathbb{R}_+^*)^n$ through $\tilde{\Psi}(d_1, \dots, d_n) = (\Psi \circ \text{diag})([d_1, \dots, d_n]^T)$, where the operator $\text{diag}: (\mathbb{R}_+^*)^n \rightarrow \mathbb{R}^{n \times n}$ associates to a positive vector $\mathbf{d} = [d_1, \dots, d_n]^T \in (\mathbb{R}_+^*)^n$ the diagonal matrix $\mathbf{D} = \text{diag}\{d_1, \dots, d_n\} \in \mathbb{R}^{n \times n}$. We use the same **writing convention** in the case of functions in the form $\Psi(x_1, \dots, x_m, \mathbf{D})$ with $x_1, \dots, x_m \in \mathbb{R}$ and $\mathbf{D} = \text{diag}\{d_1, \dots, d_n\}$, that we understand as a function $\tilde{\Psi}(x_1, \dots, x_m, d_1, \dots, d_n)$ defined on $\mathbb{R} \times \dots \times \mathbb{R} \times (\mathbb{R}_+^*)^n$.

Denote by

$$\mathcal{D} = \left\{ \mathbf{D} = \text{diag}\{d_1, \dots, d_n\} \mid 1 \leq d_i \leq d_i^M, i = 1, \dots, n \right\}, \quad (10)$$

$$\begin{aligned} \mathcal{P} = \left\{ \mathbf{P} = \text{diag}\{p_1, \dots, p_n\} \mid 1 \leq p_i \leq p_i^M, i = 1, \dots, n \right\}, \\ p_i^M = (d_i^M)^2. \end{aligned} \quad (11)$$

two sets of diagonal matrices whose diagonal entries are adequately bounded from the numerical computation point of view.

Theorem 2.

(i) Define the following two optimization problems:

(O1-i) Minimize the objective function

$$B_2: (\mathbb{R}_+^*)^n \rightarrow \mathbb{R}, B_2(\mathbf{D}) = \mu_2(\mathbf{D}^{-1}A_0\mathbf{D}) + \mu_2(\mathbf{D}^{-1}R\mathbf{D}), \quad (12-i)$$

under the constraints $\mathbf{D} \in \mathcal{D}$.

(O2-i) Minimize the objective function

$$F: \mathbb{R} \times \mathbb{R} \times (\mathbb{R}_+^*)^n \rightarrow \mathbb{R}, F(\varepsilon_1, \varepsilon_2, \mathbf{P}) = \varepsilon_1 + \varepsilon_2, \quad (13-i)$$

under the constraints $\mathbf{P} \in \mathcal{P}$ and

$$2\varepsilon_1\mathbf{P}-\mathbf{A}_0\mathbf{P}-\mathbf{P}\mathbf{A}_0^T \succeq 0, \quad (14-i)$$

$$2\varepsilon_2\mathbf{P}-\mathbf{R}\mathbf{P}-\mathbf{P}\mathbf{R}^T \succeq 0. \quad (15-i)$$

Then

(a) (O1-i) has feasible solutions.

(b) (O2-i) has feasible solutions.

(c) If $(\varepsilon_1^*, \varepsilon_2^*, \mathbf{P}^*)$ is a solution to (O2-i), then

$\mathbf{D}^* = (\mathbf{P}^*)^{1/2}$ (i.e. $(\mathbf{D}^*)^2 = \mathbf{P}^*$) is a solution to (O1-i).

(ii) Define the following two optimization problems:

(O1-ii) Minimize the objective function

$$\begin{aligned} b_2 : (\mathbb{R}_+^*)^n &\rightarrow \mathbb{R}, \\ b_2(\mathbf{D}) &= \mu_2(\mathbf{D}^{-1}(-\mathbf{A}_0)\mathbf{D}) + \mu_2(\mathbf{D}^{-1}\mathbf{R}\mathbf{D}), \end{aligned} \quad (12-ii)$$

where $\mathbf{D} = \text{diag}\{d_1, \dots, d_n\}$, under the constraints $\mathbf{D} \in \mathcal{D}$.

(O2-ii) Minimize the objective function

$$\tilde{F} : \mathbb{R} \times \mathbb{R} \times (\mathbb{R}_+^*)^n \rightarrow \mathbb{R}, \quad F(\varepsilon_1, \varepsilon_2, \mathbf{P}) = \varepsilon_1 + \varepsilon_2, \quad (13-ii)$$

under the constraints $\mathbf{P} \in \mathcal{P}$ and

$$2\varepsilon_1\mathbf{P} + \mathbf{A}_0\mathbf{P} + \mathbf{P}\mathbf{A}_0^T \succeq 0, \quad (14-ii)$$

$$2\varepsilon_2\mathbf{P} - \mathbf{R}\mathbf{P} - \mathbf{P}\mathbf{R}^T \succeq 0. \quad (15-ii)$$

Then

(a) (O1-ii) has feasible solutions.

(b) (O2-ii) has feasible solutions.

(c) If $(\varepsilon_1^*, \varepsilon_2^*, \mathbf{P}^*)$ is a solution to (O2-ii), then

$\mathbf{D}^* = (\mathbf{P}^*)^{1/2}$ (i.e. $(\mathbf{D}^*)^2 = \mathbf{P}^*$) is a solution to (O1-ii).

Proof: See Appendix II. ■

A direct consequence of Theorem 2 is presented in the sequel.

Corollary 1.

(i) If $(\varepsilon_{1,b}^*, \varepsilon_{2,b}^*, \mathbf{P}_b^*)$ is a solution to (O2-i), then the value

$$R_e^+(\mathcal{A}) = F(\varepsilon_{1,b}^*, \varepsilon_{2,b}^*, \mathbf{P}_b^*) \quad (16-i)$$

$$R_e^+(\mathcal{A}) = B_2(\mathbf{D}_b^*), \text{ with } \mathbf{D}_b^* = (\mathbf{P}_b^*)^{1/2}, \quad (16-i)$$

represents the *best estimation* for the *exact right bound* $R^+(\mathcal{A})$, provided by Theorem 1 with μ_2 and $\mathbf{D} \in \mathcal{D}$.

(ii) If $(\varepsilon_{1,b}^*, \varepsilon_{2,b}^*, \mathbf{P}_b^*)$ is a solution to (O2-ii), then the value

$$R_e^-(\mathcal{A}) = -F(\varepsilon_{1,b}^*, \varepsilon_{2,b}^*, \mathbf{P}_b^*) \quad (16-ii)$$

$$R_e^-(\mathcal{A}) = -b_2(\mathbf{D}_b^*), \text{ with } \mathbf{D}_b^* = (\mathbf{P}_b^*)^{1/2}, \quad (16-ii)$$

represents the *best estimation* for the *exact left bound* $R^-(\mathcal{A})$, provided by Theorem 1 with the measure μ_2 and $\mathbf{D} \in \mathcal{D}$.

Proof: See Appendix II. ■

Remark 4.

The constraints (14) and (15) occurring in Theorem 2 are BMIs (Bilinear Matrix Inequalities) because of the products

$\varepsilon_1\mathbf{P}$ and $\varepsilon_2\mathbf{P}$, where $\varepsilon_1, \varepsilon_2, \mathbf{P}$ are variables for the minimization problem. The MathWorks Inc. has not developed a BMI solver yet, but one can use the MATLAB-compatible software commercialized by TOMLAB Optimization [10]. ■

Remark 5.

In the particular case when the radius matrix $\mathbf{R}$ has equal entries, i.e. $r_{ij} = r > 0, i, j = 1, \dots, n$, the minimization of the functions $\hat{b}(\mathbf{D}), \hat{B}(\mathbf{D})$ occurring in inequality (8) – expressed by the 2-norm – can be numerically approached as substantially simpler minimization problems. These are also tractable by means of the **gevp** solver available in the Robust Control Toolbox developed by The MathWorks Inc. ■

IV. COMPARISON WITH THE BOUNDS PRESENTED IN [6]

Theorem 2 always provides better estimations or, at least, as accurate as those given by [6]. Indeed, by considering the particular case $\mathbf{D} = \mathbf{I}$, where $\mathbf{I}$ is the unit matrix, we have the following discussion:

- For $B_2(\mathbf{D}) = \mu_2(\mathbf{D}^{-1}\mathbf{A}_0\mathbf{D}) + \mu_2(\mathbf{D}^{-1}\mathbf{R}\mathbf{D})$, corresponding to the upper bound in inequality (6), we get $B_2(\mathbf{I}) = \frac{1}{2}\lambda_{\max}(\mathbf{A}_0 + \mathbf{A}_0^T) + \frac{1}{2}\lambda_{\max}(\mathbf{R} + \mathbf{R}^T)$ that coincides with the right margin in (5), given by [6]. On the other hand, for $\mathbf{D}^*$ resulting from Theorem 2 (i), since $\mathbf{I} \in \mathcal{D}$, we certainly have $B_2(\mathbf{D}^*) \leq B_2(\mathbf{I})$, regardless of the concrete entries of the matrices $\mathbf{A}_0$ and $\mathbf{R}$.

- For $-b_2(\mathbf{D}) = -\{\mu_2(\mathbf{D}^{-1}(-\mathbf{A}_0)\mathbf{D}) + \mu_2(\mathbf{D}^{-1}\mathbf{R}\mathbf{D})\}$, corresponding to the lower bound in inequality (6), we get

$$\begin{aligned} -b_2(\mathbf{I}) &= -\frac{1}{2}\lambda_{\max}(-\mathbf{A}_0 - \mathbf{A}_0^T) - \frac{1}{2}\lambda_{\max}(\mathbf{R} + \mathbf{R}^T) = \\ &= -\frac{1}{2}(-\lambda_{\min}(\mathbf{A}_0 + \mathbf{A}_0^T)) - \frac{1}{2}\lambda_{\max}(\mathbf{R} + \mathbf{R}^T) = \\ &= \frac{1}{2}\lambda_{\min}(\mathbf{A}_0 + \mathbf{A}_0^T) - \frac{1}{2}\lambda_{\max}(\mathbf{R} + \mathbf{R}^T), \end{aligned}$$

that coincides with the left margin in (5), given by [6]. On the other hand, for $\mathbf{D}^*$ resulting from Theorem 2 (ii), since $\mathbf{I} \in \mathcal{D}$, we certainly have $-b_2(\mathbf{I}) \leq -b_2(\mathbf{D}^*)$, regardless of the concrete entries of the matrices $\mathbf{A}_0$ and $\mathbf{R}$.

V. CASE STUDIES

For all three examples considered below we have used Corollary 1 implemented in MATLAB by the help of the Multi-Parametric Toolbox [11] and the **penbmi** solver produced by TOMLAB Optimization [10]. For each example we list the estimations for the right and left bounds of the eigenvalues, obtained by our method (Theorem 2), as well as by the method proposed in [6].

All three examples prove that our estimations are better than those resulting from the method in [6] – as theoretically shown in Section IV.

Example 1.

It is taken from [1] (Example 2.7). Let $\mathcal{A}$ be defined by (2) with

$$\begin{aligned} A_0 &= \begin{pmatrix} -4.5 & -8.5 & 14.5 & 4.8 & -1.1 \\ 17.5 & 17.5 & 1.5 & 4.5 & 10.5 \\ 17.1 & -3.1 & 2 & -12.5 & 6.2 \\ 18.5 & 2.5 & 18.5 & 5.5 & 6.5 \\ 13.5 & 18.5 & 9.5 & -17.5 & 10.5 \end{pmatrix}, \\ R &= \begin{pmatrix} 0.4192 & 0.2849 & 0.9505 & 0.1881 & 0.0690 \\ 1.2234 & 0.6951 & 0.0620 & 0.2948 & 0.8795 \\ 0.6354 & 0.1318 & 0.1189 & 0.7072 & 0.4443 \\ 0.9459 & 0.1941 & 0.9053 & 0.1022 & 0.4554 \\ 1.3267 & 1.4923 & 0.6684 & 0.8487 & 0.1203 \end{pmatrix}. \end{aligned} \quad (17)$$

• Rohn's method (5) yields the bounds $R_R^-(\mathcal{A}) = -22.8389$ and $R_R^+(\mathcal{A}) = 36.2348$.

• Our method yields the bounds $R_e^-(\mathcal{A}) = -21.1528$ and $R_e^+(\mathcal{A}) = 31.5764$, calculated from the solutions to the optimization problems (O2-i) and (O2-ii) that allow the computation of the diagonal matrices $D_B^* = (P_B^*)^{1/2}$ and $D_b^* = (P_b^*)^{1/2}$ as
 $D_B^* = \text{diag}\{11.1762, 22.4688, 15.2185, 23.2040, 28.3771\}$,
 $D_b^* = \text{diag}\{7.9971, 11.5695, 11.0384, 5.5768, 10.3056\}$. ■

Example 2.

It is taken from [1] - Example 2.8. Let $\mathcal{A}$ be defined by (2) with

$$\begin{aligned} A_0 &= \begin{pmatrix} 4 & 6 & 13 & 1 \\ -4 & -5 & -16 & -4 \\ 1 & 2 & 6 & 1 \\ 0 & -2 & -10 & -1 \end{pmatrix}, \\ R &= \begin{pmatrix} 0.4659 & 0.5192 & 0.8321 & 0.1067 \\ 0.5292 & 0.5870 & 2.3991 & 0.5847 \\ 0.0559 & 0.1721 & 0.6291 & 0.0762 \\ 0 & 0.1099 & 1.0711 & 0.1201 \end{pmatrix}. \end{aligned} \quad (18)$$

• Rohn's method (5) yields the bounds $R_R^-(\mathcal{A}) = -14.2504$ and $R_R^+(\mathcal{A}) = 16.1504$.

• Our method yields the bounds $R_e^-(\mathcal{A}) = -8.0996$ and $R_e^+(\mathcal{A}) = 10.3989$, calculated from the solutions to the optimization problems (O2-i) and (O2-ii) that allow the computation of the diagonal matrices
 $D_B^* = (P_B^*)^{1/2} = \text{diag}\{25.7839, 20.9318, 7.0546, 19.8995\}$,
 $D_b^* = (P_b^*)^{1/2} = \text{diag}\{15.8990, 12.8932, 4.0129, 8.9971\}$. ■

Example 3.

It is taken from [1] - Example 2.8 slightly modified. Let $\mathcal{A}$ be defined by (2) with

$$\begin{aligned} A_0 &= \begin{pmatrix} 4 & 6 & 13 & 1 \\ -4 & -5 & -16 & -4 \\ 1 & 2 & 6 & 1 \\ 0 & -2 & -10 & -1 \end{pmatrix}, \\ R &= \begin{pmatrix} 0.0327 & 0.0396 & 0.0445 & 0.0029 \\ 0.0136 & 0.0267 & 0.1163 & 0.0124 \\ 0.0084 & 0.0114 & 0.0222 & 0.007 \\ 0 & 0.0089 & 0.0695 & 0.0062 \end{pmatrix}. \end{aligned} \quad (19)$$

• Rohn's method (5) yields the bounds $R_R^-(\mathcal{A}) = -12.0210$ and $R_R^+(\mathcal{A}) = 13.9210$.

• Our method yields the bounds $R_e^-(\mathcal{A}) = -6.5520$ and $R_e^+(\mathcal{A}) = 8.8289$, calculated from the solutions to the optimization problems (O2-i) and (O2-ii) allow the computation of the diagonal matrices

$$\begin{aligned} D_B^* &= (P_B^*)^{1/2} = \text{diag}\{34.1985, 27.3105, 9.5085, 27.4557\}, \\ D_b^* &= (P_b^*)^{1/2} = \text{diag}\{16.6630, 13.2364, 4.2182, 9.3819\}. \quad \blacksquare \end{aligned}$$

VI. CONCLUSIONS

The paper develops a theoretical support for estimating the upper and lower bounds of the real parts of interval matrix eigenvalues. The main result is stated in terms of an arbitrary matrix measure, weighted by a diagonal matrix with positive entries. The numerical approach exploits the case of the matrix measure corresponding to the 2-nd norm and yields a minimization problem with a linear objective function and BMI constraints. We prove by simple mathematical constructions, and, we also illustrate by examples that our estimations are better, or at least as accurate as the estimations derived by Rohn [6].

REFERENCES

- [1] M. Hladík, D. Daney, and E. Tsigaridas. Bounds on real eigenvalues and singular values of interval matrices. *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 4, pp. 2116–2129, 2010.
- [2] Y. Juang, and C. Shao. Stability analysis of dynamic interval systems, *Int. J. Control*, vol. 49, no. 4, pp. 1401–1408, 1989.
- [3] H. Leng, Z. He, and Q. Yuan. Computing bounds to real eigenvalues of real-interval matrices, *Int. J. Numer. Meth. Engng.*, vol. 74, no. 4, pp. 523–530, 2008.
- [4] L. Kolev, and S. Petrakieva. Assessing the stability of linear time-invariant continuous interval dynamic systems, *IEEE Trans. on Automatic Control*, vol. 50, no. 3, pp. 393–397, 2005.
- [5] J. Rohn, Stability of interval matrices: The real eigenvalue case, *IEEE Trans. on Automatic Control*, vol. 37, no. 10, pp. 1604–1605, 1992.
- [6] J. Rohn, Bounds on Eigenvalues of Interval Matrices, *ZAMM, Z. Angew. Math. Mech.*, vol. 78, Suppl. 3, S1049–S1050, 1998.
- [7] S.S. Wang, and W.G. Lin, A new approach to the stability analysis of interval systems, *Control-Theory Adv. Technol.*, vol. 7, no. 2, pp. 271–284, 1991.
- [8] W.A. Coppel. *Stability and Asymptotic Behavior of Differential Equations*, D.C. Heath and Company: Boston, 1965.
- [9] R.A. Horn, C.R. Johnson. *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1991.
- [10] TOMLAB Optimization, <http://tomopt.com/>.
- [11] Multi-Parametric Toolbox (MPT), <http://control.ee.ethz.ch/~mpt/>.
- [12] O. Pastravanu, M. Voicu, Generalized matrix diagonal stability and linear dynamical systems, *Linear Algebra and its Applications*, no. 419, pp. 299–310, 2006.

APPENDIX I. NOTATION AND NOMENCLATURE

- For a vector $\mathbf{x} = [x_1 \dots x_n]^T \in \mathbb{R}^n$:

$\|\mathbf{x}\|$ is an arbitrary vector norm.

$\|\mathbf{x}\|_p$ is the Hölder vector p -norm defined by

$$\|\mathbf{x}\|_p = [|x_1|^p + \dots + |x_n|^p]^{1/p} \text{ for } 1 \leq p < \infty, \text{ and by } \|\mathbf{x}\|_\infty = \max_{1 \leq i \leq n} |x_i| \text{ for } p = \infty.$$

- For a square matrix $\mathbf{M} = [m_{ij}] \in \mathbb{R}^{n \times n}$:

$\|\mathbf{M}\|$ is the matrix norm induced by the vector norm

$$\|\bullet\| \text{ through } \|\mathbf{M}\| = \sup_{\mathbf{x} \neq 0} \frac{\|\mathbf{M}\mathbf{x}\|}{\|\mathbf{x}\|} = \max_{\|\mathbf{x}\|=1} \|\mathbf{M}\mathbf{x}\|.$$

$\|\mathbf{M}\|_p$ is the matrix norm induced by the vector p -norm

$$\|\bullet\|_p, 1 \leq p \leq \infty;$$

$$\mu(\mathbf{M}) = \lim_{h \downarrow 0} h^{-1} (\|\mathbf{I} + h\mathbf{M}\| - 1) \text{ is the matrix measure ([8],$$

pp. 41), based on the matrix norm $\|\bullet\|$,

$$\mu_p(\mathbf{M}) = \lim_{h \downarrow 0} h^{-1} (\|\mathbf{I} + h\mathbf{M}\|_p - 1) \text{ is the matrix measure,}$$

based on the matrix norm $\|\bullet\|_p, 1 \leq p \leq \infty$.

Let $\sigma(\mathbf{M}) = \{z \in \mathbb{C} \mid \det(z\mathbf{I} - \mathbf{M}) = 0\}$ be the spectrum of $\mathbf{M}$ and $\lambda_i(\mathbf{M}) \in \sigma(\mathbf{M}), i = 1, \dots, n$, denote its eigenvalues.

If $\mathbf{M}$ is nonnegative (all entries are nonnegative), its spectral radius $\lambda_{\max}(\mathbf{M})$ is an eigenvalue such that $|\lambda_i(\mathbf{M})| \leq \lambda_{\max}(\mathbf{M}), i = 1, \dots, n$. If $\mathbf{M}$ is essentially nonnegative (all off-diagonal entries are nonnegative), then it has a real eigenvalue, denoted by $\lambda_{\max}(\mathbf{M})$, such that $\text{Re}\{\lambda_i(\mathbf{M})\} \leq \lambda_{\max}(\mathbf{M}), i = 1, \dots, n$, - e.g. Lemma 1 in [12].

If $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{m \times n}$, then “ $\mathbf{X} \leq \mathbf{Y}$ ” and “ $\mathbf{X} < \mathbf{Y}$ ” mean componentwise inequalities. $|\mathbf{x}|$ denotes the nonnegative matrix built with the absolute values of the elements of $\mathbf{X}$.

If $\mathbf{M} \in \mathbb{R}^{n \times n}$ is symmetric, then “ $\mathbf{M} \preceq 0$ ” (“ $\mathbf{M} \succeq 0$ ”) means $\mathbf{M}$ is semi-negative (semi-positive) definite.

APPENDIX II. PROOFS OF THE KEY RESULTS

Lemma 1.

If $-\mathbf{R} \leq \mathbf{A} \leq \mathbf{R}$, then for any matrix measure μ corresponding to a monotonic (equivalently, absolute) vector norm, the following inequality holds true:

$$\mu(\mathbf{D}^{-1}\mathbf{A}\mathbf{D}) \leq \mu(\mathbf{D}^{-1}\mathbf{R}\mathbf{D}).$$

Proof of Lemma 1: For a positive scalar $h > 0$ and a vector $\mathbf{y} \in \mathbb{R}^n$, consider the vector $\mathbf{v} = (\mathbf{I} + h\mathbf{D}^{-1}\mathbf{A}\mathbf{D})\mathbf{y}$ with the generic element $v_i = h \sum_{j=1, j \neq i}^n \frac{\delta_{ij} d_j}{d_i} y_j + (1 + h\delta_{ii}) y_i, i = 1, \dots, n$.

Then, for the absolute value we can write

$$|v_i| = |h \sum_{j=1, j \neq i}^n \frac{\delta_{ij} d_j}{d_i} y_j + (1 + h\delta_{ii}) y_i| \leq h \sum_{j=1, j \neq i}^n \frac{|\delta_{ij}| d_j}{d_i} |y_j| +$$

$$(1 + h|\delta_{ii}|) |y_i| \leq h \sum_{j=1, j \neq i}^n \frac{r_{ij} d_j}{d_i} |y_j| + (1 + h r_{ii}) |y_i|,$$

meaning the componentwise vector inequality $|(\mathbf{I} + h\mathbf{D}^{-1}\mathbf{A}\mathbf{D})\mathbf{y}| \leq (\mathbf{I} + h\mathbf{D}^{-1}\mathbf{R}\mathbf{D})|\mathbf{y}|$.

For monotonic vector norms we have

$$\|(\mathbf{I} + h\mathbf{D}^{-1}\mathbf{A}\mathbf{D})\mathbf{y}\| \leq \|(\mathbf{I} + h\mathbf{D}^{-1}\mathbf{R}\mathbf{D})\mathbf{y}\| \leq$$

$$\|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{R}\mathbf{D}\| \|\mathbf{y}\| = \|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{R}\mathbf{D}\| \|\mathbf{y}\|.$$

Thus, for $\mathbf{y} \neq \mathbf{0}$ we get $\frac{\|(\mathbf{I} + h\mathbf{D}^{-1}\mathbf{A}\mathbf{D})\mathbf{y}\|}{\|\mathbf{y}\|} \leq \|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{R}\mathbf{D}\|$,

which implies the matrix norm inequality

$$\|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{A}\mathbf{D}\| \leq \|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{R}\mathbf{D}\|, \forall h > 0. \text{ Hence,}$$

$$h^{-1} (\|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{A}\mathbf{D}\| - 1) \leq h^{-1} (\|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{R}\mathbf{D}\| - 1), \forall h > 0 \Rightarrow$$

$$\lim_{h \downarrow 0} h^{-1} (\|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{A}\mathbf{D}\| - 1) \leq \lim_{h \downarrow 0} h^{-1} (\|\mathbf{I} + h\mathbf{D}^{-1}\mathbf{R}\mathbf{D}\| - 1). \quad \blacksquare$$

Lemma 2.

For any $a, b, x, y \in \mathbb{R}$, satisfying the inequality $a + b < x + y$, one can find $\alpha_1, \alpha_2 \in \mathbb{R}$ such that $a < \alpha_1$, $b < \alpha_2$ and $\alpha_1 + \alpha_2 < x + y$.

Proof of Lemma 2 There exists $\alpha \in \mathbb{R}$ such that $a + b < \alpha < x + y$. Let $\alpha_1 = a + \frac{\alpha - (a + b)}{2}$ and $\alpha_2 = b + \frac{\alpha - (a + b)}{2}$.

Then $a < \alpha_1$, $b < \alpha_2$ and $\alpha_1 + \alpha_2 = \alpha$. $\blacksquare$

Proof of Theorem 1.

Let $\mathbf{A} = \mathbf{A}_0 + \mathbf{A}$, with $-\mathbf{R} \leq \mathbf{A} \leq \mathbf{R}$, be an arbitrary matrix belonging to the interval matrix $\mathcal{A}$. For $\mathbf{D} = \text{diag}\{d_1 \dots d_n\}$, $d_1, \dots, d_n > 0$, we can write the inequality

$$\text{Re}\{\lambda(\mathbf{A})\} = \text{Re}\{\lambda(\mathbf{D}^{-1}\mathbf{A}\mathbf{D})\} = \text{Re}\{\lambda(\mathbf{D}^{-1}(\mathbf{A}_0 + \mathbf{A})\mathbf{D})\} \leq$$

$$\mu(\mathbf{D}^{-1}(\mathbf{A}_0 + \mathbf{A})\mathbf{D}) \leq \mu(\mathbf{D}^{-1}\mathbf{A}_0\mathbf{D}) + \mu(\mathbf{D}^{-1}\mathbf{A}\mathbf{D})$$

with $\mu(\mathbf{D}^{-1}\mathbf{A}\mathbf{D}) \leq \mu(\mathbf{D}^{-1}\mathbf{R}\mathbf{D})$ in accordance with Lemma 1.

Finally, we get $\text{Re}\{\lambda(\mathbf{A})\} \leq \mu(\mathbf{D}^{-1}\mathbf{A}_0\mathbf{D}) + \mu(\mathbf{D}^{-1}\mathbf{R}\mathbf{D}) = B(\mathbf{D})$.

Similarly we can write

$$\text{Re}\{\lambda(\mathbf{A})\} = \text{Re}\{\lambda(\mathbf{D}^{-1}\mathbf{A}\mathbf{D})\} = \text{Re}\{\lambda(\mathbf{D}^{-1}(\mathbf{A}_0 + \mathbf{A})\mathbf{D})\} \geq$$

$$-\mu(\mathbf{D}^{-1}(-\mathbf{A}_0 - \mathbf{A})\mathbf{D}) \geq -\mu(\mathbf{D}^{-1}(-\mathbf{A}_0)\mathbf{D}) - \mu(\mathbf{D}^{-1}(-\mathbf{A})\mathbf{D})$$

where we also apply Lemma 1 to show that $\mu(\mathbf{D}^{-1}(-\mathbf{A})\mathbf{D}) \leq \mu(\mathbf{D}^{-1}\mathbf{R}\mathbf{D})$

Hence, we obtain the lower bound as a function of $\mathbf{D}$

$$\text{Re}\{\lambda(\mathbf{A})\} \geq -\mu(\mathbf{D}^{-1}(-\mathbf{A}_0)\mathbf{D}) - \mu(\mathbf{D}^{-1}\mathbf{R}\mathbf{D}) = -b(\mathbf{D}). \quad \blacksquare$$

Proof of Theorem 2 (i).

a) Function B_2 (12-i), regarded as depending on the variables $d_i, i = 1, \dots, n$, is continuous on the compact set defined by $\mathcal{D}$ in $(\mathbb{R}_+^*)^n$. Subsequently B_2 reaches its minimum value and (O1-i) has solutions.

b) The proof is based on four steps:

Step 1. Starting from (O2-i), we define the following optimization problem:

(O2-i) Minimize the objective function

$$\underline{F}: \mathbb{R} \times \mathbb{R} \times (\mathbb{R}_+^*)^n \rightarrow \mathbb{R}, \quad \underline{F}(\varepsilon_1, \varepsilon_2, \mathbf{D}) = \varepsilon_1 + \varepsilon_2, \quad (\text{A1})$$

subject to $\mathbf{D} \in \mathcal{D}$ and

$$\mu_2(\mathbf{D}^{-1} \mathbf{A}_0 \mathbf{D}) \leq \varepsilon_1, \quad (\text{A2})$$

$$\mu_2(\mathbf{D}^{-1} \mathbf{R} \mathbf{D}) \leq \varepsilon_2. \quad (\text{A3})$$

We shall prove that (O2-i) and (O2-i) are equivalent.

Step 2. For the upper bounds $d_i^M, i=1, \dots, n$, considered in (10), we denote $\mathbf{D}^M = \text{diag}\{d_1^M, \dots, d_n^M\}$ and $\varepsilon_1^M = \mu((\mathbf{D}^M)^{-1} \mathbf{A}_0 \mathbf{D}^M)$, $\varepsilon_2^M = \mu((\mathbf{D}^M)^{-1} \mathbf{R} \mathbf{D}^M)$.

Starting from (O2-i), we define the following optimization problem that has a supplementary constraint.

(O2-i) Minimize the objective function (A1) subject to $\mathbf{D} \in \mathcal{D}$, (A2), (A3) and

$$\varepsilon_1 + \varepsilon_2 \leq \varepsilon_1^M + \varepsilon_2^M \quad (\text{A4})$$

We shall prove that (O2-i) has feasible solutions.

Step 3. We shall prove that (O2-i) has solutions and these are given by the solutions to (O2-i).

Step 4. Relying on the equivalence between (O2-i) and (O2-i), we shall prove that (O2-i) has solutions and these are given by the solutions to (O2-i).

Proof of Step 1. For the matrix inequality constraints (14-i), (15-i), by taking $\mathbf{P} = \mathbf{D}^2$, we have the equivalent forms (A2), (A3), respectively. Indeed (see Appendix I):

$$\begin{aligned} \mu_2(\mathbf{D}^{-1} \mathbf{A}_0 \mathbf{D}) \leq \varepsilon_1 &\Leftrightarrow \frac{1}{2} \lambda_{\max}(\mathbf{D}^{-1} \mathbf{A}_0 \mathbf{D} + \mathbf{D} \mathbf{A}_0^T \mathbf{D}^{-1}) \leq \varepsilon_1 \Leftrightarrow \\ &\Leftrightarrow \lambda_{\max}(\mathbf{D}^{-1} \mathbf{A}_0 \mathbf{D} + \mathbf{D} \mathbf{A}_0^T \mathbf{D}^{-1}) - 2\varepsilon_1 \leq 0 \Leftrightarrow \\ &\Leftrightarrow \lambda_{\max}(\mathbf{D}^{-1} \mathbf{A}_0 \mathbf{D} + \mathbf{D} \mathbf{A}_0^T \mathbf{D}^{-1} - 2\varepsilon_1 \mathbf{I}) \leq 0 \Leftrightarrow \\ &\Leftrightarrow (\mathbf{D}^{-1} \mathbf{A}_0 \mathbf{D} + \mathbf{D} \mathbf{A}_0^T \mathbf{D}^{-1}) - 2\varepsilon_1 \mathbf{I} \preceq 0 \Leftrightarrow \\ &\Leftrightarrow \mathbf{A}_0 \mathbf{D}^2 + \mathbf{D}^2 \mathbf{A}_0^T - 2\varepsilon_1 \mathbf{D}^2 \preceq 0 \Leftrightarrow \\ &\Leftrightarrow 2\varepsilon_1 \mathbf{D}^2 - \mathbf{A}_0 \mathbf{D}^2 - \mathbf{D}^2 \mathbf{A}_0^T = 2\varepsilon_1 \mathbf{P} - \mathbf{A}_0 \mathbf{P} - \mathbf{P} \mathbf{A}_0^T \succeq 0. \end{aligned}$$

We proceed in a similar manner to show that (15-i) $\Leftrightarrow$ (A3).

Proof of Step 2. Let us show that the set defined in $\mathbb{R} \times \mathbb{R} \times (\mathbb{R}_+^*)^n$ by the constraints $\mathbf{D} \in \mathcal{D}$, (A2), (A3), (A4) is compact. *Boundedness:* for $\mathbf{D} \in \mathcal{D}$, the continuous functions $\mu_2(\mathbf{D}^{-1} \mathbf{A}_0 \mathbf{D})$, $\mu_2(\mathbf{D}^{-1} \mathbf{R} \mathbf{D})$ reach their minimum values, denoted as $\mu_2^m(\mathbf{A}_0)$, $\mu_2^m(\mathbf{R})$, respectively. Hence, $\mu_2^m(\mathbf{A}_0) \leq \varepsilon_1$, $\mu_2^m(\mathbf{R}) \leq \varepsilon_2$. Taking (A4) into consideration, we get $\mu_2^m(\mathbf{A}_0) \leq \varepsilon_1 \leq \varepsilon_1^M + \varepsilon_2^M - \mu_2^m(\mathbf{R})$, $\mu_2^m(\mathbf{R}) \leq \varepsilon_2 \leq \varepsilon_1^M + \varepsilon_2^M - \mu_2^m(\mathbf{A}_0)$. *Closure:* by choosing any convergent sequence $(\varepsilon_1^k, \varepsilon_2^k, \mathbf{D}^k)_{k \in \mathbb{N}^*}$ inside the set, the limit $\lim_{k \rightarrow \infty} (\varepsilon_1^k, \varepsilon_2^k, \mathbf{D}^k) = (\varepsilon_1^0, \varepsilon_2^0, \mathbf{D}^0)$ satisfies the constraints.

Thus, the objective function (A1) is defined on a compact set in $\mathbb{R} \times \mathbb{R} \times (\mathbb{R}_+^*)^n$ and it reaches its minimum,

i.e. (O2-i) has solutions.

Proof of Step 3. Denote by $\underline{\Omega} \subset \mathbb{R} \times \mathbb{R} \times (\mathbb{R}_+^*)^n$ the domain of the function $\underline{F}(\varepsilon_1, \varepsilon_2, \mathbf{D})$ in problem (O2-i), and by $\overline{\Omega} \subset \mathbb{R} \times \mathbb{R} \times (\mathbb{R}_+^*)^n$ the domain of the function $\overline{F}(\varepsilon_1, \varepsilon_2, \mathbf{D})$ in problem (O2-i). Let $(\varepsilon_1^*, \varepsilon_2^*, \mathbf{D}^*) \in \overline{\Omega}$ be a solution to (O2-i), i.e. $\underline{F}(\varepsilon_1^*, \varepsilon_2^*, \mathbf{D}^*) = \varepsilon_1^* + \varepsilon_2^* = \min_{\underline{\Omega}} \underline{F}$. On the other

hand we have $\underline{\Omega} = \overline{\Omega} \cup (\underline{\Omega} \setminus \overline{\Omega})$, $\overline{\Omega} \cap (\underline{\Omega} \setminus \overline{\Omega}) = \emptyset$, and we

can write $\inf_{\underline{\Omega}} \underline{F} = \min \left\{ \inf_{\overline{\Omega}} \underline{F}, \inf_{\underline{\Omega} \setminus \overline{\Omega}} \underline{F} \right\}$, where $\inf_{\overline{\Omega}} \underline{F} = \min_{\overline{\Omega}} \underline{F} = \varepsilon_1^* + \varepsilon_2^*$ and $\inf_{\underline{\Omega} \setminus \overline{\Omega}} \underline{F} = \varepsilon_1^M + \varepsilon_2^M$. Hence, $\inf_{\underline{\Omega}} \underline{F} = \varepsilon_1^* + \varepsilon_2^* = \min_{\underline{\Omega}} \underline{F}$. In other words (O2-i) has solutions and

these are given by the solutions to (O2-i).

Proof of Step 4. It is obvious.

c) We use contradiction. Let ε_1^* , ε_2^* , $\mathbf{P}^* = (\mathbf{D}^*)^2$ be a solution to the minimization problem (O2-i) and assume that $\mathbf{D}^*$ does not ensure the minimization of the objective function $B_2(\mathbf{D}) = \mu_2(\mathbf{D}^{-1} \mathbf{A}_0 \mathbf{D}) + \mu_2(\mathbf{D}^{-1} \mathbf{R} \mathbf{D})$ considered in (O1-i). This means there exists $\mathbf{D}^{**} \neq \mathbf{D}^*$ such that $\mu_2((\mathbf{D}^{**})^{-1} \mathbf{A}_0 \mathbf{D}^{**}) + \mu_2((\mathbf{D}^{**})^{-1} \mathbf{R} \mathbf{D}^{**}) < \mu_2((\mathbf{D}^*)^{-1} \mathbf{A}_0 \mathbf{D}^*) + \mu_2((\mathbf{D}^*)^{-1} \mathbf{R} \mathbf{D}^*)$.

By applying Lemma 2, we can find $\alpha_1, \alpha_2 \in \mathbb{R}$ such that

$$\begin{aligned} \mu_2((\mathbf{D}^{**})^{-1} \mathbf{A}_0 \mathbf{D}^{**}) &< \alpha_1, \quad \mu_2((\mathbf{D}^{**})^{-1} \mathbf{R} \mathbf{D}^{**}) < \alpha_2 \text{ and} \\ \alpha_1 + \alpha_2 &< \mu_2((\mathbf{D}^*)^{-1} \mathbf{A}_0 \mathbf{D}^*) + \mu_2((\mathbf{D}^*)^{-1} \mathbf{R} \mathbf{D}^*). \end{aligned}$$

Thus, $(\varepsilon_1^*, \varepsilon_2^*, \mathbf{P}^*)$ with $\mathbf{P}^* = (\mathbf{D}^*)^2$ is a solution to (O2-i),

whereas $(\alpha_1, \alpha_2, \mathbf{P}^{**})$ with $\mathbf{P}^{**} = (\mathbf{D}^{**})^2$ satisfies the constraints (A2) $\Leftrightarrow$ (14-i), (A3) $\Leftrightarrow$ (15-i) in (O2-i), but may not ensure the minimization of the objective function. This implies $\varepsilon_1^* + \varepsilon_2^* < \alpha_1 + \alpha_2$, and we get the contradiction

$$\begin{aligned} \mu_2((\mathbf{D}^*)^{-1} \mathbf{A}_0 \mathbf{D}^*) + \mu_2((\mathbf{D}^*)^{-1} \mathbf{R} \mathbf{D}^*) &\leq \varepsilon_1^* + \varepsilon_2^* < \alpha_1 + \alpha_2 < \\ \mu_2((\mathbf{D}^*)^{-1} \mathbf{A}_0 \mathbf{D}^*) + \mu_2((\mathbf{D}^*)^{-1} \mathbf{R} \mathbf{D}^*). \end{aligned}$$

Subsequently, for any $\mathbf{D}^{**} \neq \mathbf{D}^*$ we have the inequality $\mu_2((\mathbf{D}^{**})^{-1} \mathbf{A}_0 \mathbf{D}^{**}) + \mu_2((\mathbf{D}^{**})^{-1} \mathbf{R} \mathbf{D}^{**}) \geq \mu_2((\mathbf{D}^*)^{-1} \mathbf{A}_0 \mathbf{D}^*) + \mu_2((\mathbf{D}^*)^{-1} \mathbf{R} \mathbf{D}^*)$, showing that $\mathbf{D}^*$ is a solution to (O1-i). ■

Proof of Theorem 2 (ii).

Replace $\mathbf{A}_0$ by $-\mathbf{A}_0$ in the proof presented above. ■

Proof of Corollary 1 (i).

For μ_2 and $\mathbf{D} \in \mathcal{D}$, the smallest value of $B(\mathbf{D})$ in inequality (6) is $B_2(\mathbf{D}^*) = F(\varepsilon_1^*, \varepsilon_2^*, \mathbf{P}^*)$, with $\mathbf{D}^* = (\mathbf{P}^*)^{1/2}$, as resulting from Theorem 2 (i) part (c). ■

Proof of Corollary 1 (ii).

It is identical with the proof of Corollary 1 (i). ■

Policy search reinforcement learning for automatic wet clutch engagement

Matteo Gagliolo, Kevin Van Vaerenbergh, Abdel Rodríguez, Ann Nowé
CoMo, VUB (Vrije Universiteit Brussel), Pleinlaan 2, 1050 Brussels, Belgium

mgagliolo@vub.ac.be

Stijn Goossens, Gregory Pinte, Wim Symens

FMTc (Flanders' Mechatronics Technology Centre), Celestijnenlaan 300D, 3001 Heverlee, Belgium

stijn.goossens@fmtc.be

Abstract—In most existing motion control algorithms, a reference trajectory is tracked, based on a continuous measurement of the system's response. In many industrial applications, however, it is either not possible or too expensive to install sensors which measure the system's output over the complete stroke: instead, the motion can only be detected at certain discrete positions. The control objective in these systems is often not to track a complete trajectory accurately, but rather to achieve a given state at the sensor locations (e.g. to pass by the sensor at a given time, or with a given speed). Model-based control strategies are not suited for the control of these systems, due to the lack of sensor data. We are currently investigating the potential of a non-model-based learning strategy, Reinforcement Learning (RL), in dealing with this kind of discrete sensor information. Here, we describe ongoing experiments with a wet clutch, which has to be engaged smoothly yet quickly, without any feedback on piston position.

I. INTRODUCTION

RL problems [1] are a class of machine learning problems where an agent must learn to interact with an unknown environment, using a “trial and error” approach. At a given timestep t , the agent may execute one of a set of *actions* $a \in \mathcal{A}$, possibly causing the environment to change its *state* $s \in \mathcal{S}$, and generate a (scalar) *reward* $r \in \mathbb{R}$. Both state and action spaces can be multidimensional, continuous or discrete. An agent's behavior is represented by its *policy*, mapping states to actions. The aim of a RL algorithm is to optimize the policy, maximizing the reward accumulated by the agent. Simply stated, RL consists in learning from a teacher (the environment) who cannot tell us *what* to do next (the optimal policy), but only *how good* we are doing so far (the reward signal). It therefore offers a suitable tool for controlling systems with discrete sensor information. The target state at the discrete sensors location can be incorporated in the reward signal, favoring the desired behavior.

Two main families of RL approaches can be distinguished. In *value based* methods, the expected future reward $Q(s, a)$ allowed by taking action a in state s is estimated: the policy consists in selecting the action a which maximizes Q in the current state s . This can be done storing and updating estimates in tabular form, or resorting to function approximators if $\mathcal{A}$ and $\mathcal{S}$ are too big, or continuous. In *direct policy search*, the space of policies is searched for directly, maximizing the reward.

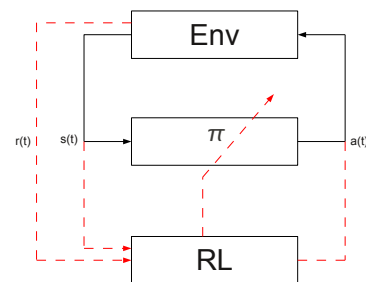


Fig. 1: Block representation of RL. Env: environment (plant being controlled). π : policy (controller), generating the action (control signal) $a(t)$ given the current state of the plant $s(t)$; RL: learning algorithm, adapting the policy based on state, action, and reward signals. Continuous lines indicate online interactions, while discontinuous lines refer to the learning process, which can be online or offline, depending on the particular algorithm.

In both cases, learning is organized in a sequence of *epochs*, each consisting of a sequence of interactions with the environment. Another independent classification can be made among *offline* learning, where the policy is updated only after one or more epochs have been completed; and *online* learning, where such update can be performed also during an epoch (Fig. 1).

II. EXPERIMENTAL SETUP

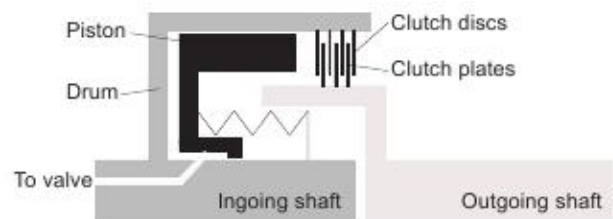


Fig. 2: Schematic representation of the wet clutch

The setup consists of a wet clutch (Figure 2), used in heavy duty transmission systems to engage/disengage differ-

ent loads. We considered in particular the automatic control of the engagement phase, which has to be fast enough, but also smooth, in order to avoid disturbing the operator. This latter requirement can be implemented minimizing the *jerk*, defined as the second derivative of the slip, which is the difference in angular speed among the two shafts.

The issue with this setup is that there is no sensor providing information about the actual position of the piston, which can only be detected when the piston actually touches the plates, and the slip phase begins. Given the available signals (oil pressure and temperature), there is no reference trajectory that can be tracked using classical control methods. In this sense, this setup is a good example of the discrete sensor systems mentioned above (see also [2]). Moreover, there currently is no reliable model of the whole engagement: an approximate model is available for the filling phase, until the piston touches the plates, but not for the following slip phase, which would allow to simulate and optimize the resulting jerk. However, it has been observed that the jerk depends on the speed of the piston when this reaches the plates: the lower the speed, the lower the jerk, and the smoother the engagement. In other words, there is a trade-off among the two objectives: on one hand, a very slow piston movement will result in a smooth engagement, which takes a long time; on the other hand, a fast piston will engage in a short time, but it will also cause a jerky engagement, and possibly damage the setup.

While the availability of a model is not in principle necessary for RL, it can in practice allow us to speed up the learning process, adopting an incremental approach. In a first phase, we can optimize the policy on the filling phase model, aiming at obtaining a fast engagement, but with a low piston velocity at engagement. In a second phase, we will transfer the obtained policy on the real clutch, and further improve its performance, reducing the actual jerk.

III. METHODS

Three *direct policy search* RL methods are being evaluated on this setup, all representing the policy as a probability density function (pdf) over actions, which is updated offline, after each epoch: the policy gradient (PG) method [3], where the actions are drawn from a Gaussian distribution; PG with parameter exploration (PGPE) [4], where the pdf is a mixture of Gaussians; and a continuous learning automaton (CARLA) [5], where the pdf over the actions is non-parametric. In the following three subsections, we briefly describe each method.

A. Policy gradients (PG)

In Policy Gradients (PG) methods [3], the policy is represented as a parametric probability distribution over the action space, conditioned by the current state of the environment. Epochs are subdivided into discrete time steps: at every step, an action is randomly drawn from the distribution, conditioned by the current state, and executed on the environment, which updates its state accordingly. After an epoch has been completed, the parameters of the policy are updated,

following a Monte Carlo estimate of the expected cumulative (discounted) reward. In this section we describe the basic ideas behind this estimation, following [6], to which the reader can refer for further details.

To avoid cluttering notation, we consider a RL problem with a scalar action $a \in \mathcal{A} \subset \mathbb{R}$, and scalar state $s \in \mathcal{S} \subset \mathbb{R}$. Be $\theta \in \mathbb{R}^d$ the d -dimensional parameter defining the conditional distribution over actions $\pi_\theta(a | s)$, from which actions are drawn, and $r_t \in \mathbb{R}$ the reward obtained at timestep $t \in \mathbb{N}$. The index we intend to maximize is the cumulative discounted reward R_0 , with $R_t = \sum_{k=t}^T \gamma^{t-k} r_k$ being the *return* at time t , where $\gamma \in (0, 1)$ is a discount factor, expressing a preference for early over late rewards, and T is the duration of an epoch. Be $H = \{s_0, a_0, \dots, s_T, a_T\}$ the full history of state-action pairs, $H \in \mathcal{H}$, and $R(H)$ the corresponding return. The probability of observing a given history H depends on θ , on the unknown state transition probability $p(s_{t+1} | s_t, a_t)$, and on the unknown initial state probability $p(s_0)$, as

$$p(H) = p(s_0) \prod_{t=0}^T p(s_{t+1} | s_t, a_t) \pi_\theta(a_t | s_t). \quad (1)$$

The expected value of R is defined as

$$J = E\{R(H)\} = \int_{\mathcal{H}} p(H) R(H) dH. \quad (2)$$

The basic idea behind policy gradients is to follow the gradient of J with respect to θ , in order to maximize the expected return. While such gradient cannot be evaluated in closed form, it can be estimated as follows. Bringing the derivative operator inside the integral, we can write:

$$\nabla_\theta J = \int_{\mathcal{H}} [\nabla_\theta p(H) R(H) + p(H) \nabla_\theta R(H)] dH. \quad (3)$$

As the observed rewards do not depend on θ , but only on H , we have $\nabla_\theta R(H) = 0$, therefore the second term of the addition can be discarded. The first term can be rewritten using the “likelihood ratio” trick, multiplying and dividing it by $p(H)$, thus obtaining

$$\begin{aligned} \nabla_\theta J &= \int_{\mathcal{H}} p(H) \nabla_\theta \log p(H) R(H) dH = \\ &= E\{\nabla_\theta \log p(H) R(H)\}. \end{aligned} \quad (4)$$

Given that (1) is a product, its logarithm can be expressed as a sum

$$\begin{aligned} \log p(H) &= \log p(s_0) + \sum_{t=0}^T \log p(s_{t+1} | s_t, a_t) + \\ &+ \sum_{t=0}^T \log \pi_\theta(a_t | s_t), \end{aligned} \quad (5)$$

with the advantage that deriving w.r.t. θ leaves only the rightmost terms, which can be evaluated analytically as π is derivable. We can therefore perform a Monte Carlo estimate

of (4), after running N epochs, as follows

$$\begin{aligned}\nabla_{\theta} J &= E\{\nabla_{\theta} \log p(H) R(H)\} \approx \\ &\approx \frac{1}{N} \sum_{n=1}^N \sum_{t=0}^{T^n} \nabla_{\theta} \log \pi_{\theta}(a_t^n | s_t^n) R(H^n),\end{aligned}\quad (6)$$

where n is used as an index, to refer to the n -th epoch.

In practice, most authors suggest to use $N = 1$, updating θ after each epoch [3], [6]. Moreover, to reduce the variance of the estimate, the term $R(H)$ is replaced by T returns R_t , each multiplied with the corresponding term $\log \pi_{\theta}(a_t | s_t)$, thus discarding past rewards; and shifted by a baseline b , which, in the simplest case, is just the average return observed so far:

$$\nabla_{\theta} J \approx \sum_{t=0}^T \nabla_{\theta} \log \pi_{\theta}(a_t | s_t) (R_t - b). \quad (7)$$

Intuitively, following the gradient (7), the probability of taking action a_t in state s_t is increased if the return R_t , accumulated starting at time t , is larger than the baseline b , representing the average return, while it is decreased otherwise. Fig. 3 illustrates the effect of PG in a simple case, with no state information, $T = 1$, and Gaussian π , with $\theta = (\mu, \sigma)$.

B. PG with parameter exploration (PGPE)

A major disadvantage of PG methods is that drawing a random action at every timestep may result in noisy control signals, as well as noisy gradient estimates. Moreover, the policy is required to be differentiable w.r.t. its parameters. To overcome these issues, PG with Parameter Exploration (PGPE) was introduced [7], [4]. In this method, the random sampling and policy evaluation steps are, in a sense, “inverted”: the policy is a parametric function, not necessarily differentiable, therefore it can be an arbitrary parametric controller; the parameter value to be used is sampled at the beginning of each epoch from a Gaussian distribution, whose parameters are in turn updated at the end of the epoch, again following a Monte Carlo estimate of the gradient of the expected return. In other words, rather than searching the parametric policy space directly, PGPE performs a search in a “meta-parameter” space, whose points correspond to probability distributions over the (parametric) policy space.

Be again θ the parameter of the policy f , where $a_t = f_{\theta}(s_t)$; and α the meta-parameter defining the distribution $p_{\alpha}(\theta)$ over parameter values. With passages analogous to PG, we can again estimate the gradient of the expected return, this time as

$$\nabla_{\alpha} J \approx \frac{1}{N} \sum_{n=1}^N \nabla_{\alpha} \log p_{\alpha}(\theta^n) (R_0^n - b), \quad (8)$$

where θ^n is the parameter used at the n -th of the N epochs considered; also in this case, we can use $N = 1$. Compared with PG, in PGPE the information about the state-action pairs actually visited, conveyed by the history H , is lost, and θ is evaluated directly based on the return R_0 obtained for the whole epoch. In the terminology of [8], PGPE

should therefore be considered a *phylogenetic* method, as evolutionary computation, while PG is *ontogenetic*.

A multimodal PGPE was proposed in [9]: in this case, the distribution is a mixture of Gaussians,

$$p_{\alpha}(\theta^i) = \sum_{k=1}^K w_k^i \mathcal{N}(\mu_k^i, \sigma_k^i), \quad (9)$$

where K is the number of modes, $i = 1, \dots, d$ denotes one of the d dimensions of the parameter, and the weights w_k^i are updated together with the parameters of each Gaussian, following an approximated gradient, with the additional constraint that $\sum_k w_k^i = 1$.

C. Continuous action reinforcement learning automata (CARLA)

Learning Automata [10] are simple reinforcement learners for single-stage, stateless environments, and discrete actions. In this kind of RL problems, which corresponds to the well-known multi-armed bandit problem [1], an epoch consists in the execution of a single action, followed by the observation of the corresponding reward. Also in this case, the policy is represented as a probability distribution over actions, which is updated after each epoch. An extensions to continuous actions, similar to PG, has been proposed in [11]; here, we adopt instead the alternative formulation of [12], termed Continuous Actions Reinforcement Learning Automata (CARLA). In this method, the distribution $f(a)$ over action space is non-parametric: its density, initially uniform, is stored explicitly for all points of an equally spaced grid. After each epoch, the probability of the action executed is reinforced, mixing the current density with a Gaussian function with constant standard deviation λ (termed *spreading rate*), centered in the action itself, whose amplitude is proportional to the observed reward:

$$f_{t+1}(a) = \gamma_t [f_t(a) + \beta_t(a_t) \alpha e^{-\frac{1}{2}(\frac{a-a_t}{\lambda})^2}] \quad \forall a \in \mathcal{A}. \quad (10)$$

Figure 4 illustrates the learning mechanism of CARLA.

IV. APPLICATION TO SETUP

In this section we describe the design decisions taken to apply RL to the wet clutch setup.

State and Action Spaces The clutch is controlled by a single continuous signal, the current to the valve controlling oil pressure. In the model, the available signals are oil pressure and estimated piston position. Of these, only oil pressure is available on the real clutch, therefore we cannot make use of the estimated piston position as state variable. Initially, we will not consider any state information, and control the clutch in open loop. The utility of adding oil pressure is questionable, as this signal tends to follow the input signal. When moving to the real setup, we will include the slip signal.

Policy For the policy, we are considering an open loop controller, defined by four parameters, the same one in the evolutionary computation approach of [13] (Figure 5); a simplified two-parameter version of the same signal, obtained

by fixing two of the four parameters; and a closed loop controller, consisting of a recurrent neural network, along the lines of [6]. In both cases, the RL problem can be represented as a stateless, single stage problem, where an epoch lasts for a single timestep, at which the policy, represented by a multidimensional parameter which corresponds to the action of the reinforcement learner, is applied to the system for a given time. For PG, a multi-stage formulation is possible.

Reward Regarding the reward signal, this has to be a scalar, favoring both objectives at once. For the simulated model, we adopted a reward of the form $r = c/(v + c)$, where v is piston velocity at engagement, $c = 10^{-3}$ is a constant. Attributing this reward at engagement time t results in a return $R = \gamma^t c/(v + c)$, where $\gamma \in [0, 1)$ is a constant discount factor. Note that the term γ^t , typically used in RL to discount rewards, favors early engagements; while the term $c/(v+c)$ tends to 0 for large velocities, is 0.5 for velocity $v = c$, and it tends to 1 when v tends to 0. In practice, γ can be used to control the trade-off among the two objectives: a low value favors early engagements, while a high value increases the impact of the second term, favoring smooth engagements (low piston velocity). Figure 6 plots the return, along with the evolution of related signals in the simulated model. When moving to the real clutch, we will instead minimize the jerk g , again using a reward of the form $\text{const}/(g + \text{const})$.

VI. EXPERIMENTS

In this section we report example plots from experiments with Multimodal PGPE, with 2 and 3 modes, using the two parameter version of the open loop signal in Fig. 5. A learning rate of 0.4 was used. After preliminary tests with higher values, the discount rate γ was set to an intermediate value, 0.5, in order to obtain a good compromise among the two objectives, favoring a smooth yet prompt engagement. Each experiment was repeated 25 times, using different random seeds, each run lasting for 2000 epochs. All runs converged to a different set of parameters (a, d) , but with a similar performance in terms of piston velocity and engagement time. Plotting the final parameters for each of the 25 runs (Figure 7) allows to identify a region of parameter settings which produces a similarly good behavior, corresponding to a “ridge” in the reward function.

To highlight the convergence of the algorithm, in Figures 8, 9 we report instead the evolution of the two objectives, with 2 and 3 modes respectively. Aggregated statistics for all 25 runs are reported in the form of box-plots, each box corresponding to a group of 100 consecutive epochs. From top to bottom, the graphs represent, respectively, the two objectives (piston velocity and engagement time), to be minimized, and the corresponding rewards, to be maximized.

VI. CONCLUSIONS

The first results obtained are promising, as PGPE manages to obtain a smooth engagement in a few hundred epochs. Ongoing research is aimed at analyzing the effect of parameter γ on the trade-off among the two objectives. We will then proceed incrementally, first using the parameters found in

this phase to initialize (a, d) in the four parameter model, and adapting all four parameters further in simulation; then transferring the best obtained controllers to the real clutch, optimizing the actual objective, i.e., reducing jerk.

VII. ACKNOWLEDGMENTS

This work has been carried out within the framework of the LeCoPro project (grant nr. 80032) of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen).

REFERENCES

- [1] R. Sutton and A. Barto, *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press, 1998.
- [2] S. Goossens, G. Pinte, W. Symens, M. Gagliolo, A. Rodriguez, and A. Nowé, “Reinforcement learning for repetitive systems with discrete sensors,” in *30th Benelux Meeting on Systems and Control — Book of Abstracts*. Gent, Belgium: Universiteit Gent, 2011, p. 149.
- [3] J. Peters and S. Schaal, “Policy gradient methods for robotics,” in *Proceedings of the IEEE Intl. Conf. on Intelligent Robotics Systems (IROS)*, 2006.
- [4] F. Sehnke, C. Osendorfer, T. Rückstieß, A. Graves, J. Peters, and J. Schmidhuber, “Parameter-exploring policy gradients,” *Neural Networks*, vol. 23, no. 4, pp. 551–559, 2010.
- [5] A. Rodríguez, R. Grau, and A. Nowé, “Continuous actions reinforcement learning automata. performance and convergence,” in *Intl. Conf. on Agents and Artificial Intelligence (ICAART)*, 2011.
- [6] D. Wierstra, A. Frster, J. Peters, and J. Schmidhuber, “Recurrent policy gradients,” *Logic Journal of IGPL*, vol. 18, no. 5, pp. 620–634, 2010.
- [7] F. Sehnke, C. Osendorfer, T. Rückstieß, A. Graves, J. Peters, and J. Schmidhuber, “Policy gradients with Parameter-Based exploration for control,” in *ICANN '08: Proceedings of the 18th international conference on Artificial Neural Networks, Part I*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 387–396.
- [8] J. Togelius, T. Schaul, D. Wierstra, C. Igel, F. Gomez, and J. Schmidhuber, “Ontogenetic and phylogenetic reinforcement learning,” *Künstliche Intelligenz*, vol. 23, no. 3, pp. 30–33, 2009.
- [9] F. Sehnke, A. Graves, C. Osendorfer, and J. Schmidhuber, “Multimodal Parameter-exploring Policy Gradients,” in *Intl. Conf. on Machine Learning and Applications*. IEEE, 2010, pp. 113–118.
- [10] K. S. Narendra and M. A. L. Thathachar, “Learning automata - a survey,” *IEEE Transactions On Systems, Man And Cybernetics*, vol. SMC-4, no. 4, pp. 323–334, 1974.
- [11] M. A. L. Thathachar and P. S. Sastry, *Networks of Learning Automata: Techniques for Online Stochastic Optimization*. Kluwer Academic Publishers, 2004.
- [12] M. N. Howell, G. P. Frost, T. J. Gordon, and Q. H. Wu, “Continuous action reinforcement learning applied to vehicle suspension control,” *Mechatronics*, vol. 7, no. 3, pp. 263 – 276, 1997.
- [13] Y. Zhong, B. Wyns, R. D. Keyser, G. Pinte, and J. Stoev, “Implementation of genetic based learning classifier system on wet clutch system,” in *30th Benelux Meeting on Systems and Control — Book of Abstracts*. Gent, Belgium: Universiteit Gent, 2011, p. 124.

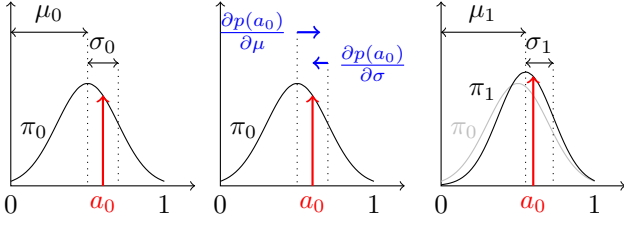


Fig. 3: A simple example illustrating the effect of one step of PG, with no state information, single stage epochs ($T = 1$), $\mathcal{A} = [0, 1]$, and Gaussian π , with $\theta = (\mu, \sigma)$. *Left*: the first epoch is executed, drawing action $a_0 \sim \pi_0(a)$, and observing reward r_0 , which in this case coincides with the return R_0 . *Center*: as $R_0 > b$, following the gradient (7) increases $\pi(a_0)$. *Right*: updated distribution π_1 , ready for the next epoch.

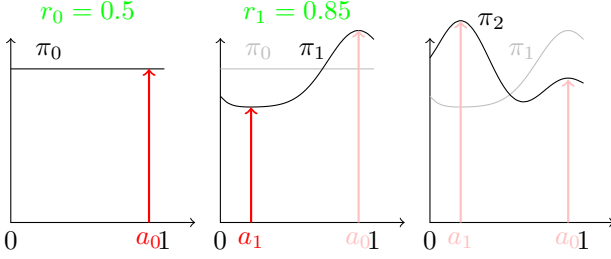


Fig. 4: An example illustrating CARLA. *Left*: the first epoch is executed, drawing action $a_0 \sim \pi_0(a)$, where π is initially uniform, and observing reward r_0 . *Center*: π is updated, mixing it with a Gaussian bell centered in a_0 , with amplitude proportional to r_0 as $R_0 > b$. An action a_1 is drawn from the updated π , resulting in a larger reward r_1 . *Right*: π is updated, this time mixing with a larger bell, as the observed reward r_1 was bigger.

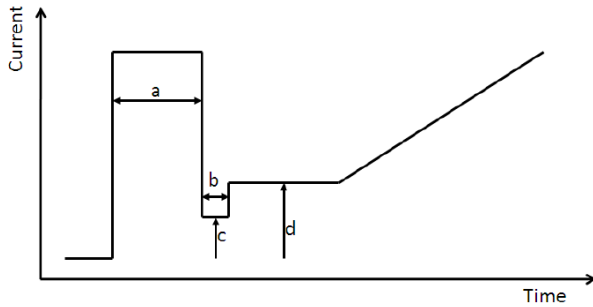


Fig. 5: Parametric input signal from [13], with four parameters (a, b, c, d). In our implementation, all parameter ranges are mapped to the unit interval $[0, 1]$. In the two parameter version, only a and d are free, while $b = 0$, and $c = d$.

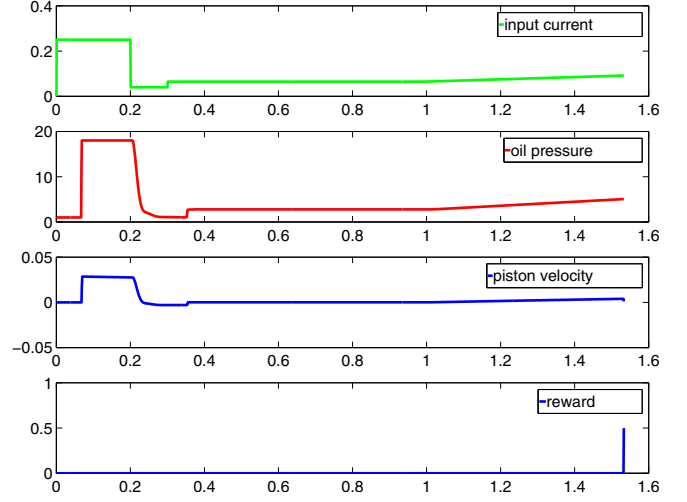


Fig. 6: A capture of the different signals available in the filling phase model of the wet clutch. Horizontal axes report time in seconds. From top to bottom: an example input signal (with $a = b = c = d = 0.5$); oil pressure; estimated piston position; reward signal, consisting of a single peak at engagement time, just above 1 second.

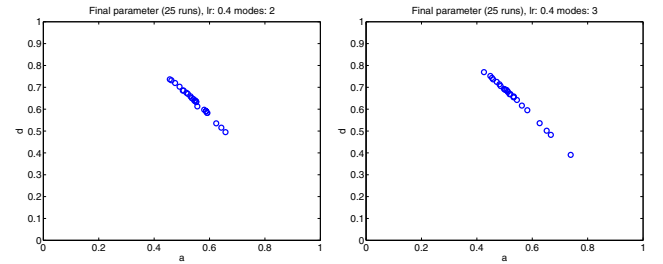


Fig. 7: PGPE with 2 (left) and 3 modes (right): Final parameter values for 25 runs, identifying a region of high reward.

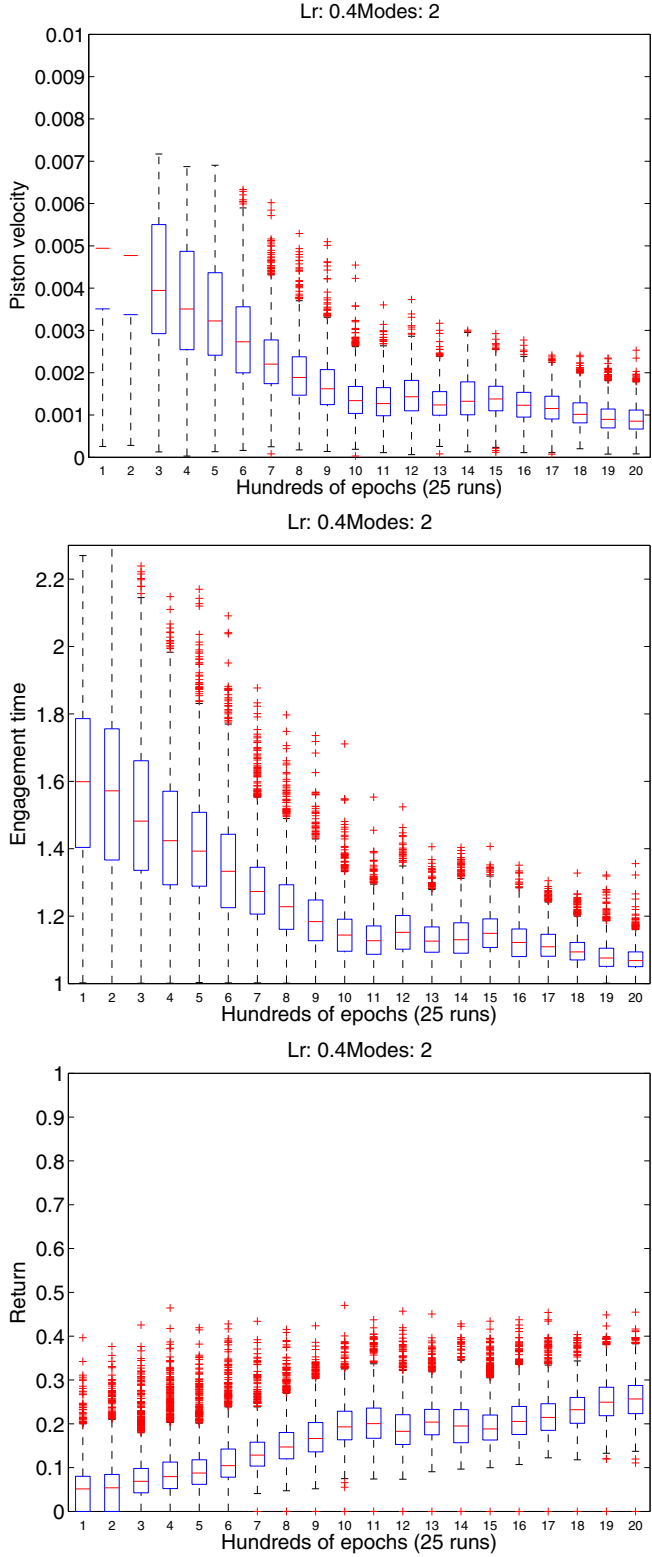


Fig. 8: PGPE with 2 modes: Performance for 25 runs. Each box corresponds to 100 consecutive epochs. Top: piston velocity, to be minimized. Center: engagement time, to be minimized. Bottom: reward, to be maximized.

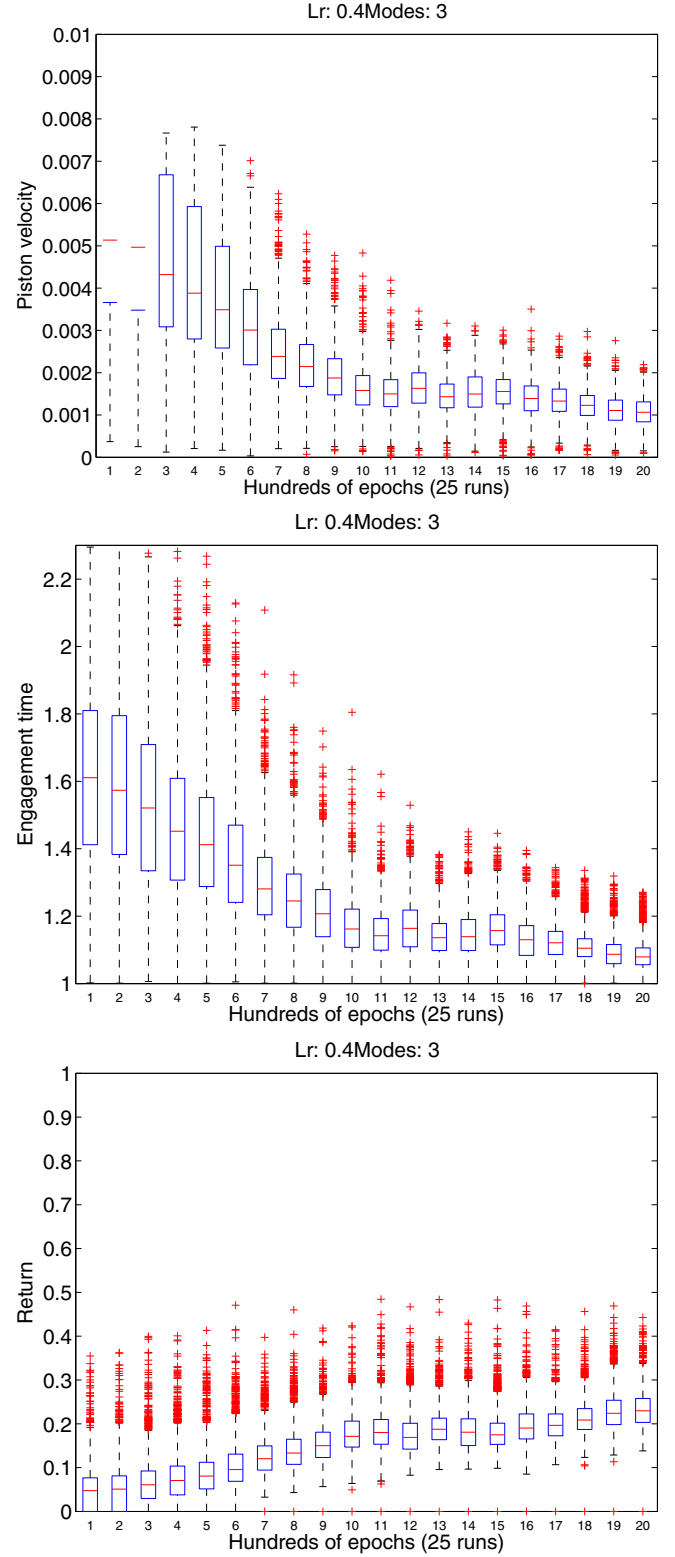


Fig. 9: PGPE with 3 modes: Performance for 25 runs, divided in bins of 100 epochs each. Top: piston velocity, to be minimized. Center: engagement time, to be minimized. Bottom: reward, to be maximized.

Gradient-based Classification and Representation of Features from Volume Data

Marius Gavrilescu, Vasile Manta, Eduard Gröller

Abstract— The extraction and representation of information from volume data are important research avenues in computer-based visualization. The interpretation of three- or multi-dimensional data from various scanning devices is important to medical imaging, diagnosis and treatment, reliability and sustainability analyses in various industrial branches, and, in more general terms, information visualization. In this paper, we present several approaches for the classification and representation of relevant information from volume data sets. The techniques are based on the gradient vector, a property directly derived from the original volume data. We show how this property can be computed and subsequently used for classification through gradient-based one- and multi-dimensional transfer functions, as well as for the enhancement of surface features. The described techniques are illustrated through images generated using our volume rendering framework, from Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) data sets. The resulting images show how gradient-based techniques are suited for improved volume classification and the better extraction of meaningful information.

I. INTRODUCTION

Volume visualization is a sub-domain of computer graphics which deals with the exploration, sampling, classification and representation of volume data. Such data most commonly originates from scanning devices based on Computed Tomography (CT), Magnetic Resonance Imaging (MRI), ultrasound etc. When an object is scanned, the result is a volume data set which contains information on the physical properties of the media inside a rectangular region of space around the object. The data is most often arranged in a regular grid of constituents referred to as

voxels [1]. These are the three-dimensional (3D) equivalents of pixels from two-dimensional (2D) images. However, they do not contain optical information (color and opacity), but rather have an associated scalar value which quantifies the reaction of the corresponding medium to the scanning device. This scalar value is referred to as *density* or *intensity*, depending on the type of device used. For example, in the case of medical CT images, the scalar values constitute an accurate mapping of the densities of various tissues [2]. Bone, for instance, has values from a higher range than muscle or blood vessels. This difference in associated scalar values is a powerful criterion for the classification of volume data sets, as it allows for the separation of various types of materials based on their differing densities. However, as we will explain later in the paper, density alone is sometimes not sufficient for proper classification, and thus other voxel properties have to be obtained and factored into the classification method.

Throughout the visualization process, volume data sets are subjected to various computational stages which are part of the *volume rendering pipeline* [3, 4]. Rendering is the process through which images are produced from abstract data or geometry. In the case of volume rendering, this pipeline involves loading the data into memory, reconstruction of the volume from the data set, sampling, classification, and various other techniques and enhancements. The result is an image which is produced by projecting the 3D data onto the 2D viewing plane [5]. The classification stage is arguably among the most important ones, since it is here that the information of interest is isolated from the rest [6]. In this paper we focus on the type of classification which makes use of the gradient vector to discriminate among voxels. In the following sections we explain what the gradient vector is, how it can be computed, and how it can be used for the purposes of volume visualization.

Section II of the paper briefly describes the volume rendering process used in our framework. Section III presents several approaches used to compute the gradient vector and illustrates their advantages and shortcomings. Section IV presents techniques for gradient-based classification, applied to CT and MRI medical data sets. The final section concludes the paper and details future extensions of the described methods.

Manuscript submitted on 18.06.2011 This work was supported in part by the "BRAIN - An Investment in Intelligence" doctoral Scholarship Program, within the Technical University of Iasi, Romania. This paper was supported by the CNCSIS project IDEI 342/2008, *Nonlinear evolution equations - theoretical aspects and applications to life and environmental sciences*.

Marius Gavrilescu is a PhD student from the Technical University of Iasi, Romania, currently working at the Vienna University of Technology, Austria (e-mail: mariusgv at cg.tuwien.ac.at, mariusgav42 at yahoo.com).

Vasile Manta is professor at the Technical University of Iasi, Romania, Faculty of Automatic Engineering and Computer Control (e-mail: vmanta at cs.tuiasi.ro).

Eduard Gröller is associate professor at the Vienna University of Technology, Austria and adjunct professor of computer science at the University of Bergen, Norway (e-mail: groeller at cg.tuwien.ac.at).

II. VOLUME RECONSTRUCTION, SAMPLING AND RENDERING

This section succinctly explains the methods used by our framework for processing and representing the volume data. For readers not directly associated with volume graphics or computer graphics in general, we feel that these explanations are necessary in order to make other sections easier to follow.

The initially discrete volume data is stored in video memory as a 3D texture [7]. This allows the reconstruction of the continuous volume via hardware-implemented trilinear interpolation [8]. The volume is now continuous and described by a function $V: R^3 \rightarrow R$, which maps scalar values to each position (x, y, z) . After reconstruction, scalar values are retrievable via texture look-ups from any such position within the volume. The distribution of the scalar values depends on the physical properties of the materials of the scanned object. Denser materials such as bone or metal have different scalar values from lighter ones such as soft tissues, plastic or various fluids.

The scalar values are the basis for retrieving information from the data. The aim is to extract the desired information from the volume while hiding unwanted data, and to generate an image which illustrates this information.

The rendering approach used by our framework is called Direct Volume Rendering (DVR) [9, 10]. Essentially, it involves sampling and displaying the volume without employing additional geometry or triangle-based surfaces to extract the desired information. As the name suggests, DVR methods work directly on the data points. The algorithm used for sampling and projecting the data is called *ray casting* [7, 11]. The concept behind this approach is illustrated in Fig. 1.

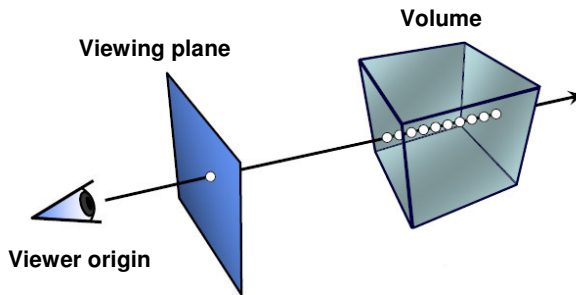


Fig. 1. Sampling of a volume data set through ray casting. A single ray is illustrated, which is emitted from the viewing plane into the box-shaped volume. Inside the volume, the white dots are positions sampled along the ray

Ray casting is a popular algorithm with multiple implementations in volume visualization [12, 13, 14]. The main steps for basic ray casting are as follows:

- for each pixel, emit a ray from the viewing plane through the volume
- sample the volume along the rays (as illustrated in Fig.1)
- assign opacity and color to each sampled point using some classification method
- combine the colors and opacities of the points along each ray (the result is the color of the pixel from which the ray was originally emitted)

The colors of the resulting pixels are combinations of the colors of the sampled points along each respective ray, weighted by the opacities of each point. Less transparent points contribute more color to their corresponding pixel than more transparent ones. Through classification, the distribution of color and transparency is controlled throughout the volume. Regions which contain uninteresting data are typically set to be completely transparent, while useful information is semi-transparent or opaque, and highlighted by a proper combination of color and opacity.

Some of the more frequently used methods for classification are collectively referred to as *transfer functions*. A transfer function maps color and opacity to sampled points in the volume depending on the properties of voxels. The most straightforward of such properties is the scalar value itself, which leads to a category of classification methods called *data-based transfer functions*. Such functions assign color and opacity based on the scalar value. These optical properties are typically RGBA values; RGB is the Red-Green-Blue color space, while A is called *alpha value* and controls opacity. Therefore, transfer functions are $T_f: R \rightarrow R^4$, where $T_f(V) = \text{RGBA}$. These are 1D functions which assign an RGBA quadruplet to each value of the volume function V described earlier. Through careful specification of transfer functions, certain regions in the volume can be emphasized, while others are hidden. Throughout the paper, we define the shape of these functions through user-interactive tools [15, 16, 17], which allows great flexibility for the fine-tuning of the rendered images. In certain cases however, taking only the scalar value into account is not enough, and other voxel properties are needed for a more refined classification. In the following section, we describe such a property and explain its usefulness for the purpose of visualization.

III. THE GRADIENT VECTOR

Considering the function V mentioned in Section II, the gradient vector (or *gradient*, for short), is its first derivative. Since V is a three dimensional function, the gradient ∇V is a three-component vector containing the partial derivatives of V in directions, x , y , and z .

The most useful property of the gradient is that its orientation locally indicates the largest degree of change

in scalar values, which can indicate a significant surface inside the volume [18]. For instance, the gradient at the interface between bone and muscle tissue is oriented perpendicularly to this interface, and this orientation can be used to identify the surface of the bone. One important application of gradient vectors is the implementation of a local illumination model. Lighting adds a significant degree of realism and quality to a rendered image, and it is featured in nearly every figure in this paper. Local illumination relies on the normal vector, which indicates the local orientation of a surface. The normal vector from triangle- and quad- based graphics can be substituted by the gradient, since, as mentioned, it is perpendicular to significant identifiable surfaces in the volume [19].

There exists a variety of techniques for the numerical estimation of the gradient from discrete data. There are basically two main directions when computing the gradient: pre-computation and dynamic computation.

The idea when pre-computing gradients is to calculate them in a pre-processing step, before any actual rendering takes place. The result is a three- or four-component volume which stores for each voxel the corresponding gradient as three components and, if desired, its magnitude as the fourth component. In the paper, we use the *central differences* approach for gradient estimation. This means that, for a specific voxel, the gradient is computed based on the differences between neighboring voxels in various arrangements. The simplest such arrangement is to use pairs of neighbors from each main axis, located in either direction, for a total of six neighboring voxels. Assuming the discrete volume is stored as a 3D array with components $V_d[i, j, k]$, where i, j, k are indexes used for traversing the array in its three respective dimensions, the gradient is computed as shown in (1).

$$\nabla V = 0.5 \begin{pmatrix} V_d[i+1, j, k] - V_d[i-1, j, k] \\ V_d[i, j+1, k] - V_d[i, j-1, k] \\ V_d[i, j, k+1] - V_d[i, j, k-1] \end{pmatrix} \quad (1)$$

An extension of this method involves using all 27 values from within a 3x3x3 cell around a voxel, i.e., all $V_d[i+x, j+y, k+z]$, where $i_x, j_y, k_z = \{-1, 0, 1\}$. This results in a more precisely estimated gradient and a smoother look of the rendered images [20].

Unlike the pre-computational approach, dynamic gradient estimation is incorporated into the volume rendering pipeline and thus is performed in real-time. The gradients are not pre-stored in memory, but rather are calculated when needed during the rendering process.

Similarly to the first method from Subsection III, six neighbors are initially used, two for each axis. The difference is that the gradient estimator often operates on the continuous, reconstructed volume. Thus, neighboring voxels are not restricted to an indexed matrix, but may be

chosen by sampling at an arbitrary distance d from the center voxel (2). d is set to a value which is small compared to the size of the volume.

$$\nabla V = \frac{1}{2d} \begin{pmatrix} V(x+d, y, z) - V(x-d, y, z) \\ V(x, y+d, z) - V(x, y-d, z) \\ V(x, y, z+d) - V(x, y, z-d) \end{pmatrix} \quad (2)$$

The method can be further improved by considering a multitude of neighbors around the center voxel. We refer to this approach as *spherically-computed gradient*. Specifically, we take samples located on the surface of a sphere centered on the voxel in question. The difference is then computed between each sample on one hemisphere and its diametrically opposed correspondent on the complementary hemisphere. Summing up these differences produces the desired gradient. The positions of the samples are calculated using spherical coordinates. Considering a voxel located at position $vPos = (x, y, z)$ with the associated scalar value $V(vPos)$, the gradient vector is obtained as depicted in (3), assuming n samples are taken at distance r from the voxel.

$$\begin{aligned} pos &= \begin{pmatrix} \cos\left(\frac{i\pi\sqrt{2/n}}{2}\right) \sin(2i\pi\sqrt{2/n}) \\ \sin\left(\frac{i\pi\sqrt{2/n}}{2}\right) \sin(2i\pi\sqrt{2/n}) \\ \cos(2i\pi\sqrt{2/n}) \end{pmatrix} \\ \nabla V &= \sum_{i=0}^{n/2-1} ((V(vPos + r \cdot pos) - V(vPos - r \cdot pos)) \cdot pos) \end{aligned} \quad (3)$$

The effects of the use of gradients as computed using the aforementioned approaches are shown in Fig. 2. We implemented local illumination using the Phong model [21], where the gradient is used in place of the normal. Using more than six neighbors results in smoother images regardless of whether the computation is carried out in a pre-processing step or dynamically. The advantages and drawbacks of both approaches stem from the often-occurring trade-off between memory and computational power requirements. Pre-computed gradients require as much as four times more memory than the volume data set. For and for large data this may well exceed the available video memory. However, once the gradients are stored, they are obtainable via a single texture fetch, with barely any impact on performance. Dynamic gradients, on the other hand, require barely any memory. Since they are computed on-the-fly and require multiple texture fetches, the impact on performance can be substantial. The spherically-

computed gradients have a particularly negative influence on the time required for rendering, but produce better quality images. A good trade-off is the pre-computed 27 value method, which is sufficient for high-quality images, provided enough memory is available.

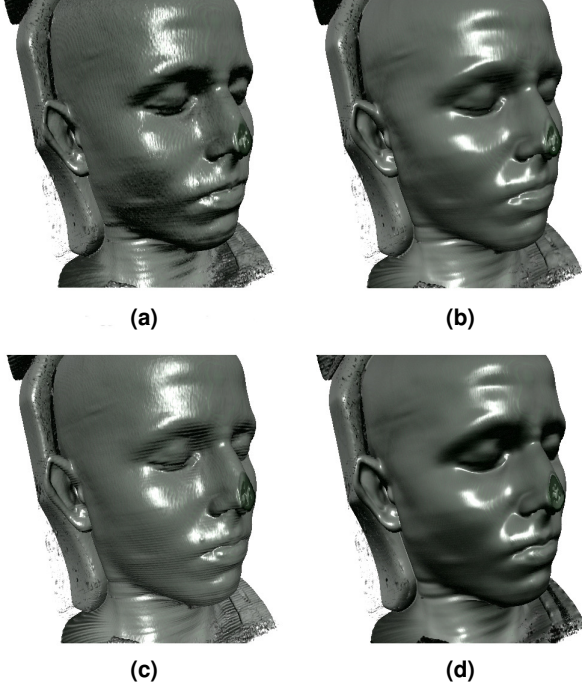


Fig. 2. Illumination of a volume data set using: pre-computed gradient using 6 neighbors (a); precomputed gradient using 27 values (b); dynamically computed gradient using 6 neighbors (c); dynamic spherically-computed gradient (d).

IV. GRADIENT-BASED CLASSIFICATION

For the purposes of volume classification, the most significant application of gradients is the identification and highlighting or elimination of surfaces. The orientation of the gradient indicates the orientation of the surface, while its magnitude indicates the difference between the materials on either side of the surface. We use the gradient magnitude as an additional voxel property (aside from the scalar value) and incorporate it into the classification process. The first classification method involves 1D gradient-based transfer functions. Similarly to data-based transfer functions, this class of functions assigns RGBA values to sampled voxels, and is described by (4).

$$\begin{aligned} T_f : R &\rightarrow R^4, \\ T_f(\|\nabla V\|) &= RGBA \end{aligned} \quad (4)$$

Fig. 3 presents the result of using gradients for classification. Fig.3(a) shows a volume rendered with a 1D data-based transfer function. A 1D gradient-based function is used In Fig.3(b). Various surfaces are clearly identified and rendered semi-transparently, while color is used to separate certain features such as the sinus cavity and the metencephalon. The voxels corresponding to the structures identified in bright green in Fig 3(b) could not be similarly highlighted using a data-based function (Fig. 3(a)) because they share scalar values with neighboring structures which occlude them. Surface information, however, allows for their separation.

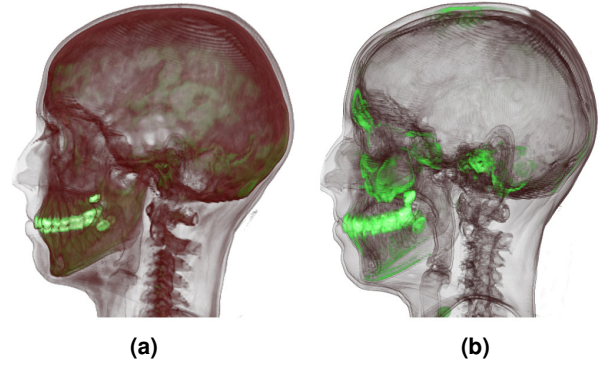


Fig. 3. Classification using: a 1D data-based transfer function (a); a 1D gradient-based transfer function (b).

The identification and highlighting of surfaces is further explored using combinations of data- and gradient-based classification. One approach is to use what we refer to as *thresholded transfer functions*. These are assigned a threshold scalar value to separate data- and gradient-based classification. With thresholded transfer functions, gradient-based classification is applied to voxels whose scalar value is below the threshold, while scalar values are used when they are above this threshold.

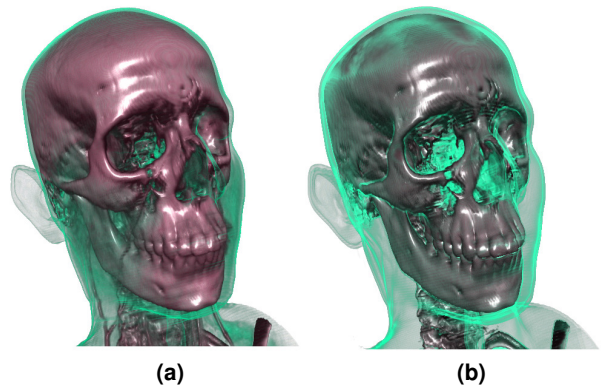


Fig. 4. Surface identification and highlighting using: a 1D data-based transfer function (a); a thresholded transfer function (b)

A result is shown in Fig. 4. The objective is to highlight the outer surface (the skin tissue), while still keeping bone structures visible. In Fig. 4(a), this was attempted using a traditional 1D data-based function. While the outer, transparent surface was correctly identified, it could only be highlighted to a certain extent, before started occluding other structures. Fig. 4(b) shows the same attempt, only this time a thresholded transfer function was used. Bone tissue was still represented from scalar values, but skin was identified using gradient information. The result is that the skin surface can be better highlighted, while still keeping the bone underneath visible.

The flexibility of the classification is further extended into the realm of 2D transfer functions [22]. For an arbitrary voxel, an RGBA quadruplet is generated considering both its properties, the scalar value and the gradient magnitude (5).

$$\begin{aligned} T_{f2} : R^2 &\rightarrow R^4, \\ T_{f2}(V, \|\nabla V\|) &= RGBA \end{aligned} \quad (5)$$

Two dimensional transfer functions have two independently-generated components: one is data-based (T_{fd}) and the other one is gradient-based (T_{fg}). The 2D function is formed from combinations of these two components, thereby incorporating both voxel properties into the classification process (6).

$$T_{f2} = w_d T_{fd} + w_g T_{fg} \quad (6)$$

Through a careful specification of the two components, as well as a fine-tuning of the w_d and w_g weights, this form of classification proves more flexible than its one-dimensional counterpart.

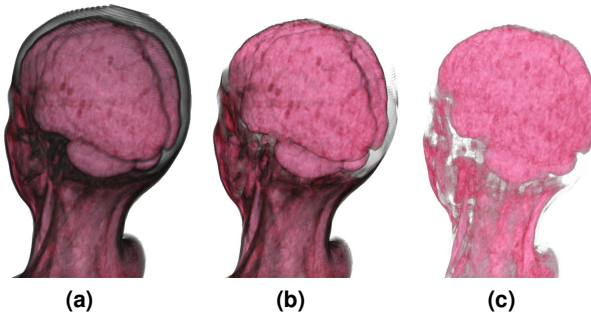


Fig. 5. Visualization of brain tissue from a CT dataset using a 2D transfer function; the layer of skin from (a) is partially removed (b) and completely removed (c).

Fig. 5 shows the application of a 2D transfer function for a CT dataset. The aim is to show soft brain tissue underneath similarly dense skin. These two tissues are inseparable based on scalar values only. The brain and skin are separately identified in Fig. 5(a). Through the manipulation of the aforementioned weights, the opacities associated with certain gradient values can be lowered.

This is reflected in Fig. 5(b), where it can be seen how the outer surface starts to lose visibility and “peel away”. The brain beneath it is still visible and largely unaffected. The limit of the technique is shown in Fig. 5(c). The outer surface is completely removed, but some surface information is also lost from the brain region.

We also tried this technique on MRI scans. MRI data is often more problematic than CT, because the scalar values for distinctive materials overlap more frequently and the noise content is substantially higher. However, soft tissue such as brain matter or blood is better identified in MRI images. Unlike the previous CT dataset, the brain can be identified using a 1D data-based transfer function (Fig. 6(a)).

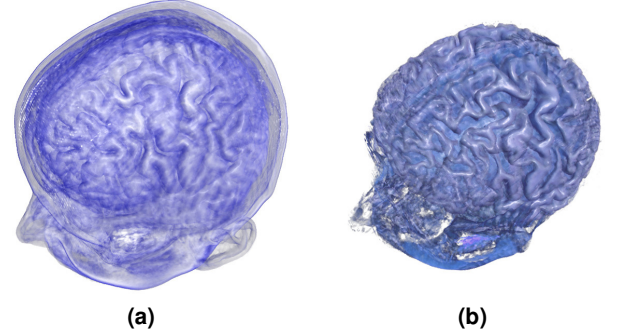


Fig. 6. Visualization of brain tissue from an MRI data set using: a 1D transfer function (a); a curvature-modulated 2D transfer function (b).

As previously, we attempted to remove the outer skin layer, but, due to the noise content and the distribution of scalar values throughout the dataset, the 2D transfer function described by (6) proved insufficient. We therefore have to further refine the classification. The transfer function from (6) is expanded by adding curvature information [23, 24, 25], since the ridges and valleys from the surface of the brain are generally more curved than the outer layer of skin. First, local curvature is estimated as shown in (7). For any vector $vect$ we use the notation $vect_n$ to denote that $vect$ is normalized.

$$k(x, y, z) = \left\| \begin{pmatrix} \nabla V(x+d, y, z)_n \cdot \nabla V(x-d, y, z)_n \\ \nabla V(x, y+d, z)_n \cdot \nabla V(x, y-d, z)_n \\ \nabla V(x, y, z+d)_n \cdot \nabla V(x, y, z-d)_n \end{pmatrix} \right\| \quad (7)$$

The curvature is estimated to be the magnitude of the vector whose components are the dot products of the normalized gradients from each respective axis. Once the curvature is obtained, the transfer function from (6) is modulated accordingly (8). An additional coefficient w_k is also introduced for fine tuning the modulation.

$$T_{f2k} = w_k k (w_d T_{fd} + w_g T_{fg}) \quad (8)$$

The result can be seen in Fig. 6(b), where the outer layer of skin has been removed almost completely.

V. CONCLUSIONS

The paper presented methods for the estimation of the gradient from volume data, and various applications involving classification via transfer functions based partially or completely on the gradient. The rendered images show that incorporating this voxel property into the classification process allows for more classification possibilities. 1D and 2D transfer functions were used for the identification of structures and the highlighting and removal of surfaces, from both CT and MRI data. When 2D functions proved insufficient, curvature information was added in order to refine the classification process. Future work in this direction involves the expansion of the classification possibilities within our rendering framework. We intend to incorporate additional voxel properties, such as occlusion or distance, into the classification pipeline, to allow for even greater flexibility. The aim is to enable the extraction of individual structures within volumes without having to rely on segmentation or costly non-interactive methods.

ACKNOWLEDGMENT

Marius Gavrilescu thanks the members of the Vis-Group at the Institute of Computer Graphics and Algorithms of the Vienna University of Technology for their valuable support and help with research.

REFERENCES

- [1] M. Chen, A. Kaufman, R. Yagel, *Volume Graphics*. Springer, 2000.
- [2] J. Hsieh, *Computed Tomography – Principles, Design, Artifacts and Recent Advances*. Bellingham, WA: The International Society for Optical Engineering, 2003.
- [3] T. Peterka, R. Ross, Y. Hongfeng, K.-L. Ma, W. Kendall, H. Jian, "Assessing improvements to the parallel volume rendering pipeline at large scale", in *Proc. Ultrascale Visualization 2008 Workshop*, Austin TX, 2008.
- [4] J.E. Vollrath, D. Weiskopf, T. Ertl, "A generic software framework for the gpu volume rendering pipeline", in *Proc. Vision, Modelling and Visualization*, Erlangen, Germany, 2005, pp. 391-398.
- [5] Chen M., Kaufman A., Yagel R., *Volume Graphics*. Springer, 2000.
- [6] E.B. Lum, J. Shearer, K.-L. Ma, "Interactive multi-scale exploration for volume classification", *Vis. Comput.*, vol. 22, no. 9, pp. 622-630, 2006.
- [7] M. Hadwiger, J. M. Kniss, C. Rezk-Salama, D. Weiskopf, and K. Engel, *Real-time Volume Graphics*. A K Peters, 2006.
- [8] H. Nguyen, *GPU Gems 3*, Addison-Wesley Professional, 2007.
- [9] G.M. Nicoletti, "Advances in direct volume rendering for visualizing large 3D data sets from scientific and medical applications", in *Proc. 5th Biannual World Automation Congress*, vol. 13, 2002, pp. 245-250.
- [10] V. Solteszova, D. Patel, S. Bruckner, I. Viola, "A Multidirectional Occlusion Shading Model for Direct Volume Rendering", *Comput. Gr. Forum*, vol. 29, no. 3, 2010, pp. 883-891.
- [11] L. Marsalek, A. Hauber, P. Slusallek, "High-speed volume ray casting with CUDA", in *Proc. IEEE Symposium on Interactive Ray Tracing*, Los Angeles, 2008, pp. 185-190.
- [12] M. Weiler, M. Kraus, M. Merz, T. Ertl, "Hardware-based ray casting for tetrahedral meshes," in *Proc. IEEE Visualization 2003*, Seattle, WA, 2003, pp. 333-340.
- [13] H. Ray, H. Pfister, D. Silver, T.A. Cook, "Ray casting architectures for volume visualization", *IEEE Trans. Vis. Comput. Gr.*, vol. 5, no. 3, 1999, pp. 210-219.
- [14] F. Rossler, R.P. Botchen, T. Ertl, "Dynamic shader generation for GPU-based multi-volume ray casting", *Comput. Gr. App.*, vol. 28, no. 5, 2008, pp. 66-74.
- [15] S. Bruckner, E. Gröller, "VolumeShop: An interactive system for direct volume illustration", in *Proc. IEEE Visualization 2005*, Minneapolis, MN, 2005, pp. 671-678.
- [16] J. Meyer-Spradow, T. Ropinski, J. Mensmann, K. Hinrichs, "Voreen: A Rapid-Prototyping Environment for Ray-Casting-Based Volume Visualizations", *IEEE Comput Gr. App.*, vol. 29, no. 6, 2009, pp. 6-13.
- [17] M. Gavrilescu, M.M. Malik, E. Gröller, Custom interface elements for improved parameter control in volume rendering, in *Proc. ICSTC 2010*, Sinaia, Romania, pp. 219-224.
- [18] M. Levoy, "Display of surfaces from volume data", *IEEE Comput. Gr. App.*, vol. 8, no. 3, 1988, pp. 29-37.
- [19] C. Rezk-Salama, M. Hadwiger, T. Ropinski, P. Ljung, "Advanced illumination techniques for GPU volume raycasting", *SIGGRAPH Course Notes*, 2009.
- [20] L. Neumann, B. Csebfalvi, A. König, Eduard Gröller, "Gradient estimation in volume data using 4D linear regression", *Comput. Gr. Forum*, vol 19, no 3, 2000, pp. 351-358.
- [21] T. Whitted, "An improved illumination model for shaded display", *Communications of the ACM*, vol. 23, no. 6, 1980, pp. 343-349.
- [22] J. Kniss, G. Kindlmann, C. Hansen, "Multidimensional transfer functions for interactive volume rendering", *IEEE Trans. Vis. Comput. Gr.*, vol. 8, no. 3, 2002, pp. 270-285.
- [23] J. Hladuvka, A.H. König, E. Gröller, "Curvature-based transfer functions for direct volume rendering", in *Proc. Spring Conference on Computer Graphics 2000*, vol. 16, 2000, pp 58-65.
- [24] G. Kindlmann, R. Whitaker, T. Tasdizen, T. Möller, "Curvature-based transfer functions for direct volume rendering: methods and applications", in *Proc. IEEE Visualization 2003*, Washington DC, 2003, pp. 67-76.
- [25] S. Bruckner, E. Gröller, "Style transfer functions for illustrative volume rendering", *Comput. Gr. Forum*, vol. 26, no. 3, 2007, pp. 715-724.

Singular perturbation approach to RBNN adaptive control of unknown flexible joint manipulators

M. Ghateei, B. Karimi, Gh. Shahgholian and M. Mosayyebi

Abstract— In this paper, a composite control approach to the adaptive RBNN presented for rigid link flexible joint (RLFJ) with unknown parameters. Singular perturbation method used for robot motion dynamics in two subsystems: a slow subsystem and a fast subsystem. Furthermore, simultaneous tracking and suppression of vibration in joint of manipulator is possible by application of the composite controller and the stability of the closed loop system is guaranteed through Lyapunov stability analysis. After proposing the controller, we perform some simulation studies for a rigid link flexible joint manipulator system. The simulation results show the effectiveness of the approach.

I. INTRODUCTION

In the past decades, there has been increased research in the development of nonlinear control theories to the control of manipulators. Many classical controller (PI, PID and some other type of classical controller) schemes were developed for rigid link flexible joint (RLFJ) manipulator control with parametric uncertainties [1], [2]. Also one big problem for RLFJ was that the joint flexibility effects have been neglected. Furthermore, effects of flexibility in joints may cause instability behavior in the system [3]. Therefore, reference [4] proposed a precise model for RLFJs. First of all in [5], authors proposed a nonlinear controller for FJR's. After that nonlinear controllers was developed for control of manipulators. Flexibility in manipulator makes an act with two time-scale behavior and this idea led to used singular perturbation method for manipulators [6]. Since, most of controllers had lack of robustness of parametric uncertainty, the adaptive control was proposed in [7]. Reference [10] proposed an adaptive controller based singular perturbation method with assumption of weak elasticity in joint which requires only position and velocity feedback. The author in [8], proposed some assumptions for change the adaptive rules and used for FJR' that was well-known for RLRJ manipulators. In [19], authors proposed a controller using neural networks for development of RLFJ manipulators, also a lot of authors used this idea in researches [12], [13], [14]. This work presents an adaptive controller for rigid link flexible joint manipulator with parameter uncertainty using radial basis neural networks (RBNN). The stability of the closed-loop system is guaranteed by lyapunov's stability theory and the proposed control scheme makes that all

signals in the closed-loop system are bounded and the tracking error goes asymptotically to zero.

This paper is organized as follows: in section 2, the development of the non-linear dynamic model of the RLFJ manipulator and singular perturbation method are described. Section 3 presents the main results of the adaptive RBNN controller for slow subsystem. Composite control and state-feedback control for fast subsystem demonstrate in section 4. Simulation results are given in section 5. Finally, section 6 concludes the paper.

II. PROBLEM FORMULATION

A. Dynamic Model

The model for an n-link RLFJ robot is given as follows [15]:

$$M(q_f, q_l) \begin{bmatrix} \ddot{q}_f \\ \ddot{q}_l \end{bmatrix} + \begin{bmatrix} f_1(q_f, \dot{q}_f, q_l, \dot{q}_l) \\ f_2(q_f, \dot{q}_f, q_l, \dot{q}_l) \end{bmatrix} + \begin{bmatrix} g_1(q_f, q_l) \\ g_2(q_f, q_l) \end{bmatrix} + \begin{bmatrix} 0 \\ kq_l \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix}, \quad (1)$$

where $q_f, \dot{q}_f, \ddot{q}_f$ referring to the link position, velocity and acceleration, $q_l, \dot{q}_l, \ddot{q}_l$ are the motor shaft angle, angular velocity and angular acceleration, respectively, $M(q_f, q_l) \in \mathbb{R}^{n \times n}$ is the inertia matrix, $f_i \in \mathbb{R}^n$ is coriolis and centripetal term, $g_i \in \mathbb{R}^n$ is the gravity vector, $k = k^T \in \mathbb{R}^n$ is the stiffness coefficients matrix and $\tau \in \mathbb{R}^n$ is the control Torque. Since the inertia matrix M is positive definite, it can be inverted and denoted by T , which can be partitioned as follows [16]:

$$T = M^{-1} = \begin{bmatrix} T_{11}(q_f, q_l) & T_{12}(q_f, q_l) \\ T_{21}(q_f, q_l) & T_{22}(q_f, q_l) \end{bmatrix}. \quad (2)$$

Equation (2) then becomes:

$$\ddot{q}_f = T_{11}(-f_1 - g_1 + \tau) + T_{12}(-f_2 - g_2 - kq_l), \quad (3)$$

$$\ddot{q}_l = T_{21}(-f_1 - g_1 + \tau) + T_{22}(-f_2 - g_2 - kq_l). \quad (4)$$

B. Singular perturbation method:

In this section, the dynamic model of a manipulator with flexible joint is converted into a two-time-scale singular perturbation model. Let k_c be a common scale factor between all the stiffness constants of the joint [17]. The inverse of this scale factor is chosen as a perturbation parameter. ($k_c = \min(k_1, k_2, \dots, k_n)$). With this common scale factor, torsional spring constant k can be scaled by k , such that $\tilde{k} = k / k_c$. Now defining $\mu = 1 / k_c$, the next step is to define $q_l = \mu \sigma$. The dynamic model of the RLFJ obtained and transformed into a two-time-scale singular perturbation

Manuscript received May 15, 2011.

M. Ghateei and Gh. Shahgholian are with the Department of Electrical Engineering, Islamic Azad University, Najafabad Branch, Isfahan, Iran.

B. Karimi and M. Mosayyebi are with the Department of Electrical Engineering, Malek-ashtar University of Technology, Shahinshahr, Isfahan, Iran, (e-mail: bahram-karimi@aut.ac.ir).

model and then $\ddot{q}_f$ and $\ddot{q}_l$ can be determined from equations (3) and (4) as follows:

$$\begin{aligned} \ddot{q}_f = & T_{11}(q_f, \mu\sigma)(-f_1(q_f, \dot{q}_f, \mu\sigma, \mu\ddot{\sigma}) \\ & - g_1(q_f, \mu\sigma) + \tau) + T_{12}(q_f, \mu\sigma) \\ & (-f_2(q_f, \dot{q}_f, \mu\sigma, \mu\ddot{\sigma}) - g_2(q_f, \mu\sigma) - \tilde{k}\sigma), \end{aligned} \quad (5)$$

$$\begin{aligned} \mu\ddot{\sigma} = & T_{21}(q_f, \mu\sigma)(-f_1(q_f, \dot{q}_f, \mu\sigma, \mu\ddot{\sigma}) \\ & - g_1(q_f, \mu\sigma) + \tau) + T_{22}(q_f, \mu\sigma) \\ & (-f_2(q_f, \dot{q}_f, \mu\sigma, \mu\ddot{\sigma}) - g_2(q_f, \mu\sigma) - \tilde{k}\sigma), \end{aligned} \quad (6)$$

which obtain the slow and fast subsystem for the singular perturbation model of the flexible joint manipulator with $\mu \rightarrow 0$ in equation (5) and (6), actually, setting $\mu=0$ and solving for $\bar{\sigma}$ in (6), as follows:

$$\begin{aligned} \tilde{k}\bar{\sigma} = & T_{22}^{-1}(\bar{q}_f, 0)[T_{21}(\bar{q}_f, 0)f_1(\bar{q}_f, \dot{\bar{q}}_f, 0, 0) \\ & - T_{21}(\bar{q}_f, 0)g_1(\bar{q}_f, 0) + T_{21}(\bar{q}_f, 0)\tau \\ & - T_{22}(\bar{q}_f, 0)f_2(\bar{q}_f, \dot{\bar{q}}_f, 0, 0) \\ & - T_{22}(\bar{q}_f, 0)g_2(\bar{q}_f, 0)]. \end{aligned} \quad (7)$$

Note that the over bars in equation (7) are used to indicate rigid model. Substitution equation (7) into equation (5) gives:

$$\begin{aligned} \ddot{\bar{q}}_f = & [T_{11}(\bar{q}_f, 0) - T_{12}(\bar{q}_f, 0)T_{22}^{-1}(\bar{q}_f, 0)T_{21}(\bar{q}_f, 0)] \\ & [\tau_s - f_1(\bar{q}_f, \dot{\bar{q}}_f, 0, 0) - g_1(\bar{q}_f, 0)]. \end{aligned} \quad (8)$$

where τ_s is control input for slow subsystem.

Note that:

$$M_{11}^{-1}(\bar{q}_f, 0) = [T_{11}(\bar{q}_f, 0) - T_{12}(\bar{q}_f, 0)T_{22}^{-1}(\bar{q}_f, 0)T_{21}(\bar{q}_f, 0)], \quad (9)$$

where $M(\bar{q}_f) \in \mathbb{R}^{n \times n}$ is positive definite for every rigid robot (M_{11} is first block of the inertia matrix). Therefore equation (8) can be written as:

$$\ddot{\bar{q}}_f = M_{11}^{-1}(\bar{q}_f, 0)[\tau_s - f_1(\bar{q}_f, \dot{\bar{q}}_f, 0, 0) - g_1(\bar{q}_f, 0)] \quad (10)$$

Now we want to use the state-space theorem for this system. Use $x_1 = \bar{q}_f$, $x_2 = \dot{\bar{q}}_f$, $z_1 = \bar{\sigma}$, $z_2 = \varepsilon\bar{\dot{\sigma}}$ and $\varepsilon = \sqrt{\mu}$ [18], to write the state-space model of system as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= T_{11}(x_1, \varepsilon^2 z_1)(-f_1(x_1, x_2, \varepsilon^2 z_1, \varepsilon z_2) \\ & - g_1(x_1, \varepsilon^2 z_1) + \tau) + T_{12}(x_1, \varepsilon^2 z_1) \\ & (-f_2(x_1, x_2, \varepsilon^2 z_1, \varepsilon z_2) - g_2(x_1, \varepsilon^2 z_1) - \tilde{k}z_1) \end{aligned} \quad (11)$$

$$\begin{aligned} \varepsilon\dot{z}_1 &= z_2 \\ \varepsilon\dot{z}_2 &= T_{21}(x_1, \varepsilon^2 z_1)(-f_1(x_1, x_2, \varepsilon^2 z_1, \varepsilon z_2) \\ & - g_1(x_1, \varepsilon^2 z_1) + \tau) + T_{22}(x_1, \varepsilon^2 z_1) \\ & (-f_2(x_1, x_2, \varepsilon^2 z_1, \varepsilon z_2) - g_2(x_1, \varepsilon^2 z_1) - \tilde{k}z_1). \end{aligned} \quad (12)$$

Now, we define slow subsystem for singular perturbation model then formally obtained by setting $\varepsilon=0$ as follows:

$$\begin{aligned} \dot{\bar{x}}_1 &= \bar{x}_2 \\ \dot{\bar{x}}_2 &= M_{11}^{-1}(\bar{x}_1, 0)[\tau_s - f_1(\bar{x}_1, \bar{x}_2, 0, 0) - g_1(\bar{x}_1, 0)]. \end{aligned} \quad (13)$$

Now for fast subsystem, we introduce the fast time scale $t_s = t/\varepsilon$. Then we rewrite systems (11) and (12) in the fast time-scale as follows:

$$\begin{aligned} \frac{dx_1}{dt_s} &= \varepsilon x_2 \\ \frac{dx_2}{dt_s} &= \varepsilon[-T_{11}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))f_1(x_1, x_2, \varepsilon^2(\eta_1 + \bar{\sigma}), \varepsilon\eta_2) \\ & - T_{11}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))g_1(x_1, \varepsilon^2(\eta_1 + \bar{\sigma})) + T_{11}\tau \\ & - T_{12}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))f_2(x_1, x_2, \varepsilon^2(\eta_1 + \bar{\sigma}), \varepsilon\eta_2) \\ & - T_{12}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))g_2(x_1, \varepsilon^2(\eta_1 + \bar{\sigma})) \\ & - T_{12}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))\tilde{k}(\eta_1 + \bar{\sigma})], \end{aligned} \quad (14)$$

$$\begin{aligned} \frac{d\eta_1}{dt_s} &= \eta_2 \\ \frac{d\eta_2}{dt_s} &= \varepsilon[-T_{21}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))f_1(x_1, x_2, \varepsilon^2(\eta_1 + \bar{\sigma}), \varepsilon\eta_2) \\ & - T_{21}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))g_1(x_1, \varepsilon^2(\eta_1 + \bar{\sigma})) + T_{21}\tau \\ & - T_{22}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))f_2(x_1, x_2, \varepsilon^2(\eta_1 + \bar{\sigma}), \varepsilon\eta_2) \\ & - T_{22}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))g_2(x_1, \varepsilon^2(\eta_1 + \bar{\sigma})) \\ & - T_{22}(x_1, \varepsilon^2(\eta_1 + \bar{\sigma}))\tilde{k}(\eta_1 + \bar{\sigma})], \end{aligned} \quad (15)$$

where the new fast variables are obtained as follows:

$$\eta_1 = z_1 - \bar{\sigma}, \eta_2 = z_2 \quad (16)$$

Let $\varepsilon=0$, then gives $\frac{dx_1}{dt_s} = \frac{dx_2}{dt_s} = 0$ and note that in fast

(boundary layer) subsystem x_1 and x_2 are constant furthermore it can be found easily $g_1 = g_2 = f_1 = f_2$ are zero. Therefore fast subsystem can be found as follows:

$$\begin{aligned} \frac{d\eta_1}{dt_s} &= \eta_2 \\ \frac{d\eta_2}{dt_s} &= -T_{22}(\bar{x}_1, 0)\tilde{k}\eta_1 + T_{21}(\bar{x}_1, 0)\tau_f. \end{aligned} \quad (17)$$

III. NEURO-ADAPTIVE CONTROLLER FOR SLOW SUBSYSTEM:

In this section presents an adaptive controller [20] for slow subsystem in last section. Consider a system as follows:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= x \end{aligned} \quad (18)$$

where x is state vector, $f(x)$ and $g(x)$ are unknown smooth functions, $u \in \mathbb{R}$ and $y \in \mathbb{R}$ is output of the system. Since $f(x)$ and $g(x)$ are unknown, traditional controller are not applicable for the system.

Assumption 1: the desired trajectory vector x_d is given and bounded. We know $\|x_d\| \leq L$ and L being a known bound.

Assumption 2: the system can be define with input gain bounded by $0 < g_0 < g(x)$ and $g(x)$ only depends on the x .

Consider the following state dependent transformation:

$$v = g(x)u^* \quad (19)$$

where u^* is the ideal control function and v is commonly referred to as the pseudo control signal. The pseudo control system is chosen in this derivation as a linear operator. It may also, in general, be nonlinear, for example if a sliding mode component is including [21], [11].

Typically the desired trajectory satisfying by assumption 1 can be generated by the following system.

$$\dot{x}_n^d = -m_1 x_1^d - \dots - m_n x_n^d, \quad (20)$$

where $m_1 + m_2 + \dots + m_n$ are chosen that the polynomial

$$s^n + m_n s^{n-1} + \dots + m_1$$

is Hurwitz. Now define tracking error $e = x^d - x$ where x^d is desired trajectory in (20).

Therefore, the error dynamics can be written as:

$$\dot{e}_n = \dot{x}_n^d - \dot{x}_n. \quad (21)$$

We can rewrite (21) we can write:

$$\dot{e}_n = \dot{x}_n^d - [f(x) + g(x)(u + u^* - u^*)]. \quad (22)$$

Based on the transformation introduced in equation (19) we can rewrite (22):

$$\dot{e}_n = \dot{x}_n^d - v - [f(x) + g(x)(u - u^*)]. \quad (23)$$

The pseudo control v is designed as:

$$v = \dot{x}_n^d + k_n e_n + \dots + k_1 e_1 + v', \quad (24)$$

where coefficients are chosen so that each $L(s) = s^n + k_n s^{n-1} + \dots + k_1$ has its roots in the open left-half open, i.e., $L(s)$ is Hurwitz, and v' is a nonlinear term approximated later. From (23) and (24), equation (23) becomes:

$$\dot{e}_n = k_n e_n + k_{n-1} e_{n-1} L + k_1 e_1 - [f(x) + g(x)(u - u^*)] - v'. \quad (25)$$

By introducing $e = [e_1, e_2, \dots, e_n]^T$ the error dynamics (25) can be further written in a matrix form as:

$$\dot{e}_n = A e - b[f(x) + g(x)(u - u^*) + v'], \quad (26)$$

where Hurwitz matrix A and vector b are given by

$$\begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 1 \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & 0 & \dots & 1 \\ -\alpha_0 & -\alpha_1 & -\alpha_2 & \dots & -\alpha_{n-1} \end{pmatrix}, b = [0, 0, \dots, 1]^T. \quad (27)$$

Since A is Hurwitz, a unique positive-definite solution P to the following Lyapunov equation exists:

$$A^T P + P A = -Q, \quad (28)$$

where the matrix Q is positive definite.

Let $f(x)$ be a smooth function from $\mathbb{R}^n$ to $\mathbb{R}^q$. Then, given a compact $s \in \mathbb{R}^n$ and a positive number ε_M , there exists a NN system such that:

$$f(x) = \hat{B}^T \Psi(z) + \varepsilon \quad (29)$$

where $\hat{B}$ is a vector of network weights. $\Psi(z) \in \mathbb{R}^p$ is the RBNN function and ε is the NN approximation error that $\|\varepsilon\| \leq \varepsilon_M$ with ε_M constant for all x and integer p is number of NNs. An adaptive algorithm is then defined to estimate B^* with $\hat{B}$. These estimates are used to formulate the control law as:

$$u = \hat{B}^T \Psi(z) + u_K + u_R, \quad (30)$$

with

$$u_R = \text{sgn}(b^T p e) \hat{\theta} + \text{sgn}(b^T p e) v'. \quad (31)$$

In the above control law $\hat{B}^T \Psi(z)$ is used to approximate an ideal controller for the system through the input vector z .

The second term, u_K in (30) is a prior continuous controller (possibly PI, PID or some other type of controller) designed in advanced via heuristic or past experience and $\text{sgn}(b^T p e) \hat{\theta}$, $\text{sgn}(b^T p e) v'$ are utilized for countering uncertainties in the RBNN approximation error.

Consider the following update laws:

$$\dot{\hat{B}} = \Gamma_B (b^T p e) \Psi(z) \quad (32)$$

$$\dot{\hat{\theta}} = \Upsilon_\theta |b^T p e| \quad (33)$$

$$\dot{v}' = \Upsilon_{v'} |b^T p e| \quad (34)$$

where $\Gamma_B, \Upsilon_\theta, \Upsilon_{v'} > 0$ are constant design parameters. The main results of this section are now summarized in the following theorem, a proof of which is given in [20].

Theorem: Given the error dynamical system (26) for the system (13) then the control law (30) with adaptation law (32) – (34), makes the tracking error asymptotically converge to zero and all signals in the closed loop system bounded.

V. DESIGN OF COMPOSIT CONTROLLER

Combined control method is considered as following relation using singular perturbation theorem

$$U = u_f + u_s \quad (35)$$

where u_f is for the fast subsystem and u_s is for the slow subsystem respectively. Slow subsystem controlled by nero-adaptive controller presented in last section and for fast subsystem used state feedback control. State feedback need to a pair (A_f, B_f) , they are given in equation (17). Slow and fast subsystem have a different in time, then the slow subsystem need to time scale $(t_s = t / \varepsilon)$ for matching with fast subsystem.

VI. Simulations

In this section the adaptive controller based on the RBNN proposed in section 3 is then applied for two examples. The parameter values for RLFJs are shown in table 1.

Example 1: A rigid link flexible joint manipulator (RLFJ) with one link is used for proposed RBNN controller. An input desired signal is chosen for this system as follows:

$x_d = \sin(t)$. The control objective is to make the flexible-joint trajectory follow the given desire trajectory. We applied the controller:

$$u = \hat{B}^T \Psi(z) + u_K + \text{sgn}(b^T p e) \hat{\theta} + \text{sgn}(b^T p e) v' \quad (36)$$

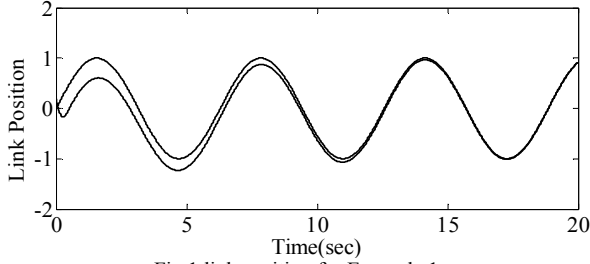


Fig.1 link position for Example 1

where $u_K = -50(e - 0.1 \frac{de}{dt} - 0.1 \int_0^t e dt) \cdot \hat{B}$, $\hat{\theta}$ and v' are updated by adaptive rules (32- (34). The basis functions $\Psi(z)$ are chosen as $\Psi(z) = \exp(-\|z - \zeta_{\psi_q}^2\| / \sigma_{\psi_q}^2)$, where ζ_{ψ} and σ_{ψ} are the centers and size of influences of the basis function respectively ($q = 1, 2, \dots, 80$). The input vector for the RBNN basis $\Psi(Z)$ is $z = [x_1, x_2, v']^T$. The RBNN nodes are chosen as $q = 80$, with the centers

TABLE 1
SYSTEM PARAMETERS

PARAMETER	VALUE
Link 1,2 length (m)	1
Gravity acceleration (m / s ²)	9.81
Spring constant l (N m r a d ⁻¹)	100
Link 1,2 inertia (k g m ²)	0.1
Motor 1,2 inertia (k g m ²)	0.1
Load (kg)	0.1
Hub 1,2 inertia (k g m ²)	0.1
Mass of motor 1,2 (kg)	1
Mass of link 1,2 (kg)	1
Mass of hub 1,2 (kg)	0.5
Mass of link 1,2 (kg)	1

$\zeta_{\psi} = [\zeta_{\psi,q1}, \zeta_{\psi,q2}, \zeta_{\psi,q3}]^T$ evenly spaced between $[-1,1]$, $[-5,5]$, $[-35,35]$ respectively, and the size of

influences $\sigma_{\psi} = 0.5$ and ($q = 1, 2, \dots, 80$). The initial RBNN weights

$\hat{B}$, $\hat{\theta}$ and v' are simply set to zeros. The controller parameters are taken as $\Gamma_B = 25 \times 10^2$, and $\Upsilon_{\theta} = \Upsilon_{v'} = 25 \times 10^3$. Figures 1-3 are shown the simulation results for the design controller. The gains k_f are obtained for the fast controller as $k_f = -3.0909, -0.2000$.

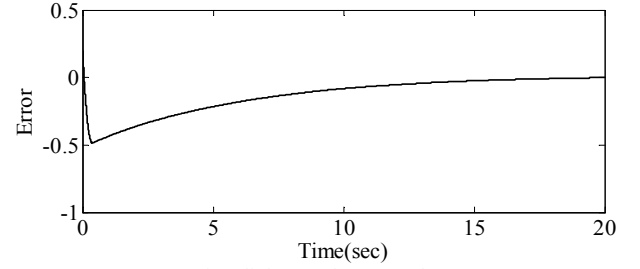


Fig.2 link error for Example 1

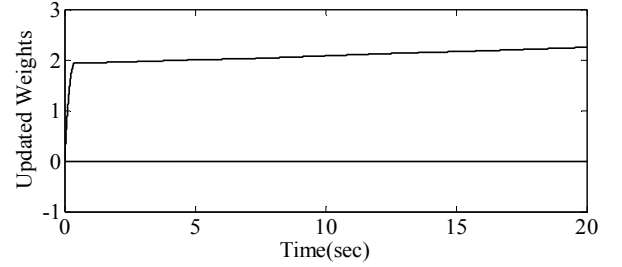


Fig.3 updated weights for Example 1

Example 2: consider a RLFJ with two link, we used the RBNN for this system. The parameter values are shown in table 2. Two desired trajectory for this system are $x_{d1} = x_{d2} = \sin(0.1t)$. Simulation results for Example 2 are shown in Figures (4-8). The gains k_f are obtained for the fast controller as $k_f = 10.6311, -16.0656, -3.1022, -2.2996$.

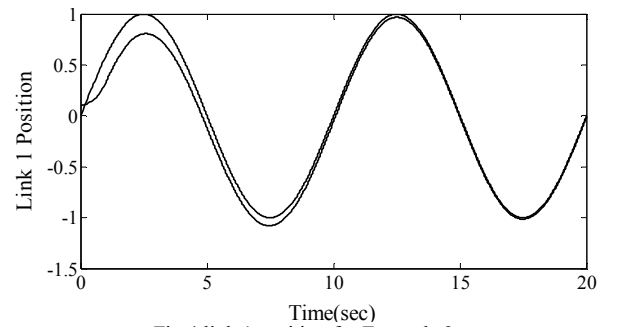


Fig.4 link 1 position for Example 2

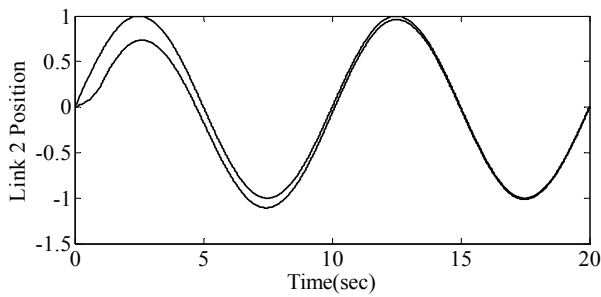


Fig.5 link 2 position for Example 2

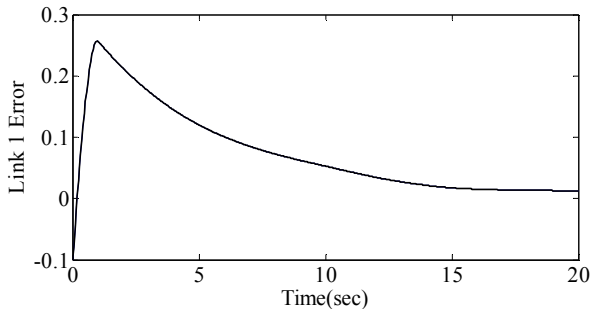


Fig.6 link 1 error for Example 2

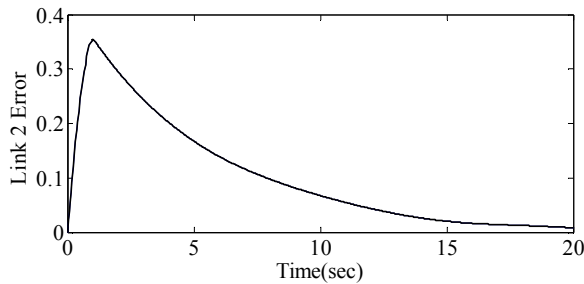


Fig.7 link 2 error for Example 2

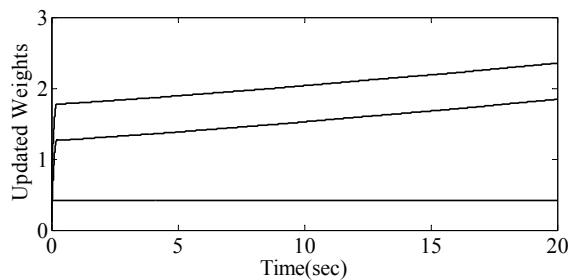


Fig.8 updated weights for Example 2

Compare between *Ex.1* and *Ex.2* depicted answer of the controller is better for *Ex.1*.

CONCLUSION

This paper introduced a RBNN adaptive controller for flexible joint manipulator with unknown parameters. We used the singular perturbation method for robot dynamics. The proposed RBNN adaptive controller ensured the closed-loop stability and convergence of the tracking errors asymptotically to zero. The simulation examples approved the validity of the proposed controller.

REFERENCES

- [1] Larry M. Sweet; Malcolm C. Good, "Re-Definition of the Robot Motion Control Problems: Effects of Plant Dynamics, Drive System Constraints, and User Requirements," *Proc. IEEE Conf. Decis. Contr.* Vol: 23 (1984) , pp. 724 - 732.
- [2] Cesareo, G. and R. Marino, "On the Controllability of Elastic Robots," in *Proc. INRIA Int. Conf., Lecture Notes in Control and Information and Science, Springer-Verlag* (1984).
- [3] Er, M.J. Lee, S.C. Tan, L.L., "Digital-signal-processor-based multirate PID control of a two-link flexible-joint robot," *IEEE TENCON. Digital Signal Processing Applications*, Vol.2, Conf. Site. Perth, WA, Nov 1996 , PP. 900 -905.
- [4] Widmann, G.R. and S. Ahmad, "Control System Design of Robots with FJs," in *Recent Trends in Robotics: Modeling, Control and Education.* (1986).
- [5] Marino, R. and M.W. Spong, "Nonlinear Control Techniques for Flexible Joint Manipulators," *Proc. IEEE Int. Conf. Rob. Autom.*, vol. 3, (1986) , pp. 1030 – 1036.
- [6] Han, M.C. Chen, Y.H., "Robust control design for uncertain flexible-joint manipulators: a singular perturbation approach," *Proc. IEEE Conf. Decis. Contr.*, 1993, pp. 611-616.
- [7] Al Ashoor, R.A., R.V. Patel, and K. Khorasani, "Robust Adaptive Controller Design and Stability for FJM," *IEEE Trans. Syst., Man Cybern.*, Vol.23 (1993) , pp. 589-595.
- [8] Khorasani, K., "Adaptive Control of FJR," *Proc. IEEE Int. Conf. Rob. Autom.*, vol. 3 (1991) , pp. 2127 – 2134.
- [9] Slotine, J.J. and W. Li, "Adaptive Manipulator Control: A Case Study," *IEEE Trans. Automat. Contr.*, Vol. 33, No.11 (1988), pp. 995-1003.
- [10] Ghorbel, F., J.H. Hung, and M.W. Spong, "Adaptive Control of FJMs," *IEEE Contr. Syst. Mag.*, Vol. 2 (1989) , pp. 1188 - 1193.
- [11] J.T.Spooner and K.M. Kramer, and L.H.Ungar, "Decentralized adaptive control of nonlinear systems using radial basis neural network," *IEEE trans. Autom. control*, Vol. 44, No. 11, 1999, pp. 2050-2057.
- [12] Chaoui, H., P. Sicard, and A. Lakhsasi, "Reference Model Supervisory Loop for Neural Network Based Adaptive Control of a Flexible Joint with Hard Nonlinearities," *Proc. Canadian Conf. Electrical Comput. Eng.*, Vol. 4 (2004) , pp. 2029-2034.
- [13] Kim, H. and J.K. Parker, "Artificial Neural Network for Identification and Tracking Control of a FJ Single Link Robot," *Proc. IEEE SSST, Southeastern Symp. Syst. Theory* (1993) , pp. 233 - 237.
- [14] Kwan, C.M., F.L. Lewis, and Y.H. Kim, "Robust Neural Network Control of FJR," *Proc. IEEE Conf. Decis. Contr.*, vol. 2 (1995) , pp. 1296 - 1301.
- [15] W.J. Book, " Recursive Lagrangian Dynamics of Flexible Manipulator Arms," *Int. J. Rob. Res.*, Vol. 3, 1984, pp. 87-101.
- [16] Siciliano, B., and Book, W. J., A singular perturbation approach to control of lightweight flexible manipulators. *International Journal of Robotics Research*, vol. 7. 1988, pp. 79–90.
- [17] B. Subudhi and Morris. A. S, "singular perturbation approach to trajectory tracking of flexible robot with joint elasticity," *International Journal of System Science*, Vol. 34, Feb 2003, pp. 167-179.
- [18] Khorasani, K. and M.W. Spong, "Invariant Manifolds and Their Application to Robot Manipulators with FJs," *Proc. IEEE Int. Conf. Rob. Autom.*, vol. 2 (1985) , pp. 978 - 983.
- [19] Zeman, V. Patel, R. V. Khorasani, K, "A Neural Net Controller for Flexible-Joint Manipulators," *American Control conference*, , 1990 pp. 3025-3033.
- [20] Karimi, B, Menhaj. M. B, Saboori. I, "Decentralized Adaptive Control of Large Scale Nonaffine Nonlinear System Using Radial Basis Function Neural Network," *IEICE trans, FUNDAMENTALS*, Vol. E90-A, No. 10, Oct 2007.
- [21] S.S.sastri, *nonlinear systems*, springer, 1999.

Probabilistic car-like robot path planning from temporal logic specifications

Narcis Ghita, Marius Kloetzer and Octavian Pastravanu

Abstract—An algorithmic framework is developed for automatic deployment of car-like robots based on Linear Temporal Logic (LTL) formulae over a set of regions of interest in the environment. The environment and the regions of interest are a priori known, and the robot has non-negligible size and restricted steering capabilities. The approach relies on constructing a probabilistic finite-state abstraction of the car-like robot and on finding a trajectory (run) in this abstraction such that the probability of satisfying the LTL formula is maximized. The feasibility of our approach is supported by simulations under Matlab environment.

I. INTRODUCTION

Temporal logics (LTL, CTL and CTL*) were initially intended as a framework for specifying and verifying the correctness of digital circuits and computer programs [1]. However, in the last years, Linear Temporal Logic (LTL) became a useful tool for expressing robot motion tasks, due to its rich expressivity and its resemblance to natural language [2], [3], [4], [5].

There can be identified two main directions in the area of mobile robot path planning. The first one focuses on complex environments and rich tasks, while considering robots with simple dynamics, *e.g.* fully actuated or differential-wheel driven with negligible size. The second one focuses on simpler specifications and environments, while considering robots with complex mathematical models, as for example car-like robots with nonholonomic constraints and non-negligible size [6], [7], [8]. In [7], a car-like robot is controlled from a LTL specification, but the environment is specifically restricted to a parking scenario. In [6] an LTL-based planner for a car-like robot is envisioned, but the approach is not fully automated. [8] develops an automatic strategy for planning an controlling a car-like robot, but the framework is restricted only to obstacle avoidance and reachability of a target position, rather than more complex specifications.

This paper proposes an automated framework for planning car-like robots from LTL specifications over a set of polygonal regions of interest from the environment. The main steps imply constructing of a finite representation for the car-like robot and finding a trajectory that the robot can follow, based on a sequence of regions satisfying the specification. Our approach is probabilistic, the probabilities arising from the way we abstract the car-like robot to a finite state representation.

The authors are with The "Gheorghe Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering, Department of Automatic Control and Applied Informatics

The corresponding author is Marius Kloetzer kmarius@ac.tuiasi.ro

The problem of probabilistic planning for mobile robots based on high level specifications was addressed in [9], [10]. In [9], a controller that increases the probability of satisfying an LTL formula was designed for a stochastic linear system. The solution consists in abstracting the environment to a partially observed Markov decision process, using the tool from [4] for finding a satisfying run, and finally designing a controller that maximizes the probability of following the returned sequence of states. However, the tool from [4] is designed for deterministic rather than probabilistic systems, and therefore a maximum probability trajectory cannot be found as in [9]. [10] overcomes this problem by finding an optimum run over all possible behaviors, but it restricts to CTL rather than LTL specifications, and it lacks an automated construction of the Markov decision process. Moreover, the approach from [10] is applicable to fully-actuated mobile robots, and its extension to nonholonomic robots is not straightforward.

The main contribution of this paper is to provide an algorithm for designing a car-like robot trajectory that has the maximum probability of satisfying a given LTL formula. As important additions to the robotic LTL planning field, our solution implies a strategy for constructing a probabilistic finite-state representation for a car-like robot, and an extension of algorithm from [4] such that an optimum run for a probabilistic representation is returned.

The remainder of the paper is organized as follows. Sec. II presents some preliminary notions and formulates the problem. The solution is described in Sec. III, some simulation results are presented in Sec. IV, and final remarks are given in Sec.V.

II. PRELIMINARIES AND PROBLEM STATEMENT

This section first introduces the car-like model and some notions that will be used throughout the paper. Then, it formulates the specific problem we focus on and outlines the proposed solution.

A. Preliminary Notions

We consider a car-like robot as in Fig. 1, described by the kinematic model (1).

$$\begin{aligned}\dot{x} &= v \cos \theta \\ \dot{y} &= v \sin \theta \\ \dot{\theta} &= \frac{v}{L} \tan \varphi\end{aligned}\tag{1}$$

where (x, y) represents the coordinates of the point P_m located at mid-distance of the rear (actuated) wheels, θ characterizes the robots chassis orientation in a static world frame S , v is

the vehicles forward velocity, imposed by the rear wheels, φ represents the vehicles steering wheel angle, bounded by $\varphi \in [-\varphi_{max}, \varphi_{max}]$, and L, l are the distances between robot's wheels.

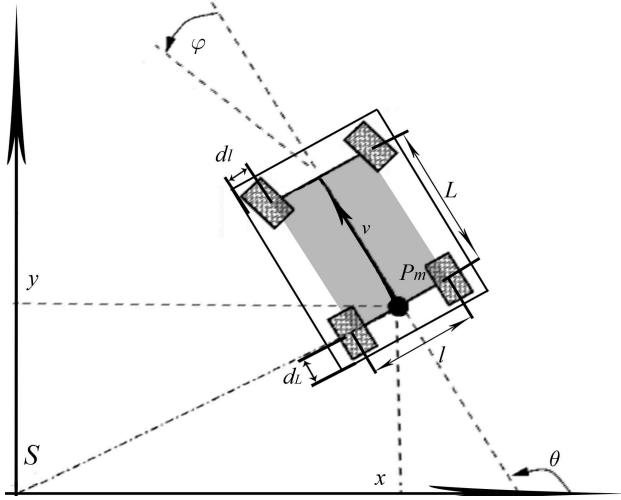


Fig. 1. Geometrical representation of a car-like robot

Although the kinematic model of the car-like robot may seem restrictive, it is a common choice for solving various motion planning problems [6], [7], [11], [12], [13], whereas considering a dynamical model leads to increased computation costs [14] and is beneficial especially when the robot is capable of moving with high speeds. Regardless of the considered model, the nonholonomic constraints imply that a car-like robot can only follow certain smooth paths. In [15], [16] it was proved that an optimal trajectory between a start and a goal position of a car-like robot consists from smooth connections between arcs of circle and line segments. The radius of necessary circle arcs is imposed by bounds on φ , and an element of such a trajectory (arc of circle or line segment) is called *CL curve*.

As part of the solution for the problem we'll focus on, we will abstract the motion capacities of a car-like robot with model (1) into a probabilistic abstraction as defined below.

Definition 1: (Markovian Transition System) A Markovian transition system is a tuple $T = (Q, Q_0, \mathcal{P}, \Pi, h)$, where:

- Q is a finite set of states,
- $Q_0 \subseteq Q$ is a set of initial states,
- $\mathcal{P} : Q \times Q \rightarrow [0, 1]$ defines probabilistic transitions between states of T ,
- Π is the observation (output) set,
- $h : Q \rightarrow \Pi$ is a satisfaction relation.

For simplicity of exposition we did not include input control symbols in definition 1. For a given state $q \in Q$, a certain input may be applied to T with the goal of changing the current state to $q' \in Q$. However, the transition from q to q' is probabilistic, meaning that state q' is directly reached from q with probability $\mathcal{P}(q, q')$. A trajectory (or run) of T starting from $q \in Q_0$ is an infinite sequence $r \in Q^\omega$, $r = r_1, r_2, r_3, \dots$, with the property that $r_1 = q$ and $\mathcal{P}(r_i, r_{i+1}) \neq 0$, $\forall i \geq 1$.

The elements of T will gain a specific meaning related to the car-like robot in Section III.

The motion tasks for the car-like robot will be expressed as Linear Temporal Logic (LTL_X¹) formulae over a set of predicates (regions) Π from the environment. An LTL_X formula is composed by standard Boolean operators ($\neg$ - negation, $\vee$ - disjunction, $\wedge$ - conjunction, $\Rightarrow$ - implication, and $\Leftrightarrow$ - equivalence) and by temporal operators with straightforward meaning ($\Box$ - always, $\Diamond$ - eventually, and $\mathcal{U}$ - until). Rich, human-like specifications can be obtained by interconnecting these operators, and details on syntax, semantics, and usage of LTL_X in connection with continuous systems can be found in [1], [2], [4].

An LTL formula can be transformed into a Büchi automaton, whose structure is given in definition 2.

Definition 2: (Büchi automaton) A Büchi automaton is a tuple $B = (S, S_0, \Sigma_B, \rightarrow_B, F)$, where:

- S is a finite set of states,
- $S_0 \subseteq S$ is a set of initial states,
- Σ_B is the input alphabet,
- $\rightarrow_B \subseteq S \times \Sigma_B \times S$ is a nondeterministic transition relation,
- F is the observation (output) set,
- $F \subseteq S$ is the set of accepting (final) states.

The semantics of a Büchi automaton are defined over infinite input words. An input word $\omega \in \Sigma_B^\omega$ is *accepted* by B if and only if there exists a run produced by ω that visits infinitely often set F . Whenever B has at least one accepting run, then there exists a run in a *prefix-suffix* form, consisting from a finite string called prefix, followed by infinite repetitions of another finite string called suffix. Of course, such a prefix-suffix representation of an infinite run can be stored on a finite memory machine.

For any LTL formula ϕ over a set of predicates Π , there exists a Büchi automaton B_ϕ with input alphabet $\Sigma_B = 2^\Pi$ accepting all and only the infinite strings over Π satisfying formula ϕ [17]. If multiple predicates cannot be simultaneously satisfied (as it will be our case), then $\Sigma_B = \Pi$. In this paper we will use conversion algorithms from LTL formulae to Büchi automata from [18], further adapted for solving planning tasks in [4].

B. Problem Formulation

We consider a car-like robot evolving in a 2D rectangular environment where a set Π of convex polygonal regions of interest is defined.

Problem 1: Given a motion task in form of an LTL_X formula ϕ over the set of regions Π , find a control strategy for car-like robot (1) that maximizes the probability of satisfying the formula ϕ .

The motion of car-like robot (1) satisfies an LTL formula if the sequence of visited regions from Π satisfies the formula. We say that the robot visits a region if point P_m is included in that region for a time moment, and the robot avoids a region if any of its points do not intersect that region for any time moment.

¹LTL_X lacks the “next” operator, which exists in standard LTL, but this operator is meaningless for continuous trajectories.

The proposed approach for solving problem 1 follows three main steps. (i) The environment and the regions of interest are partitioned using a cell decomposition where adjacent cells exactly share the common facet, as in triangular decompositions [11] and polytopal ones [4]. (ii) We abstract the motion capabilities of the car-like robot in the partitioned environment into a Markovian transition system T , where the probabilities reflect the possibility of steering the car-like robot from the current cell to an adjacent one, without assuming any information on the initial placement of the robot in the current cell. Although the assumption adds conservativeness to our solution, it is necessary for having a Markovian (history-free) abstraction, on which we can further solve the problem. (iii) The LTL formula is converted into a Büchi automaton B and a specific product automaton between T and B is performed. An accepting run with maximum probability of being followed is found in this product, and it is further projected into a run of T , which yields the control strategy for the car-like robot.

III. SOLUTION

This section provides details on the above-mentioned steps for obtaining a solution to problem 1. Details on cell decomposition methods (step (i)) are not included, and we refer to [11] for general descriptions and to [19] for implementations. Subsection III-A details step (ii) of the solution, subsection III-B presents step (iii), and subsection III-C unifies all steps in an algorithmic description that solves Problem 1.

A. Construction of T

The set Q of states of T equals the set of cells from the obtained decomposition, and the set of initial states corresponds to initial possible deployment of the car-like robot. The observation set equals the set of regions of interest (Π), and $h(q) = \pi$ if and only if cell q is included or equal with region π , while $h(q) = \emptyset$ for all cells q that correspond to the left-over (free) space (not included in any region from Π). Two cells are called adjacent if and only if they share a facet (not a single vertex). Let $\delta : Q \rightarrow 2^Q$ be a function capturing the adjacency among cells, *i.e.* for any state $q \in Q$, $\delta(q)$ is the set of cells adjacent to cell q , without including q .

The difficult part in constructing T is determining the probabilistic transition function between its states ($\mathcal{P}$). Before explicitly constructing $\mathcal{P}$, we provide some background intuition on car-like deployment policy in partitioned environments. If there is available a succession of adjacent cell regions that the robot must follow, first an angular path is constructed by linking the midpoints of common facets of successive pairs of cells. Since the car-like robot cannot directly follow the resulting succession on line segments, a smooth path is constructed by adding arc circles between line segments with different slope, the result being a succession of CL curves. The radius of each arc circle is greater or equal to R_{min} , which is imposed by steering limits. The value of

R_{min} is obtained from Ackermann geometries as:

$$R_{min} = \sqrt{\left(\frac{l}{2} + \frac{L}{\sin(\phi_{max})}\right)^2 - L^2} - \frac{l}{2} \quad (2)$$

Assume that at a time moment the robot enters a cell q from the environment partition. If one would know the history of visited cells until q is entered, then one can compute the entering point and orientation in q . Further, this point and orientation reduces the set of feasible CL curves, and thus one can decide the next possible cells that can be visited.

However, at the abstraction stage we cannot know the history of visited cells (since the task specification has not been considered, yet), so we cannot deterministically decide the transitions between states of T . Therefore, we compute the probabilistic transitions between adjacent cells. As we will further explain, the probabilities are based on the existence of a controller between two adjacent cells, under all possibilities of previous- and following- cells. Also, the probabilities will account situations when the physical size of the robot induces visits to other cells than the intended ones.

More specifically, let us compute the probability $\mathcal{P}(q_i, q_j)$ of going from state q_i directly to state q_j , $\forall q_i, q_j \in Q$. We obviously have $\mathcal{P}(q_i, q_j) = 0$ if q_i and q_j are not adjacent, *i.e.* $q_j \notin \delta(q_i)$, and $\mathcal{P}(q_i, q_j) = 1$ if $q_i = q_j$ (in this case, stopping the robot in cell q_i guarantees the desired transition).

In the following, we refer to the general case $q_j \in \delta(q_i)$. First, we aim in finding the number $N_{ctrl}^{i,j}$ of cases when it is possible to control the robot from cell q_i to cell q_j . For this, let us assume that the car-like robot should follow the sequence of states $q'q_iq_jq''$, with $q' \in \delta(q_i) \setminus \{q_j\}$ and $q'' \in \delta(q_j) \setminus \{q_i\}$. We denote by $N^{i,j}$ the total number of such sequences of four states. Any such sequence is sufficient for testing if a smooth path traversing q_i and q_j can be constructed, by checking if an arc of circle with radius greater than R_{min} can be used for smoothing out the angular path given by the midpoints of common facets of successive cells. Algorithms for performing such tests were used in [8], and therefore $N_{ctrl}^{i,j}$ (with $N_{ctrl}^{i,j} \leq N^{i,j}$) can be found. At this point, it may seem tempting to simply consider the ratio $N_{ctrl}^{i,j}/N^{i,j}$ as being the searched probability, but this would be erroneous in some cases, because we did not consider the size of the robot, yet. This is because while traveling on the smooth path from q_i to q_j , the robot may span area outside of cells q_i and q_j , and this may lead to violating the imposed LTL formula, simply because those areas may satisfy regions from Π that are deemed as obstacles by the formula.

Therefore, in the second step for constructing $\mathcal{P}(q_i, q_j)$, we search for the number $N_{feas}^{i,j}$ of cases in which the movement from q_i to q_j also induces that all regions from $\Pi \setminus \{h(q_i) \cup h(q_j)\}$ are avoided during this movement. For this, we construct the area swept by the robot in all cases out of $N_{ctrl}^{i,j}$ ones. Consider a generic case from the $N_{ctrl}^{i,j}$ situations when a smooth trajectory can be constructed, as depicted in Fig. 2. By using the notations introduced by Fig. 2, the area

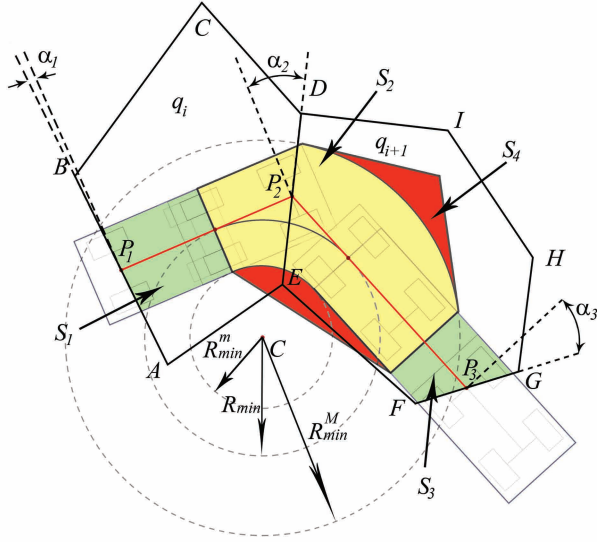


Fig. 2. Area swept by the car-like robot while moving in two adjacent cells q_i and q_{i+1}

swept by the robot from the instant when the robot's point P_m enters q_i until it leaves q_j is $S_1 \cup S_2 \cup S_3$. S_4 is a polyhedral over-approximation of S_2 that will be used in some tests.

By denoting with $\mathcal{F}$ the set of all facets of regions q_i and q_j , and by referring to angles α_1 , α_2 , α_3 marked in the geometrical representation from Fig. 2, the sufficient conditions for the robot to stay inside $q_i \cup q_j$ are:

$$\begin{aligned} \cos(\alpha_1) &\geq \frac{(l+2dl)}{AB} \\ \cos(\alpha_3) &\geq \frac{(l+2dl)}{FG} \\ S_4 \cap \mathcal{F}_k &= \emptyset, \forall \mathcal{F}_k \in \mathcal{F} \setminus \{[A B], [E D], [F G]\} \end{aligned} \quad (3)$$

The first two conditions from (3) guarantee that when entering q_i and when leaving q_j the robot crosses only facet AB , respectively FG . The last condition² guarantees that the over-approximation S_4 does not include regions outside $q_i \cup q_j$. Although the last condition could be replaced with two linear programming problems, we use the current formulation because it gives us more flexibility, as further detailed.

If $S_4 \cap \mathcal{F}_k \neq \emptyset$ for some $\mathcal{F}_k \in \mathcal{F}$ (as is the situation from Fig. 2), we develop necessary conditions (4) such that the robot remains inside $q_i \cup q_j$, even though the over-approximation S_4 does not.

$$\begin{aligned} C_o E &\leq R_{min}^m, \text{ if } E \in \mathcal{V}^{inf} \\ C_o \mathcal{V}_i^{sup} &\geq R_{min}^M \\ d(C_o \mathcal{F}_i) &\geq R_{min}^M \text{ for those } \mathcal{F}_i \text{ that have no vertex in } S_4 \end{aligned} \quad (4)$$

In (4), we denote by $\mathcal{V}$ the set of vertexes for which $\mathcal{V}_k \in S_4$, by $\mathcal{V}^{inf} \subset S_4$ the set of vertexes for which $C_o \mathcal{V}_k^{inf} \leq C_o P_2$, $\mathcal{V}_k^{inf} \in \mathcal{V}^{inf}$, and by $\mathcal{V}^{sup} \subset S_4$ the set of vertexes for which $C_o \mathcal{V}_k^{sup} \geq R_{min}$, $\mathcal{V}_k^{sup} \in \mathcal{V}^{sup}$. $d(C_o \mathcal{F}_i)$ is the distance

²In the particular case when P_1 , P_2 and P_3 are collinear, the third condition from (3) reduces to $\cos(\alpha_2) \geq \frac{(l+2dl)}{ED}$

from the circle center C_o to the line segment $\mathcal{F}_i$, and:

$$\begin{aligned} R_{min}^m &= \sqrt{\left(\frac{l}{2} - \frac{L}{\sin(\phi_{max})}\right)^2 - L^2 - l - dl} \\ R_{min}^M &= \sqrt{\left(\frac{l}{2} - \frac{L}{\sin(\phi_{max})} + dl\right)^2 + dL^2} \end{aligned}$$

Until now, we developed conditions for testing if the area swapped by the robot remains inside $q_i \cup q_j$. If it does, the robot clearly avoids all regions from $\Pi \setminus \{h(q_i) \cup h(q_j)\}$ while moving from q_i to q_j . If it does not, we use conditions similar to (3), (4) for checking intersections between the area spanned by the robot (i.e. $S_1 \cup S_2 \cup S_3$) with any region from $\Pi \setminus \{h(q_i) \cup h(q_j)\}$. Whenever such an intersection exists, the current sequence from the total number of $N_{ctrl}^{i,j}$ tested sequences may lead to violations of the LTL formula, and therefore $N_{feas}^{i,j}$ is not increased in such situations. In all other cases, $N_{feas}^{i,j}$ is increased. $N_{feas}^{i,j}$ is initialized with 0 and after all $N_{ctrl}^{i,j}$ situations were tested, we end up with $N_{feas}^{i,j} \leq N_{ctrl}^{i,j}$. The equality $N_{feas}^{i,j} = N_{ctrl}^{i,j}$ happening only when there is no case such that the movement from q_i to q_j yields intersections with regions from $\Pi \setminus \{h(q_i) \cup h(q_j)\}$. Now, we construct the probability $\mathcal{P}(q_i, q_j)$ as in equation (5), based on $N_{ctrl}^{i,j}$, $N_{feas}^{i,j}$ and $N_{feas}^{i,j}$. As suggested by (5), $\mathcal{P}(q_i, q_j)$ depends on the percentage of cases when a smooth trajectory can be constructed between q_i and q_j . This percentage is lowered based on the difference $N_{ctrl}^{i,j} - N_{feas}^{i,j}$, which represents the number of cases when the actual movement may lead to overlaps with other regions of interest.

$$\mathcal{P}(q_i, q_j) = \begin{cases} \left(\frac{N_{feas}^{i,j}}{N_{ctrl}^{i,j}}\right)^{(1+2 \cdot (N_{ctrl}^{i,j} - N_{feas}^{i,j}))} & \text{if } q_j \in \delta(q_i) \\ 1, & \text{if } q_i = q_j \\ 0, & \text{if } q_j \notin \delta(q_j) \end{cases} \quad (5)$$

B. Finding a run of T

For finding a run of T satisfying the LTL formula ϕ , we construct the product of T with the Büchi automaton corresponding to ϕ . This product is an adaptation from [2], [4] of the product automata constructed for deterministic transitions systems T .

Definition 3: (Probabilistic Product automaton) The product automaton $\mathcal{A}_{\mathcal{P}} = T \times B$ between the Markovian transition system $T = (Q, Q_0, \mathcal{P}, \Pi, h)$ and the Büchi automaton $B = (S, S_0, \Pi, \rightarrow_B, F)$ is defined as the tuple $\mathcal{A}_{\mathcal{P}} = (S_{\mathcal{A}}, S_{\mathcal{A}0}, \mathcal{L}_{\mathcal{A}}, F_{\mathcal{A}})$, where:

- $S_{\mathcal{A}} = Q \times S$ is the set of states,
- $S_{\mathcal{A}0} = Q_0 \times S$ is the set of initial states,
- $\rightarrow_{\mathcal{A}} \subseteq S_{\mathcal{A}0} \times S_{\mathcal{A}0}$ is the transition relation, with $((q_i, s_j), (q_k, s_l)) \in \rightarrow_{\mathcal{A}}$ if and only if $\mathcal{P}(q_i, q_k) > 0$ and $(s_j, h(q_i), s_l) \in \rightarrow_B$,
- $\mathcal{L}_{\mathcal{A}} : S_{\mathcal{A}} \times S_{\mathcal{A}} \rightarrow [0 + \infty)$ defines the transition costs, with $\mathcal{L}_{\mathcal{A}}((q_i, s_j), (q_k, s_l)) = -\log(\mathcal{P}(q_i, q_k))$ if $\mathcal{P}(q_i, q_k) > 0$, and $\mathcal{L}_{\mathcal{A}}((q_i, s_j), (q_k, s_l)) = +\infty$ otherwise,
- $F_{\mathcal{A}} = Q \times F$ is the set of accepting (final) states.

Informally, the product $\mathcal{A}_{\mathcal{P}}$ maintains possible transitions from T that yield feasible observations for transitions of

B. The acceptance condition for $\mathcal{A}_{\mathcal{P}}$ is similar with the acceptance condition for Büchi automata (see Section II-A), and it requires finding an infinite run $r_{\mathcal{A}_{\mathcal{P}}}$ that infinitely often visits set $F_{\mathcal{A}}$. Now we aim in finding an optimal accepted run in a prefix-suffix form, which we denote by $r_{\mathcal{A}_{\mathcal{P}}} = \text{pref}(r_{\mathcal{A}_{\mathcal{P}}})\text{suff}(r_{\mathcal{A}_{\mathcal{P}}})\text{suff}(r_{\mathcal{A}_{\mathcal{P}}})\dots$. The projection of $r_{\mathcal{A}_{\mathcal{P}}}$ to transitions of T satisfies the LTL formula and the optimality of $r_{\mathcal{A}_{\mathcal{P}}}$ means that the run of T should have the maximum probability of being followed. Since the probability of following a run is the product of all its transition probabilities, we define a weighting function $\mathcal{L}_{\mathcal{A}}$ on transitions of $\mathcal{A}_{\mathcal{P}}$ as follows: $\mathcal{L}_{\mathcal{A}} : S_{\mathcal{A}} \times S_{\mathcal{A}} \rightarrow [0 + \infty)$, with $\mathcal{L}_{\mathcal{A}}((q_i, s_j), (q_k, s_l)) = -\log(\mathcal{P}(q_i, q_k))$ if $\mathcal{P}(q_i, q_k) > 0$, and $\mathcal{L}_{\mathcal{A}}((q_i, s_j), (q_k, s_l)) = +\infty$ otherwise. This weighting function allows us to use a standard Dijkstra search algorithm (minimizing the sum of weights), simply because maximizing the product of probabilities means maximizing the logarithm of this product, which equals the sum of probability logarithms.

After the weighting function is created, the optimum run $r_{\mathcal{A}_{\mathcal{P}}}$ is found by first searching an optimal prefix to each final state, then searching an optimal suffixes for any final state, and finally choosing the pair prefix-suffix with a minimum cost (meaning that the probability of following prefix and one iteration of suffix is maximized).

C. Algorithm *PLTLCon.Car.Like*

Algorithm *PLTLCon.Car.Like* iteratively searches for an optimum run of T satisfying formula ϕ , and based on this run it tries to construct a smooth path that the car-like robot can follow.

Algorithm 1 *PLTLCon.Car.Like*

```

1: Perform cell decomposition and obtain  $Q$ 
2: Construct  $T$ , Büchi automaton  $B_{\phi}$  for  $\phi$ , and  $\mathcal{A}_{\mathcal{P}}$ 
3:  $W = \emptyset$ 
4: while TRUE do
5:   Find run  $\text{run}_i(\phi)$  of  $\mathcal{A}_{\mathcal{P}}$  that maximizes the probability of satisfying the formula
6:   if  $\text{run}_i(\phi) \cap W \neq \emptyset$  then
7:      $nr_{\text{run}_i} = nr_{\text{run}_i} + 1$ 
8:     if  $nr_{\text{run}_i} \geq 4$  then
9:       break ▷ Problem is unfeasible
10:    end if
11:    continue ▷ Goto line 4
12:  end if
13:  if  $|W| \geq VaL$  then
14:    break ▷ Problem is unfeasible
15:  end if
16:  Construct the angular and the smooth path for current run  $\text{run}_i(\phi)$ 
17:  if The smooth path was not created then
18:    Modify  $\mathcal{P}$  and  $\mathcal{A}_{\mathcal{P}}$ 
19:    Add element  $W_i = \{\text{run}_i(\phi), 1\}$  to  $W$ 
20:  end if
21: end while

```

Once the optimum run of T is obtained as a straightforward projection of $r_{\mathcal{A}_{\mathcal{P}}}$ to states of T , we construct an angular path and a smooth path corresponding to this run, as in [8]. However, in some cases it is possible that a smooth path cannot be constructed for the angular path corresponding to the found run (see explanations on $N_{ctrl}^{i,j}$ versus $N^{i,j}$ in section III-A). When such a case is encountered, we add the current run to the set W . Also, we modify (lower) $\mathcal{P}$ for cells preventing the construction on smooth path, by penalizing the corresponding transitions with the hope that successive runs will avoid them. Each element of set W is of form $W_i = \{\text{run}_i(\phi), nr_{\text{run}_i}\}$ where nr_{run_i} is the number of times when Algorithm *PLTLCon.Car.Like* returns $\text{run}_i(\phi)$. The algorithm iterates until cardinality of set W becomes larger than an imposed value VaL , or until $nr_{\text{run}_i} \geq 4$ for some element i of W . The condition $nr_{\text{run}_i} \geq 4$ stops the algorithm when the same run is returned for a number of times (even though such a run has the maximum probability, it cannot be converted into a smooth path). The condition limiting the cardinality of W stops the algorithm when many runs with maximum probability do not yield feasible paths, and thus the Algorithm *PLTLCon.Car.Like* ends in a finite number of steps.

IV. EXAMPLE

In order to illustrate the usability of our algorithm, we consider the example depicted in Fig. 3, where a rectangular environment with size $20 \times 24m$ is cluttered with 4 regions of interest $\Pi = \bigcup \pi_i$, $i = \{1, 2, 3, 4\}$. The considered car-like robot has parameters (see Fig 1): $l = 0.21m$, $L = 0.26m$, $dl = dL = 0.1m$ and $\phi_{max} = 45^\circ$, and the initial deployment of the point P_m is $(x, y) = (2, 3)m$.

By partitioning the environment with a triangular cell decomposition (as in Fig. 3(a)), an abstraction T with 30 states is obtained. If a polytopal decomposition is used (as in Fig. 3(b)), a transition system with 63 states results.

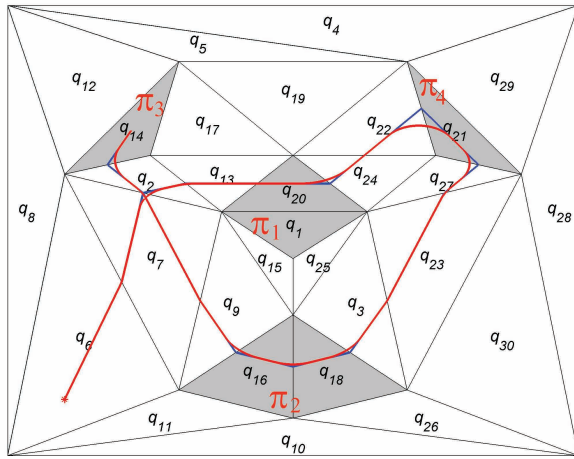
We impose the motion specification “Reach region π_1 and then reach π_4 , π_2 and π_3 while always avoiding π_1 , from current position”, which can be expressed as the following LTL formula: $\phi = \Diamond(\pi_1 \wedge \Diamond(\Box \neg \pi_1 \wedge \Diamond(\pi_2 \wedge \Diamond(\pi_3))))$.

For the triangular decomposition our algorithm returns the following run that has probability 1: $q_6q_7q_2q_{13}q_{20}q_{24}q_{22}q_{21}q_{27}q_{23}q_{31}q_{16}q_{9}q_7q_2(q_{14})$, where (q_{14}) means to infinitely repeat this element. For the case of polytopal cell decomposition, the optimum run has probability 0.8 of satisfying ϕ and it has the following structure: $q_{46}q_{57}q_{58}q_{54}q_6q_{14}q_{41}q_{40}q_{11}q_{10}q_{21}q_{23}q_{24}q_6q_{50}q_{48}q_{53}q_{44}q_{45}q_{67}q_{65}(q_{64})$.

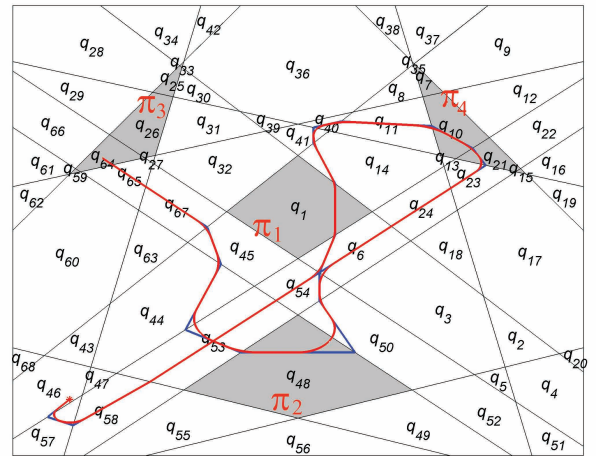
Assuming that the initial orientation of the robots chassis is chosen in line with the first segment of the angular path, while the final orientation is not of interest to us, the above runs can be converted to the smooth trajectories depicted with red in Fig. 3(a) and 3(b).

V. CONCLUSIONS

We presented a fully computational framework for probabilistic automatic deployment of a car-like mobile robot from



(a) Triangular cell decomposition



(b) Polytopal cell decomposition

Fig. 3. Simulation results considering a rectangular environment cluttered with 4 regions of interest.

temporal logic specifications over regions of interest from environment. Although not complete, the approach returns a sequence of cells from a partitioned environment that maximizes the probability of designing a feasible smooth path for the car-like robot. The solution combines cell decomposition techniques, car-like planning methods for environments cluttered with obstacles, and algorithms dealing with temporal logic formulae for mobile robots. The current work extends the classical navigation problem for a car-like robot by allowing rich, human-like specifications and by providing a probabilistic solution to such complex problems.

VI. ACKNOWLEDGMENTS

The first author acknowledges the support of the EU-RODOC (Doctoral Scholarships for research performance at European level) project, financed by the European Social Fund and the Romanian Government. The second author acknowledges the support of the CNCS-UEFISCDI project PN-II-RU PD code 333/2010.

REFERENCES

- [1] E. M. Clarke, O. Grumberg, and D. A. Peled, "Model Checking," The MIT Press, 2000.
- [2] G. E. Fainekos, H. Kress-Gazit, and G. J. Pappas, "Hybrid controllers for path planning: a temporal logic approach," *In IEEE Conference on Decision and Control*, pages 4885-4890, Seville, Spain, 2005.
- [3] C. Belta, A. Bicchi, M. Egerstedt, E. Frazzoli, E. Klavins, and G. J. Pappas, "Symbolic Planning and Control of Robot Motion," *IEEE Robotics and Automation Magazine - special issue on Grand Challenges of Robotics*, vol. 14, no. 1, pp. 61-71, 2007.
- [4] M. Kloetzer, and C. Belta, "A fully automated framework for control of linear systems from temporal logic specifications," *IEEE Transactions on Automatic Control*, 53(1): pp. 287-297, 2008.
- [5] G. E. Fainekos, A. Girard, H. Kress-Gazit, G. J. Pappas, "Temporal logic motion planning for dynamic robots," *Automatica* 45, pp 343-352, 2009.
- [6] D. C. Conner, A. A. Rizzi, and H. Choset, "Integrated Planning and Control for Convexbodied Nonholonomic systems using Local Feedback Control Policies," *In: Robotics: Science and Systems II*, Philadelphia, PA, 2006.
- [7] D. C. Conner, H. Kress-Gazit, H. Choset, A. A. Rizzi, and G. J. Pappas, "Valet Parking Without a Valet," *Proc. of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, USA, pp. 572-577, Oct 29 - Nov 2 2007.
- [8] N. Ghita, and M. Kloetzer, "Motion planning and controlling a car-like robot in an environment Cluttered with static obstacles," *Buletinul Institutului Politehnic din Iasi*, Tome LVI (LX), Fasc. 3, Automatic Control and Computer Science section, pp. 81-94, 2010.
- [9] M. Lahijanian, S. B. Andersson, and C. Belta, "A Probabilistic Approach for Control of a Stochastic System from LTL Specifications," *Joint48thIEEEConferenceonDecisionandControland 28thChineseControlConference*, Shanghai, P.R.China, December 16-18, 2009.
- [10] M. Lahijanian, J. Wasniewski, S. B. Andersson, and C. Belta, "Motion Planning and Control from Temporal Logic Specifications with Probabilistic Satisfaction Guarantees," *2010IEEEInternationalConferenceonRoboticsandAutomation*, Anchorage, Alaska, USA, May 3-8, 2010.
- [11] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*, Springer Verlag, pp. 9-62, 799-827, 2008.
- [12] F. Lamiraux and J. P. Lamond, "Smooth motion planning for Car-Like Vehicles," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 498-502, August 2001.
- [13] A. Scheuer, and Th. Fraichard, "Continuous-Curvature Path Planning for Multiple Car-Like Vehicles," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, September, pp. 8-12, Grenoble, France, 1997.
- [14] F. Lamiraux, E. Ferré, and E. Vallée, "Kinodynamic motion planning: Connecting exploration trees using trajectory optimization methods," *In IEEE International Conference on Robotics and Automation*, volume 4, pages 3987-3992, New Orleans, LA, USA, May 2004.
- [15] L. E. Dubins, "On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents," *Amer. J. Math.*, vol. 79, pp. 497-516, 1957.
- [16] J. A. Reeds and R. A. Shepp, "Optimal paths for a car that goes both forward and backward," *Pacific J. Math.*, vol. 145, no. 2, pp. 367-393, 1990.
- [17] P. Wolper, M. Vardi and A. Sistla, "Reasoning about infinite computation paths," *in Proceedings of the 24th IEEE Symposium on Foundations of Computer Science*, E. N. et al., Ed., Tucson, AZ, pp. 185-194, 1983.
- [18] P. Gastin and D. Oddoux, "Fast LTL to Büchi automata translation," *in Proceedings of the 13th Conference on Computer Aided Verification (CAV01)*, ser. Lecture Notes in Computer Science, H. C. G. Berry and A. Finkel, Eds., no. 2102. Springer, pp. 53-65, 2001.
- [19] M. Kloetzer and N. Ghita, "Software tool for constructing cell decompositions," *IEEE Conference on Automation Science and Engineering*, Trieste, Italy, 2011 (accepted).

On the Real-time Modelling of a Robotic Scene Perception and Estimation System

Sorin M. Grigorescu, *Member, IEEE*, Gigel Macesanu, Tiberiu T. Cocias and Florin Moldoveanu *Member, IEEE*

Abstract—In this paper a real-time machine vision system for robotic scene perception and estimation is proposed. Its goal is to determine the 3D structure of the imaged scene together with the relative position and orientation of the robot with respect to the environment. Mainly, the vision system is composed of two basic elements: a 3D camera-scene depth computation method and a camera egomotion estimator. The image processing operations have been implemented into a sequential machine vision pipeline. A performance evaluation of the proposed approach is given through experimental results within an indoor environment.

I. INTRODUCTION

One of the main components of a robotic application is represented by the robot's machine vision system. In the robotics community, it has been noticed over the past decade how the vision algorithms evolved from their classical 2D visual servoing approaches to complex 3D visualization systems that perceive both the 3D structure of the environment, as well as the relative pose (*position and orientation*) of the robot with respect to the imaged scene [1]. The computation of the robot's pose over time is determined relative to the robot's camera pose, a process also known as *egomotion estimation*.

In 3D scene perception, there are commonly two types of vision sensors used for providing visual information data: *stereo vision cameras* and *range sensors* such as laser scanners or 3D *Time-of-Flight* (ToF) cameras [2]. In the process of stereo vision based 3D perception and egomotion estimation, the stereo correspondence problem has to be solved, i.e. the corresponding feature points, necessary for 3D reconstruction, have to be extracted from both stereo images [3]. In contrast, stereo vision range sensing devices provide direct capturing of 3D scenes, delivering a pure stereo depth image in form of 3D point clouds [4]. In the case of range sensors, the obtained depth information can have different error values, depending on the sensed surface. This phenomenon makes stereo vision a more reliable solution for autonomous robotic systems that operate in real world environments.

This work was supported by the Sectoral Operational Program Human Resources Development (SOP HRD).

The authors are with the Department of Automation, Transilvania University of Braşov, Mihai Viteazu 5, 500174, Braşov, Romania {s.grigorescu, gigel.macesanu, tiberiu.cocias, moldof}@unitbv.ro

Source code for the proposed scene perception system can be found at rovis.unitbv.ro

Camera motion estimation, or egomotion, has been studied within the *Simultaneous Localization and Mapping* (SLAM) context [5]. Using detected visual information, motion estimation techniques can provide a very precise egomotion of the robot. In SLAM also, the basic sensor used is the stereo camera [6], [7]. The main operation involved in stereo based robotic perception is the computation of the so-called correspondence points used for calculating the 3D pose of the robot's camera. Correspondence calculation is also known as *feature matching*, a process which, in the case of 3D sensing, has to be applied on the stereo image pairs and also on consecutive sequence of stereo images [8]. The most common features used in this context are points localized through corner detectors such as *Harris* [9]. Another well-known technique for feature detection is the *Scale Invariant Feature Transform* (SIFT) [10], in which the computed features are invariant to rotation, scaling, translation and partially to changes in illumination. Other approaches are based on the extraction of lines or edges [7], [11] which have the advantage to provide geometrical information about the environment, but are computational expensive. Based on the extracted features, the robot's motion can be extracted with the help of estimators such as the *Kalman* [12] or *Particle Filter* [13].

In this paper the authors propose a real-time dual egomotion and scene estimation system for the purpose of reliable 3D robotic perception. The novelty of the presented research work lies on the developed 3D perception pipeline used to fuse color stereo images with the computed egomotion of the camera, as well as on its usage to sense cluttered indoor robotic environments.

The rest of the paper is organized as follows. In Section II the theory behind 3D depth sensation is presented, followed in Section III by the egomotion estimation description. The implementation of the 3D perception pipeline is given in Section IV together with performance evaluation results. Finally, conclusions and outlook are presented in Section V.

II. MECHANICS OF DEPTH SENSATION

The 3D depth estimation approach used in this paper is based on a triangulation algorithm between the left and right images of a stereo camera and an imaged real world 3D point [14]. The principle of the method is illustrated in Fig. 1.

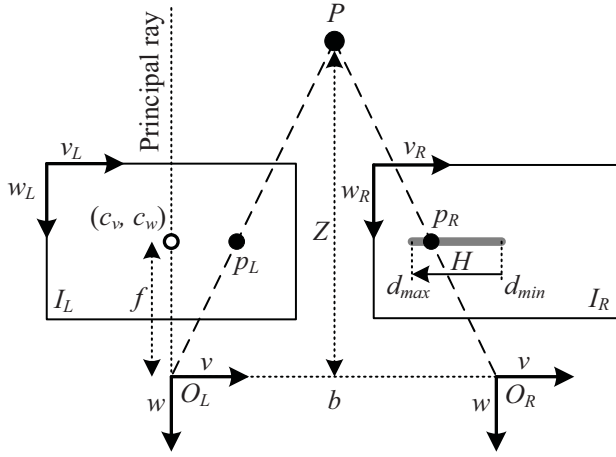


Fig. 1. 3D position estimation of a point P using a pair of rectified stereo images.

A. Stereo Camera Configuration

In order to calculate the 3D position of a point, its corresponding 2D projections onto the left and right image planes of the stereo camera have to be determined [14]. The projection can only be calculated if the stereo camera is calibrated, namely if its *intrinsic* and *extrinsic parameters* are known. These parameters are obtained through the camera calibration process which calculates on one hand the internal, or intrinsic, camera parameters K , such as the focal length f , optical centers (c_v, c_w) of both sensors, aspect ratio and skew constant and, on the other hand, the external, or extrinsic parameters, represented by the *rotation* and *translation* $[R, t]$ of each sensor of the stereo camera with respect to a fixed world reference point. Once these parameters are known, they can be used to compute the projection matrices $Q_L, Q_R \in \mathbb{R}^{3 \times 4}$ of each sensor, in the following homogeneous matrix multiplication form:

$$Q = \begin{bmatrix} f & 0 & c_v & 0 \\ 0 & f & c_w & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} r_{00} & r_{01} & r_{02} & t_x \\ r_{10} & r_{11} & r_{12} & t_y \\ r_{20} & r_{21} & r_{22} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = K \cdot [R|t], \quad (1)$$

where K is the intrinsic camera matrix. In the proposed system, the pose of the right sensor is always calculated with respect to the left sensor. Having in mind that the two are parallel to each other and on the same x axis, the pose of the right sensor is obtained as a pure translation with respect to the left one, namely:

$$[R_R|t_R] = [I|[b, 0, 0]^T], \quad (2)$$

where I is the identity matrix. The position of the right sensor is given by its translation over the x axis of the Cartesian space by a coefficient b known as the *baseline* between the optical centers of the two sensors.

The triangulation method can be efficiently used only if the projections of the object points on the image planes

are known, that is, only if the correspondence of each feature point is known in both the left and right images. This procedure, also encountered under the name of *feature matching*, determines for each real world 3D point $P = [X \ Y \ Z \ 1]^T$, its corresponding projections onto the left and right image planes of the stereo camera:

$$\begin{cases} p_L = \begin{bmatrix} v_L & w_L & 1 \end{bmatrix}^T, \\ p_R = \begin{bmatrix} v_R & w_R & 1 \end{bmatrix}^T, \end{cases} \quad (3)$$

where (v, w) are pixels within the 2D image plane. As can be seen from Fig. 1, the 2D image positions p_L and p_R are projected as the intersection of the image planes with the line connecting point P in world coordinates with the optical centers O_L and O_R of the stereo camera. The *principal plane*, that is the image, is located at the focal distance f from the optical centre of the sensor. The origin of the image coordinate system is considered as the top-left image corner (v_0, w_0) .

B. Disparity Computation

Having in mind the calculated stereo camera parameters and points correspondences, the goal of depth computation is to determine the camera-scene distance Z . The output of the depth estimation procedure is a depth map $f(x, y, z)$ containing *voxels* representing the estimated 3D positions of the matched feature points in meters.

As already explained, the feature matching procedure determines for each pixel in the left image $I_L(v, w)$ its corresponding point in the right image $I_R(v, w)$. A straightforward approach to feature matching is to compare each pixel in $I_L(v, w)$ with each pixel $I_R(v, w)$. In this paper, a pair of pixels $\{p_L, p_R\}$ are considered corresponding points if their normalized cross correlation is the highest.

However, having in mind the high computational complexity (e.g. for a pair of typical 1024x768px images a number of app. 6^{11} calculations have to be performed), this approach is not feasible for a real-time 3D perception system. The problem has been solved through the usage of *rectified stereo images* which have the property that the pixels on the v axes of $I_L(v, w)$ and $I_R(v, w)$ are parallel to each other. Image rectification is obtained via the projection matrices Q_L, Q_R and the epipolar geometry constraint [14].

The above simplification allows the corresponding search to be performed only along the v axis of the image plane, usually on a predefined interval $H = [d_{min}, d_{max}]$, also known as the *horopter*. Hence, the 3D position estimation of a point is obtained using the so-called *disparity computation*:

$$X = (v_L - c_v) \cdot \frac{b}{d}, \quad (4)$$

$$Y = (w_L - c_w) \cdot \frac{b}{d}, \quad (5)$$

$$Z = f \cdot \frac{b}{d}, \quad (6)$$

where d is the disparity of the projected point P :

$$d = v_L - v_R. \quad (7)$$

One of the most popular disparity computation algorithm, also used in this work, is the so-called *Block Matching* (BM) [3] approach. In order to obtain a 3D model of the imaged scene, each calculated depth map $f(x, y, z)$ has to be associated with its corresponding camera pose, as will be further explained.

III. CAMERA EGOMOTION ESTIMATION

Self-localization is one of the main requirements in the field of autonomous robotics. This process, known as *egomotion estimation*, is normally accomplished by estimating the pose of the robot's stereo camera while it moves through space. In this paper, we have considered the problem of egomotion estimation through stereo vision only, as illustrated in Fig. 2.

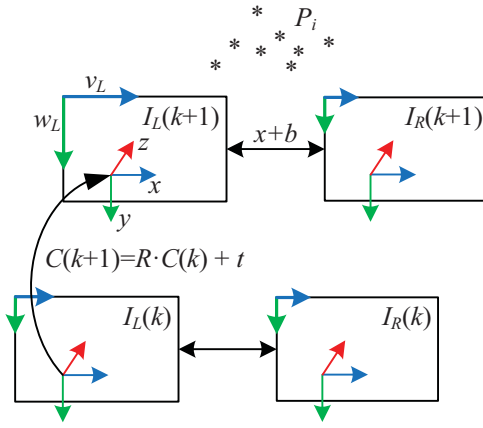


Fig. 2. Stereo vision based egomotion estimation principle.

The goal of egomotion estimation is to determine the new camera pose $C(k+1)$ with respect to the previous one $C(k)$ given a set of matched 3D points P_i and the intrinsic and extrinsic camera matrices. The P_i set of 3D points are needed to establish the *homography*, or *perspective transformation*, between the two neighboring views of the camera, that is, $C(k)$ and $C(k+1)$. A set of minimum 4 points are needed to determine the homography between two views [14]. Since only a small number of points are needed for calculating the homography, there is no need to obtain a dense, computation expensive, depth map such as the one obtained via the BM procedure explained in the previous section. On the other hand, in order to obtain a precise reconstruction of the camera's egomotion, the corresponding projected 2D points have to be precisely detected in the stereo images.

In the proposed system, a Harris corner detector [9] feature matching algorithm is used to compute from P_i the following corresponding pairs:

- a set of corresponding 2D points $\{p_{Li}(k), p_{Ri}(k)\}$ from the left and right images at camera pose $C(k)$;

- the reconstructed 3D positions $P_{RCi}(k)$ of points P_i from $\{p_{Li}(k), p_{Ri}(k)\}$ and the camera's intrinsic and extrinsic information;
- the corresponding 2D points $p_{Li}(k+1)$ in the new camera pose $C(k+1)$ for every matched 2D point $p_{Li}(k)$ in the left image of the vision sensor.

The above mentioned pairs are used in a *Perspective-N-Point* estimation manner for obtaining the new camera pose $C(k+1)$ as follows. The reconstructed 3D points $P_{RCi}(k)$, determined from the stereo pair $C(k)$ are reprojected onto the left image of $C(k+1)$. Knowing, with respect to the left image, the 2D corresponding points $p_{Li}(k) \rightarrow p_{Li}(k+1)$, a reprojection error can be calculated as:

$$E_d(p_{Li}, \hat{p}_{RCi}) = \sqrt{(v_{Li} - \hat{v}_{RCi})^2 + (w_{Li} - \hat{w}_{RCi})^2}, \quad (8)$$

where $\hat{p}_{RCi}$ are the image reprojected points from $P_{RCi}(k)$ and $E_d(\cdot)$ is the Euclidean distance. The reprojection error from Eq. 8 is further minimized through a Gauss-Newton optimization procedure, which adapts the new camera's rotation and translation $[R(k+1)|t(k+1)]$. The new pose of the camera is thus given by the values of $[R(k+1)|t(k+1)]$ which minimize Eq. 8. Finally, the new pose is calculated as:

$$C(k+1) = R(k+1) \cdot C(k) + t(k+1). \quad (9)$$

IV. REAL-TIME 3D MACHINE VISION PIPELINE AND EXPERIMENTAL RESULTS

The proposed 3D perception approach has been implemented as the image processing pipeline from Fig. 3. The first operation on the pipeline is the image acquisition one, which reads color images from the stereo camera. Once the left and right image pair is available, two distinct processing threads are started. The first thread computes the depth image, which contains dense camera-object distances, while the second thread deals with feature matching and the calculation of the camera's egomotion. When both the camera's pose $[R|t]$ and the depth map $f(x, y, z)$ are available, they are annotated to the virtual 3D model of the imaged scene. The 3D coordinates of the depth map are rendered with respect to the computed camera pose.

We have considered a processing cycle to begin with image acquisition and end with camera and depth annotation to the virtual 3D model, as depicted in Fig. 3. In the experimental setup, a mechanically calibrated Point Grey Bumblebee[®] stereo camera system has been used to acquire a sequence of 519 indoor images. The average computational time needed by the considered processing operations is shown on the timeline from Fig. 3. As can be seen, a processing cycle lasts just over 700ms, the value being low enough to consider the proposed system as a real-time one. The implemented image processing program has been tested on a typical portable computer running a 64 bits operating system on an Intel[®] i3 dual core CPU, each processor having a 2.40GHz clock speed.

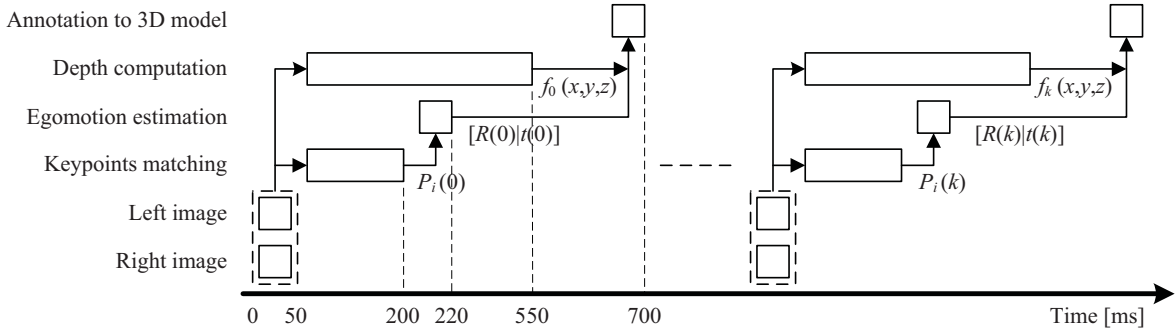


Fig. 3. Computational pipeline of the proposed 3D perception system.

An snapshot of a processing slice can be seen in Fig. 4. One of the main operations in egomotion estimation is the calculation of the reprojection error, shown in Fig. 4(a). The reprojection error is represented as the blue line linking the correspondences between the $C(k)$ and $C(k+1)$ left images (yellow circles) and the 3D reprojected points from $C(k)$ onto the $C(k+1)$ left image (red circles). An example of egomotion estimation and depth map annotation over a sequence of 15 images is presented in Fig. 4(b). The rendering of the annotated 3D virtual model is performed on top of an OpenGL[®] render.

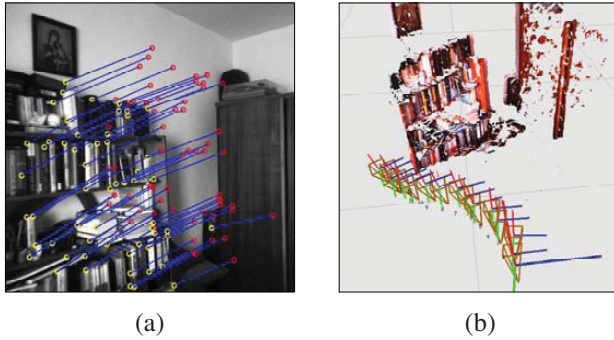


Fig. 4. Slices taken from the 3D perception pipeline (best viewed in color). (a) The reprojection error (8) over a typical indoor scene. (b) Camera egomotion and scene structure annotation over a sequence of 15 stereo images.

V. CONCLUSIONS AND OUTLOOK

In the presented paper, a real-time 3D perception system has been proposed. Its main elements are the depth sensing algorithm together with the camera egomotion estimator. The goal of the proposed system is to be used on autonomous robotics platforms that need a real-time 3D machine vision component. The system has been implemented as a real-time processing pipeline, one full operational cycle being computed in just over 700ms, thus meeting the real-time requirements.

As future work, the authors consider the speed enhancement of the proposed system using state of the art parallel processing equipment, such as FPGAs and GPUs which, in the authors opinion, would exponentially decrease the processing time (e.g. feature matching for the left and right

images, as well as matching for neighboring camera poses would be done in parallel). Also, a real-time feedback optimization technique of the proposed perception system is considered for coping with uncertain situations that can occur in the imaged environment (e.g. variable illumination, moving objects, occluded areas etc.).

ACKNOWLEDGMENT

This paper is supported by the Sectoral Operational Program Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the projects POSDRU/89/1.5/S/59323, POSDRU/88/1.5/S/59321 and POSDRU/107/1.5/S/76945.

REFERENCES

- [1] A. Andreopoulos, S. Hasler, H. Wersing, H. Janssen, J. Tsotsos, and E. Koerner, "Active 3D Object Localization Using A Humanoid Robot," *IEEE Trans. on Robotics*, vol. 27, no. 1, pp. 47–64, 2011.
- [2] S. Hussmann and T. Liepert, "Robot Vision System based on a 3D-TOF Camera," in *Instrumentation and Measurement Technology Conference-IMTC 2007*, (Warsaw, Poland), 2007.
- [3] M. Brown, D. Burschka, and G. Hager, "Advances in Computational Stereo," *IEEE Trans. on Pattern Recognition and Machine Intelligence*, vol. 25, no. 8, pp. 993–1008, 2003.
- [4] R. Rusu, Z. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D Point Cloud based Object Maps for Household Environments," *Robotics and Autonomous Systems*, vol. 56, pp. 927–941, 2008.
- [5] J. Neira, A. Davison, and J. Leonard, "Special Issue on Visual SLAM," *IEEE Trans. on Robotics*, vol. 24, no. 5, 2008.
- [6] L. Paz, P. Pinies, J. Tardos, and J. Neira, "Large-Scale 6-DOF SLAM With Stereo-in-Hand," *IEEE Trans. on Robotics*, vol. 24, no. 5, pp. 946–957, 2008.
- [7] T. Lemaire, C. Berger, I. Jung, and S. Lacroix, "Vision-Based SLAM: Stereo and Monocular Approaches," *Int. Journal of Computer Vision*, vol. 74, no. 3, pp. 343–364, 2007.
- [8] A. Geiger, J. Ziegler, and C. Stiller, "StereoScan: Dense 3D Reconstruction in Real-time," in *IEEE Intelligent Vehicles Symposium*, (Baden-Baden, Germany), June 2011.
- [9] C. Harris and M. Stephens, "A Combined Corner and Edge Detection," in *Proc. of the Fourth Alvey Vision Conference*, pp. 147–151, 1988.
- [10] D. G. Lowe, "Object Recognition from Local Scale-invariant Features," in *Int. Conf. on Computer Vision*, (Corfu, Greece), pp. 1150–1157, Sept. 1999.
- [11] E. Eade and T. Drummond, "Edge Landmarks in Monocular SLAM," in *British Machine Vision Conference-BMVC 2006*, (Edinburgh, U.K.), pp. 469–476, 2006.
- [12] G. Welch and G. Bishop, *An Introduction to the Kalman Filter*. UNC-Chapel Hill, 2006.
- [13] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Trans. on Signal Processing*, vol. 50, no. 2, 2002.
- [14] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

Needle-Stability Maps for Brain-Tumor Biopsies

Paul Hergehelegiu, Vasile Manta, and Eduard Gröller, *Member, IEEE Computer Society*

Abstract—Minimally invasive surgical procedures like biopsies require the insertion of a needle to reach a specific tissue from where samples are taken. This process is affected by a number of errors that can lead to a different needle pathway than originally planned. In the case of brain-tumor biopsies, these errors can be caused by the brain shifting when the skull is drilled, measurement errors due to the physical framework that supports the biopsy needle or some other human errors. To provide support for preoperative planning, we introduce the concept of a biopsy-needle stability-map. This map provides information on how stable a biopsy-needle pathway is. We also introduce an algorithm for finding the reachable points on the tumor border from one specific entry point. A technique for automatically selecting the needle pathway that offers the biggest minimal distance to any blood vessel is also presented. A 3D representation of the regions of interest (tumor, blood vessels) together with the needle pathway and the reachable points of the tumor has also been implemented. The resulting stability map visually represents, using color-coding, how close to any blood vessel the pathways surrounding the main one are.

I. INTRODUCTION

BIOPSIES are a common medical procedure mainly used for a better pathology diagnosis. They represent a medical test that involves collecting tissue from a living human body structure using a biopsy needle. When dealing with brain tumors, a biopsy represents a mandatory step before more advanced treatment plans are developed. The tests run on the samples taken by biopsies will determine the type of the tumor, if a resection should be considered, if the tumor growth can be stopped by gamma rays or, if it is filled with fluid, which properties this fluid does have. The first step the physician has to perform after scanning the patient is to establish a possible needle pathway. MRI (Magnetic Resonance Imaging) is the most commonly used scanning technique for this type of pathologies as it offers high contrast images for soft tissues. For defining the biopsy needle pathway, the currently used procedures involve the manual inspection of the MRI slices for establishing the ending and the entry point of the needle. This process requires a sensitive manual adjustment of the needle

pathway so that it will not hit any blood vessel. After completing these laborious steps, the used medical framework will provide the position data that accurately describes the needle pathway. These data have to be manually correlated with the physical framework that supports the needle insertion process. The described process is prone to human errors as well as measurement errors. Our work helps the physician by computing the best needle pathway and provides a visual representation of the pathway stability with respect to specific errors.

In order to better understand the 3D shape of a structure or for locating it more easily, a segmentation algorithm that separates regions of interest in the volume is usually implemented. We use the brain-tumor segmentation for computing the possible ending points reachable by the biopsy needle in an automated manner. The segmented cerebral vasculature will provide the input data for computing the distances from any point in the volume to the closest blood vessel. For this, we use a distance-transform algorithm that produces a volume of the same size as the original one, where the voxel values represent the distances to the closest blood vessel.

The lack of a 3D representation of the tumor or of the needle direction further increases the possibility that errors might occur in the process of defining the pathway. A visual representation of how stable this pathway is with respect to the errors that can occur (measurements errors, computational errors or human errors) can help the physicians in understanding the risks of not reaching the ending point of the biopsy needle. We introduce a visual representation for the stability of the pathway, called biopsy-needle stability-map. Our algorithm also creates a hierarchy of the possible needle pathways that provides an automatic computation and visualization of the best pathway. The quality of the pathways is given by the minimum distance to any surrounding blood vessel, considering all the points of the pathway.

This paper is organized as follows. Section II provides an overview of existing techniques in the field of tumor segmentation, distance transform computation, and stability maps. Section III contains the detailed description of the used segmentation algorithm. Section IV presents the technique used for computing the distance transform. Section V describes the approach used for finding the reachable tumor points. Section VI presents the concept of the biopsy-needle stability-map. Section VII concludes the paper.

Manuscript received June 17, 2011.

P. Hergehelegiu is with the Faculty of Automatic Control and Computer Engineering, Technical University "Gheorghe Asachi" Iasi, Str. Prof. dr. doc. Dimitrie Mangeron, 27, Iasi, 700050, Romania. E-mail: phergehelegiu@cs.tuiasi.ro.

V. Manta is with the Faculty of Automatic Control and Computer Engineering, Technical University "Gheorghe Asachi" Iasi, Str. Prof. dr. doc. Dimitrie Mangeron, 27, Iasi, 700050, Romania. E-mail: vmanta@cs.tuiasi.ro.

E. Gröller is with the Vienna University of Technology, Institute of Computer Graphics and Algorithms, Favoritenstr. 9/5, Stock, A-1040 Vienna, Austria. E-mail: groeller@cg.tuwien.ac.at.

II. RELATED WORK

The computation of a distance transform (DT) represents an important part of many computer vision or image processing applications. A DT algorithm applied on a binary image will provide as output another image of the same size. Each pixel represents the distance to the region of interest (non-zero pixel). The DT is an important operator in shape analysis [1], with numerous applications in research areas like separation of overlapping objects in microscopic images [2], computing morphological operations (erosion, dilation) [3], extracting the medial axis (skeletonisation) [4], segmentation of structures in medical images [5], analysis of three-dimensional interactions between biological structures [6].

Stability maps are commonly introduced in applications for providing visual support and guidelines for human decision making. They do not aim to replace the humans in the decision making. Stability maps are used in various research fields such as visualizing classified structures prone to large computational errors coming for example from an automatic classification algorithm [7]. Such a visualization system focuses the user onto the structures that require manual correction. Another application is the stability analysis of 3DCT scan positions, guiding the user on how to position objects during scanning. The goal is to reduce scanning artifacts [8].

Various methods for simulating the insertion of a biopsy needle were developed. A technique for simulating the insertion in deformable tissues was introduced by Alterovitz et al. [9]. They compute the error as Euclidian distance, and only consider the error that result from inserting the needle. Also a stability measurement for the simulation process is presented. A needle-path planning for breast biopsies was developed [10]. The errors considered are due to inserting the needle through soft tissue. This approach is not suitable in our situation with the brain tumor. Brain tissue behaves differently from muscular tissue. Thus the needle insertion will not change the position of the tumor and of the blood vessels.

III. TUMOR SEGMENTATION

In order to extract useful information from volumetric data, segmentation methods had to be implemented. The segmentation algorithm classifies the voxels of a volume as belonging to either a region of interest, or to less important background data. We have segmented the brain tumor using the watershed algorithm [11]. The algorithm uses the variation of the gradient to classify voxels. It provides good results in segmenting homogeneous regions that have well defined border areas (ridges). A brain tumor in MRI T1 weighted sequence has such properties. The gradient input image is treated as a height function, with the assumption that higher intensity values in the image indicate border areas. The watershed algorithm works by “flooding” this height field and by assigning an unique value for each lower intensity area.

Two parameters control the output of the segmentation algorithm: threshold and level. Threshold sets the absolute minimum height value used in the segmenting process. In other words, it discards the voxels with values below the threshold. A high value for the threshold will lead to a segmentation result with fewer regions. The level parameter controls the depth of the flooding. A value of 1.0 will provide the result of flooding the entire image resulting only one segmented region. A value of 0.0 will produce a basic segmentation, with a lot of oversegmented areas. Fig. 1 shows the result of applying the watershed algorithm on an MRI T1 weighted contrast enhanced volume. The used threshold is 0.06 and the level value is 0.5. These values worked for all tested MRI T1 weighted volumes.

As the result of the watershed algorithm is represented by various regions for areas bounded by ridges, an additional operation for selecting the tumor points is needed. We used a simple interval threshold filter to discard all the points that are not belonging to the region of interest, outputting only the tumor mask. As the watershed algorithm assigns different values for each segmented region, the simple inspection of the segmented volume is needed for selecting the value corresponding to the tumor. The approach of using the watershed algorithm followed by an interval threshold filter is suited when dealing with multiple brain tumors in the same dataset. Multiple brain tumors are commonly encountered in metastatic diseases.

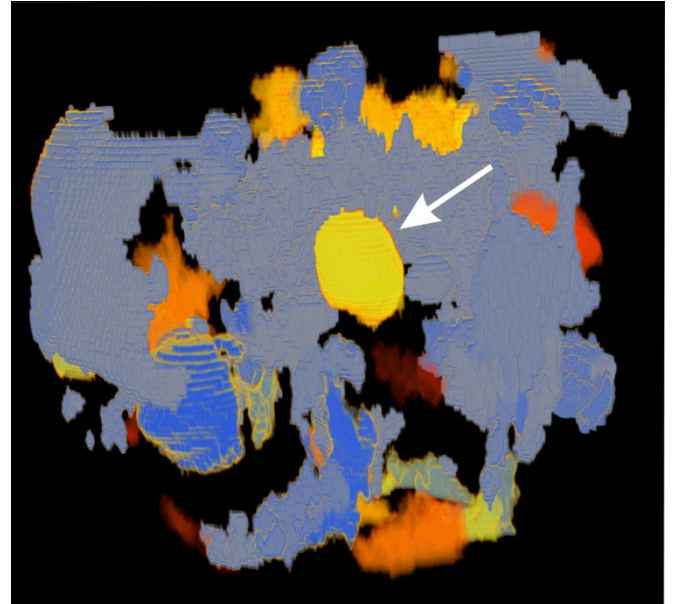


Fig. 1. 3D view of the watershed segmentation algorithm applied for segmenting a brain tumor. The arrow is pointing to the tumor.

IV. DISTANCE TRANSFORM

Computing a DT was mainly done in for 2D [1], but it can be easily extended to 3D. In the 3D case, the DT determines the distance of each voxel of the volume to a given subset of it, in our case the blood vessels. The blood vessel mask was obtained by a manual segmentation. Let $Vol: \Omega \subset \mathbb{Z}^3 \rightarrow$

$\{0,1\}$ be a binary volume where the Ω domain is convex. An object $\mathcal{O}$ will be represented by all the white voxels of the volume:

$$\mathcal{O} = \{p \in \Omega \mid Vol(p) = 1\}$$

The voxels of $\mathcal{O}$ are referred as foreground, and its compliment, $\mathcal{O}^c$ is called background and consists of black voxels of Ω . The distance map of volume Vol (or the distance map of $\mathcal{O}$) is given by the following definition:

$$D(p) = \min\{d(p, q) \mid q \in \mathcal{O}^c\}$$

where $d(p, q)$ is the distance between p and q , usually taken as the Euclidian distance.

In this paper, we used a distance transform algorithm introduced by Felzenszwalb [12]. We choose this algorithm because it is easy to reproduce and provides satisfactory speed and results. When dealing with non-isotropic voxels, as usually encountered in medical imaging techniques, a correction of the Euclidian distance computation has to be applied. In our dataset, the pixel spacing on x, y, z axes are 0.54mm, 0.54mm and respectively 1.5mm. Computations performed on the z axis had to be corrected by multiplying the result with $1.5/0.54=2.77$. Fig. 2 presents a slice of the DT volume. The darker areas correspond to areas closer to a blood vessel while the brighter areas denote a bigger distance to any surrounding blood vessel.

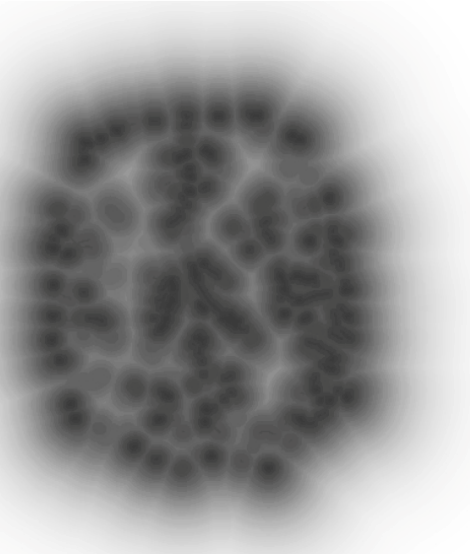


Fig. 2. Slice of the DT volume computed from the blood vessels mask. A darker region indicates the presence of a blood vessel in the neighboring area.

V. TUMOR ENDING POINTS

We define the reachable points of the biopsy needle as the tumor points that are reachable from one fixed entry point and the pathways to these points have the property of hitting directly and only the tumor reachable point. We use the segmented tumor data as input for computing the reachable

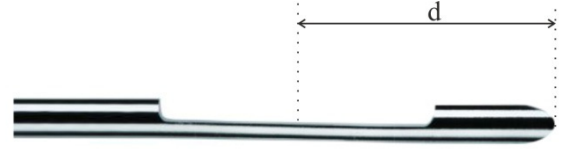


Fig. 3. Tip of a biopsy needle. The gap can be opened or closed by an inner layer controlled by the physician.

points of the biopsy needle. We propose a straightforward and easy to implement algorithm for finding the reachable tumor points for one fixed entry point. Because the biopsy needle does not take tissue samples from the tip and features an opening at one side close to the tip (see Fig. 3), the computed reachable points need to have specific characteristics to fit the physical model of the needle.

The automated computation of the reachable points helps the physician in seeing what tumor points can be reached by the biopsy needle, for one specified entry point. The position of the blood vessels does not influence this algorithm, as the computation of all reachable tumor points is desired. The pseudocode for computing the possible reachable points for the biopsy needle is provided as Algorithm 1.

Algorithm 1 DetermineReachablePoints

```

For the entry point EP
  For every tumor point TP
    If (The previous voxel of TP on the line
        given by EP and TP is a non-tumor point)
      AND
      (The next n voxels after TP on the line
        given by EP and TP are tumor points)
    Then
      Add TP to reachable tumor points set

```

The number n is given by the physical specification of the needle, namely the distance from the tip of the biopsy needle to the middle of the sample taking gap (distance d in Fig. 3). The computed reachable tumor points from one specific entry point EP are presented in Fig. 4 as a white mesh covering the tumor.

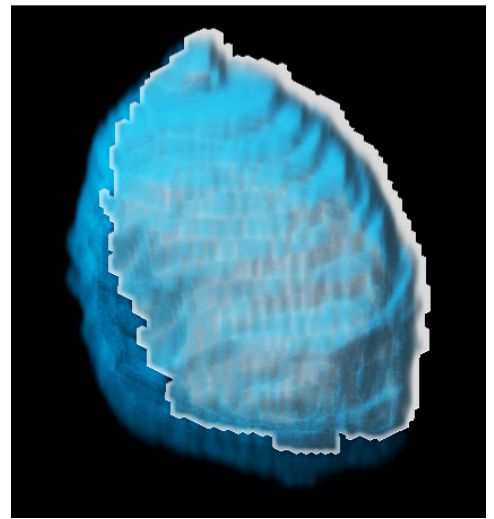


Fig 4. Reachable tumor points (white) from one specific entry point (side view). The tumor points are colored in blue.

VI. BIOPSY-NEEDLE STABILITY-MAP

For creating the stability map, we first create a disk D_s in the normal plane to the needle pathway (NP), with the centre in the chosen reachable tumor point (RTP). This disk projected along the needle direction will serve as a proxy object for the stability map.

An example of one needle pathway starting from the entry point EP, along with the disk created in the normal plane of NP with the center in one chosen RTP is shown in Fig. 5.

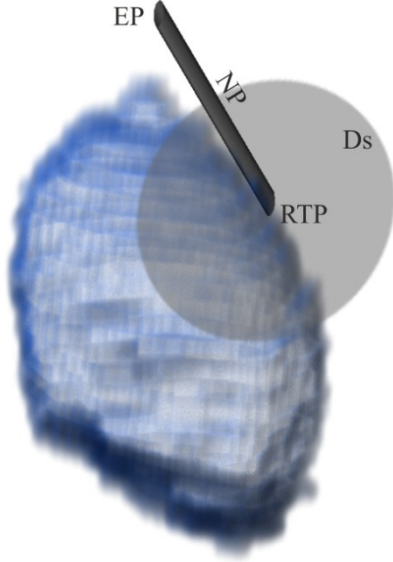


Fig 5. A disk D_s created in the normal plane of one biopsy needle pathway NP, with the centre in the considered reachable tumor point RTP.

We consider the cost of each needle pathway given by the minimum distance from any point belonging to the pathway to any blood vessel, using the previously calculated distance transform. As presented in Section IV, the DT volume stores the distances to the closest blood vessel for each voxel of the volume. For determining the cost of which rays will be represented in the stability map, we first determine the needle pathways that will correspond to the points of the stability-map. The directions of these pathways are determined by the entry point EP and by the voxels of the disk D_s . Depending on the disk orientation and position, the pathway might hit the tumor in a point situated in front or in back of the disk. In some cases, it is also possible that the pathway will not hit the tumor. The pseudocode for assigning a color to each point of the stability map is given by Algorithm 2.

Algorithm 2 CreateStabilityMap:

```

For each voxel  $V \in$  disk  $D_s$ 
  Try to find a tumor point (TP) on the
  direction defined by  $V$  and EP
  If (TP was found as belonging to the tumor)
    Then
      Compute the cost of the pathway defined by
      EP and TP
      Assign a color according to the cost of
      the pathway

```

```

Draw the corresponding points on the
stability map
Else
  Draw the corresponding points on the
  stability map in white, as the
  corresponding needle pathway is not
  reaching the tumor

```

According to the computed cost of each considered pathway, a color coding scheme was implemented (Fig. 6). White indicates areas where the direction given by the entry point and the corresponding disk point does not intersect the tumor. The green color indicates that the corresponding direction has a large distance to any blood vessel. The red color indicates the region where the direction is hitting a blood vessel. In particular case represented in Fig. 6, even if the considered pathway ensures a large distance to any blood vessels, the physician might opt for another needle pathway, as the biopsy needle might not hit the tumor (areas in white) if any error will occur. By sorting the directions according to the cost described above, we can represent in an automated manner the pathway with the best cost. The physician can then inspect the stability-map of the pathway and decide if it suits the medical requests.

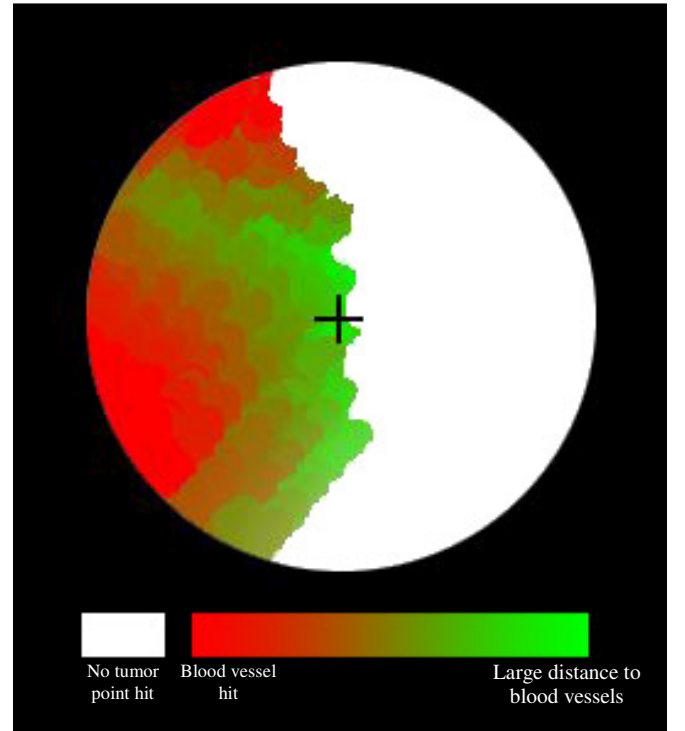


Fig 6. Biopsy-needle stability-map for a needle pathway. Green regions indicate a needle direction that is further away from a blood vessel and red regions indicates that the pathway is hitting a blood vessel. White areas indicate that the tumor is not hit.

The presented work was implemented as a VolumeShop [13] plugin, with a 3D volume renderer implemented in GLSL, an OpenGL geometry renderer and a stability-map editor implemented using Qt libraries.

VII. CONCLUSIONS

In this paper, we presented an application for creating a stability map for the needle in brain tumor biopsies. We presented the algorithms used for segmenting the brain tumor and for computing the distance transform of the blood vessels volume. The created stability map guides the physician in visualizing the needle pathways that are most stable. If errors happen in the needle insertion process, the patient will not sustain damage due to punctuated blood vessels. The introduced stability map also provides information on which needle pathways close to the considered one will not hit the tumor. The application can also suggest a needle pathway in an automated manner according to the minimum distance from any point belonging to the pathway to any blood vessel. The work described in this paper received good feedbacks from the neurosurgeons it was presented to. Future work will include integration into current medical frameworks, so the users will not need to learn two interfaces.

ACKNOWLEDGMENT

The authors would like to thank members of the Institute of Computer Graphics and Algorithms from the Vienna University of Technology for help in the implementation phase.

REFERENCES

- [1] R. Fabbri, L. da F. Costa, J. C. Torelli, and O. M. Bruno, "2D Euclidean distance transform algorithms: a comparative survey", *ACM Computing Surveys*, vol 40(1), article 2, 2008.
- [2] O. Cuisenaire, B. Andmacq, "Fast Euclidean distance transformation by propagation using multiple neighborhoods". *Comput. Vis. Image Under.*, vol. 76(2), pp. 163–172, 1999.
- [3] O. Cuisenaire, "Locally adaptable mathematical morphology using distance transformations". *Patt. Recog.* vol. 39(3), pp. 405–416, 2006.
- [4] A. V. Saude, M. Couprie, and R. A. Lotuffo, "Exact Euclidean medial axis in higher resolution". *Discrete Geometry for Computer Imagery. Lecture Notes in Computer Science*. Springer, pp. 605–616, 2006.
- [5] T. Sebastian, J. Crisco, and B. Kimia, "Segmentation of carpal bones from CT images using skeletally coupled deformable models". *Med. Image Anal.* 7, vol 1, pp. 21–45, 2003.
- [6] B. A. Travencolo, C. Martinez Debat, M.E. Beletti, J. R., Sotelo Silveira, R. Ehrlich, and L. F. Costa, "A new method for quantifying three-dimensional interactions between biological structures". *J. Anat.* vol 210(2), pp. 221–231, 2007.
- [7] E. Frontoni, A. M. Bernardini, E. S. Malinverni, A. Mancini, P. Zingaretti, "Stability maps for really exploitable automatic classification results", *17th International Conference on Geoinformatics*, pp 1–6, 2009.
- [8] A. Amirkhanov, C. Heinzl, M. Reiter, M. E. Gröller, "Visual Optimality and Stability Analysis of 3DCT Scan Positions", *IEEE Transactions on Visualization and Computer Graphics*, vol 16(6), pp 1477–1487, 2010.
- [9] R. Alterovitz, K. Y. Goldberg, J. Pouliot, I. C. Hsu, "Sensorless Motion Planning for Medical Needle Insertion in Deformable Tissues", *IEEE Transactions on Information Technology in Biomedicine*, vol 13(2), pp 217 – 225, 2009
- [10] L. Vancamberg, A. Sahbani, S. Muller, and G. Morel, "Needle path planning for digital breast tomosynthesis biopsy", *International Conference on Robotics and Automation - ICRA*, pp. 2062-2067, 2010.
- [11] F. Meyer, "Topographic distance and watershed lines," *Signal Processing*, vol. 38, pp. 113-125, 1994.
- [12] P. Felzenszwalb, and D. Huttenlocher, "Distance transforms of sampled functions," *Cornell University CIS*, Tech. Rep. 2004-1963, 2004.
- [13] S. Bruckner, and M. E. Gröller, "VolumeShop: An interactive system for direct volume illustration", *In Proceedings of IEEE Visualization 2005*, pp 671–678, 2005.

The mathematical model of a biologically inspired electronic neuron for ease the design of spiking neural networks topology

Mircea Gh. Hulea

Abstract—This paper presents the mathematical model of a biologically inspired neuron which was previously tested and implemented in hardware. This neuron mimics the spiking nature of natural neurons and adds the learning mechanisms of biological synapses. The goal of the mathematical model elaboration is to ease the process of developing the topology of analogue spiking neural networks using computer simulations while keeping the computation time low.

I. INTRODUCTION

SPIKING neurons are the most plausible models of biological neurons because they accurately mimic the natural mechanisms of information processing and learning. First category of processes includes temporal integration of the incoming excitatory and inhibitory stimulation, membrane potential, activation threshold and refractory period, while the second category of processes which gives the learning ability to the neural network are the short-term potentiation and long-term potentiation of the synapses.

The processing and learning mechanisms were modeled in hardware using analogue circuits which are combined into a neuron model introduced by the papers [1], [2] and presented in detail by a Romanian patent [3]. One of the advantages of spiking neurons analogue implementation is that they are multiplexed in space obtaining the real time operation of the entire neural network. However, the disadvantage of this approach in implementing the artificial neural networks is the difficulty in developing network topology due to practical issues. A common method to ease the process of elaborating the network synaptic configuration is to use software simulations of the network behavior starting from the mathematical model of a single neuron. The state of the art in

this field started with the conventional Spike-Response-Model (SRM) which describes mathematically in a simple way the variation of the membrane potential during action potential and refractory period [4], [5] and [6]. Other mathematical models of biologically inspired neurons are optimized for real time response of the neural network [7], [8] and [9] or for biological plausibility such as Izhikevich models [10] and [11]. A simple but biologically plausible rule for modeling the synaptic mechanisms of learning (SAPR) is presented in [12] and [13]. More complex models of neurons which mimic rigorously the synaptic chemical and physical processes were developed and simulated by Henry Markram *et. al* in the framework of the Blue Brain Project [14], [15] and [16].

The mathematical model presented in this paper has the main advantage that it describes the biological mechanisms of associative learning in a computationally effective way while being suitable for hardware implementation. Also, the presentation of the mathematical model includes several aspects for improving the hardware version of the neuron model.

II. ARTIFICIAL NEURON MODEL

The elaboration of the mathematical model which describes the behavior of the electronic neuron started from the physical laws which govern the passive and semiconductor electronic components. Depending on the modeled functions of the natural neurons, this artificial neuron is divided into three modules which simulate the main information processing elements, the associative learning properties and the output generation.

A. Information processing

The behavioral characteristics of the biological neuron modeled by the electronic neuron are the input stimuli integration, the membrane potential, the neuron activation threshold and the refractory period. The input voltage level of the artificial neuron which models the equilibrium potential of the postsynaptic membrane is denoted with V_M . In this paper the positive spikes refers to voltage impulses whose maximum

Manuscript received on July 1, 2011. This work was supported by the project PERFORM-ERA "Postdoctoral Performance for Integration in the European Research Area" (ID-57649), financed by the European Social Fund and the Romanian Government.

M. Gh. Hulea is with the Technical University Gheorghe Asachi of Iasi, Bd. Dimitrie Mangeron, no. 67, P.O. 700050, Iasi, Romania, phone: +40744824115; e-mail: mhulea@tuiasi.ro.

amplitude is above the V_M while the negative spikes refers to voltage impulses whose minimum amplitude is below V_M . As shown in figure 1, the neuron uses for the integration of the positive or negative spikes a capacitor C_I which gives the artificial neuron input potential, V_I . During the neuron idle state the C_I charging and discharging currents keep V_I to the equilibrium potential V_M which is lower than T_1 base-emitter voltage, V_{BE} . Thus, $I_{RS} = \frac{V_M - V_{REF}}{R_M}$ and the equilibrium potential is:

$$V_M = I_{RS} R_M + V_{REF}, V_M < V_{BE} \quad (1)$$

If n_E represents the number of active excitatory synapses that stimulates simultaneously the neuron and n_I is the number of inhibitory synapses which fulfill the same condition, the variation of the potential V_I given by the excitatory or inhibitory stimuli is:

$$V_I = \begin{cases} V_{I_0} + (V_{OE} - V_{I_0}) \cdot \left(1 - e^{-\frac{n_E t}{R_E C_I}}\right), n_E \neq 0 \wedge n_I = 0 \\ (V_{I_0} - V_{OI}) \cdot e^{-\frac{n_I t}{R_I C_I}}, n_E = 0 \wedge n_I \neq 0 \end{cases} \quad (2)$$

where V_{I_0} is the initial voltage stored by C_I during the stimulus onset, while V_{OE} and V_{OI} are the peak amplitudes of the impulses generated by the excitatory and respectively, inhibitory synapses. In this paper circuit node potential or capacitor voltage at the moment $t=0$ of a charging or discharging event are denoted $V_{[X]_0}$ where $[X]$ represents the letter associated with the capacitor or circuit node.

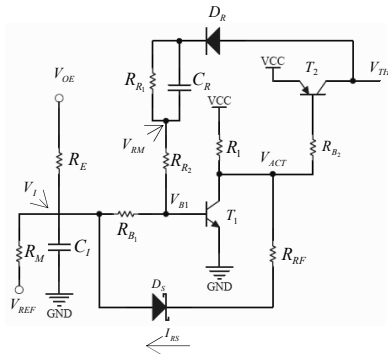


Fig. 1. Input processing module

When $n_E \neq 0$ and $n_I \neq 0$, C_I is charged or discharged by the potential $V_{DIV} = (V_{OE} - V_{OI}) \cdot \frac{n_E R_I}{R_E n_I + R_I n_E}$ as:

$$V_I = \begin{cases} V_{I_0} + (V_{DIV} - V_{I_0}) \cdot \left(1 - e^{-\frac{n_E t}{R_E C_I}}\right), V_{DIV} > V_{I_0} \\ V_{I_0} + (V_{DIV} - V_{I_0}) \cdot \left(e^{-\frac{n_I t}{R_I C_I}}\right), V_{DIV} < V_{I_0} \end{cases} \quad (3)$$

During neuron idle state while $n_E = n_I = 0$ the input potential V_I varies as:

$$V_I = \begin{cases} V_{I_0} + \frac{V_M - V_{I_0}}{R_{RS}} \left(1 - e^{-\frac{t}{R_{RS} C_I}}\right), V_M > V_{I_0} \\ V_{I_0} + \frac{V_M - V_{I_0}}{R_M} \left(e^{-\frac{t}{R_M C_I}}\right), V_M < V_{I_0} \end{cases} \quad (4)$$

Neuron activation takes place when $V_I > V_{BE}$ which opens T_1 and T_2 . In this state the input potential which mimics the postsynaptic membrane potential varies as follows:

$$V_I = V_{DS} + (V_{I_0} - V_{DS}) \cdot \left(e^{-\frac{t}{R_{RS} C_I}}\right) \quad (5)$$

$$V_{CR} = (V_{TH} - V_F - V_{BE}) \cdot \left(e^{-\frac{t}{R_{R2} C_R}}\right) \quad (6)$$

where $V_{RM} = V_{TH} - V_F - V_{CR} > V_{BE}$. Because in this case T_1 is saturated $I_{R_{R2}} \approx I_{BE1}$. One condition for artificial neuron proper operation is $V_I < V_{BE}$ when $V_{RM} \leq V_{BE}$.

After biological neuron activation follows an absolute refractory period while the neuron is not able to fire. For modeling this characteristic of the natural neuron the input potential of the electronic neuron must remain under the neuron activation threshold even if T_1 is opened or V_{OE} has the maximum value. This condition implies:

$$R_1 \beta_1 \cdot \frac{V_{CC} - V_{BE1}}{R_p + R_{B1}} > V_{CC} - V_{EB2} \quad (7)$$

where $R_p = \frac{R_I \cdot (R_1 + R_{DS} + R_F)}{R_I + R_1 + R_{DS} + R_F}$, β_1 is the T_1 current gain and:

$$\frac{R_I}{n_{E \max}} > \frac{(V_R R_{B1} - V_{BE} R_e)(R_{RF} + R_e) - V_{OE} R_e R_{RF} (R_{B1} - R_e)}{(R_{B1} - R_e)(V_R R_{RF} + V_{DS} R_e) - V_R R_{B1} + V_{BE} R_e} \quad (8)$$

where $R_e = R_{R1} + R_{R2} + R_{B1}$, $V_R = V_{CC} - V_F$ and V_{DS} is the forward voltage of the Schottky diode. Condition (7) shows that the current through T_1 base should be lower than the current which produces T_2 opening. Condition (8) computes minimum R_I which prevents the saturation state stability of T_1 and T_2 .

The lowest value reached by the input potential at the beginning of the refractory period is the shottky diode forward voltage V_{DS} . During this period the input potential returns to the equilibrium state following equation (9).

$$V_I = V_{I_0} + V_{REF} \left(1 - e^{-\frac{t}{R_M C_I}} \right) + \frac{I_{RS} \cdot t}{C_I} \quad (9)$$

The capacitor C_I is recharged during refractory period by reference voltage and by reverse current of D_S .

B. Learning

The main advantage of the artificial neuron which was taken into consideration for developing the mathematical model is the implementation of the natural mechanisms of learning using only passive and semiconductor electronic components. Depending on the membrane role in synaptic communication the learning mechanisms are split into two categories – presynaptic and postsynaptic. The presynaptic elements of learning include *posttetanic potentiation* (PTP) which increases the synaptic weight by adjusting the transmission power in presynaptic membrane as a consequence of repeated neuron activation. The postsynaptic mechanisms of learning are the *short term potentiation* (STP) and the *long term potentiation* (LTP) of the synapses which increases the synaptic weights if the postsynaptic neuron fires shortly after the presynaptic neuron activation. Therefore, a spiking neural network which has this neuron as the processing unit is trained in an associative manner satisfying the Hebbian learning rule.

The training process of the electronic network implies the variation of capacitor charge which models the synaptic weights. Figure 2 shows the electrical circuit of the artificial synapse learning mechanism together with the control elements of this module. The key component of the learning module is the capacity C which stores the synaptic weight. The weight of the artificial synapse is denoted by V_C which equals V_L during neuron idle state and $V_L - V_{CC}$ if the neuron is active.

From the biological point of view, the postsynaptic elements of learning – STP and LTP – bring a significantly

higher contribution to the synaptic potentiation than the PTP which is the presynaptic element of learning.

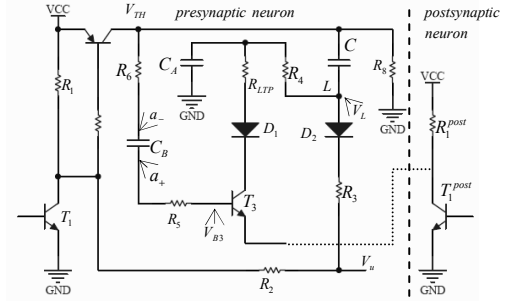


Fig. 2. Learning module

1) *Short term potentiation* includes two components which are the sudden increase in the synaptic weight followed by slow efficiency decay. Because during neuron activation the synapse is potentiated by the first component of the STP and by PTP, the synaptic weight variation is described by:

$$V_L = V_{L_0} - (V_{L_0} - V_{CA}) \cdot e^{-\frac{t}{R_4 C}} - (V_{L_0} - V_F) \cdot e^{-\frac{t}{(R_1 + R_2 + R_3)C}} \quad (10)$$

from where we can compute: $V_C = V_L - V_{TH}$.

During the efficiency decay which begins after neuron inactivation the weight decrease to the value V_{L_0} which was stored in C before STP. This process which happens immediately after neuron transition from active to idle state is explained using the circuit branch from figure 3.

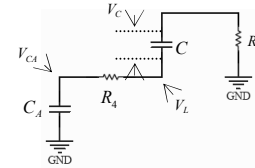


Fig. 3. The electronic components for STP decay

The capacitor C charge variation is defined as:

$$V_C = V_{C_0} + (V_{CA} - V_{C_0}) \cdot \left(1 - e^{-\frac{-t}{(R_4 + R_8)C}} \right) \quad (11)$$

$$V_L = V_C + (V_{CA} - V_C) \cdot R_8 / (R_4 + R_8) \quad (12)$$

2) *Long term potentiation* stops the slow decay of the synaptic weights which makes the synaptic efficiency to remain at its current values until the subsequent neuron activation. For the electronic neuron the LTP imply the equalization of the charges in capacitors C and C_A from figure

2 by sudden discharge of C_A using T_3 . Therefore, while $V_{B3} > V_{BE}$ the potential V_{CA} varies as:

$$V_{CA} = (V_{CA_0} - V_F) \cdot e^{-\frac{t}{R_5 C_A}} \quad (13)$$

Denoting by t_{LTP} the LTP duration and by t_{ED} the STP efficiency decay, we have $t_{LTP} \ll t_{ED}$ because LTP is a sudden process. If $R_4 \gg R_{LTP}$ then $I_{R4} \ll I_{R_{LTP}}$ which makes possible the approximation $I_{R4} \approx 0$. This means that the capacitor C charge variation during LTP, $\Delta V_C \approx 0$.

Therefore, the LTP discharges the capacitor C_A while $V_{CA} > V_C$ which is equivalent with $V_{TH} > 0$. In order to obtain $V_{B3} > V_{BE}$ while $V_{TH} > 0$, between the negative plate a_- and the positive plate a_+ of the capacitor C_B is established a potential that compensate V_{BE} of the transistor T_3 . Thus, whether $V_{a-} - V_{a+} \approx -V_{BE}$ and $V_{C_B} \approx const$ during the capacitor C_A discharging through T_3 , the condition $V_{TH} > 0$ will keep T_3 opened. Considering that $I_{R4} \approx 0$ we have $V_{TH} = V_{CA} - V_C$. This condition implies that the LTP is produced while:

$$V_{CA} \geq V_C + I_{BE_3} \cdot (R_5 + R_6) \quad (14)$$

where $V_{CA} = V_F + (V_{CA_0} - V_F) \cdot e^{-\frac{t}{R_{LTP} C_A}}$ and I_{BE_3} is the minimum base current which keeps T_3 opened. The capacity C_B is reloaded to the value $-V_{BE}$ during neuron activation using the circuit from figure 4.

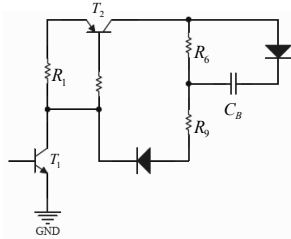


Fig. 4. Charging circuit for compensation capacity C_B

When the transistors T_1 and T_2 are saturated the maximum potential between C_B plates will be:

$$V_{CB_{max}} = V_{cc} \cdot R_6 / (R_6 + R_9) + V_F \cdot R_9 / (R_6 + R_9) \quad (15)$$

which gives the variation with time of the capacitor C_B potential:

$$V_{CB} = V_{CB_{max}} \cdot \left(1 - e^{-\frac{t}{R_9 C_B}} \right) \quad (16)$$

There is scientific evidence that in some conditions the efficiencies of the biological synapses decreases generating *long term depression* (LTD) [17]. As was observed in several experiments done on biological neurons *in vitro* the synaptic efficiencies decrease if the postsynaptic neuron action potential precedes the presynaptic neuron activation in a limited time window [18], [19] and [20]. However, this behavior was not evident in experiments done on biological neural networks *in vivo* [21].

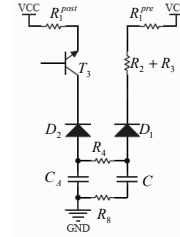


Fig. 5. Long-term depression mechanism

Therefore, because not all the LTD mechanisms are known, for the electronic synapses considered in this paper it is supposed that during neuron idle state the synaptic efficiencies are decreased at significantly lower rate than the PTP rate. This property is modeled by the electronic sub-circuit presented in figure 5.

The variation of capacitor C potential is given by the equations which are valid during neuron idle state:

$$V_C = V_{CC} \left(1 - e^{-\frac{t}{R_C C}} \right) \quad (17)$$

where: $R_C = R_4' \parallel R_{e2} + R_8$; $R_4' = R_4 \left(1 + \frac{V_{CC} - V_{CA}}{V_{CC}} \right)$;

$$V_{CA} = V_{CC} \left(1 - e^{-\frac{t}{R_{e1} C_A}} \right);$$

$$R_{e1} = R_1^{post} + R_{T3} + R_{D2} \text{ and } R_{e2} = R_1^{pre} + R_2 + R_3 + R_{D1}.$$

The equivalent resistance which gives the reverse current of diodes D_1 and D_2 is denoted by R_{D1} and respectively, R_{D2} . Also, the resistance of emitter-collector junction of T_3 is denoted by R_{T3} . The capacitor C is charged continuously during inactive period of the artificial neuron. When the neuron fires the capacitor is discharged by STP and PTP mechanisms.

C. Output generation

The output module of the electronic neuron mimics the presynaptic activity of the biological synapses. Considering that the neural message transmitted inside the natural synapses is coded by frequency and power of the excitatory and inhibitory stimuli, the artificial neuron outputs positive and negative electrical impulses. The energy of these impulses is variable and depends on the synaptic weight stored by the learning module in capacitor C .

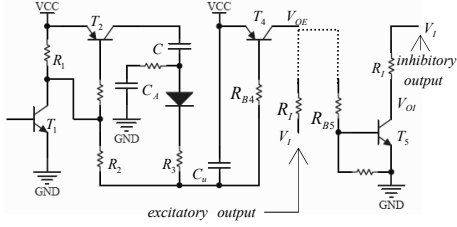


Fig. 6. Spike generation module

In order to evaluate the duration of the spike produced by collector of transistor T_4 it is necessary to determine the potential variation in T_4 base.

Therefore, the minimum potential V_{u_0} which discharges the capacitor C_u is $V_{u_0} = (V_L - V_F) \cdot \frac{R_2}{R_2 + R_3}$. If $V_{u_{min}} < V_{CC} - V_{EB4}$ then T_4 is opened and V_{OE} equals V_{CC} . Depending on the neuron state, the charge variation of C_u follows two rules:

- 1) *Active state:* $V_u = V_{u_0} + (V_{CC} - V_{u_0}) \cdot e^{-\frac{t}{R_2 C_u}}$;
- 2) *Inactive state:* $V_u = V_{u_0} + (V_{CC} - V_{u_0}) \cdot \left(1 - e^{-\frac{t}{(R_1 + R_2) C_u}}\right)$.

If $V_u > V_{CC} - V_{EB4}$, V_{OE} is open collector.

III. RESULTS

The mathematical model elaboration started from the physical laws which describe the behavior of the electronic neuron circuit. For each of the three modules of the electronic neuron there are simulated in MATLAB the variables which describes the main characteristics of the biological neuron. These variables whose expressions are detailed in the corresponding sections of this work are briefly presented in table I. The simulation of the several of these variables is presented in the sequel.

Thus, figure 7 (a) presents the input potential variation before neuron activation focusing on the temporal integration of the incoming stimulation. The signal diagram on the left shows the potential for the electronic neuron and on the right the variable V_I variation. Figure 7 (b) focuses also on the

neuron activation and on the refractory period. Note that the mathematical model variables mimic rigorously the electronic neuron behavior.

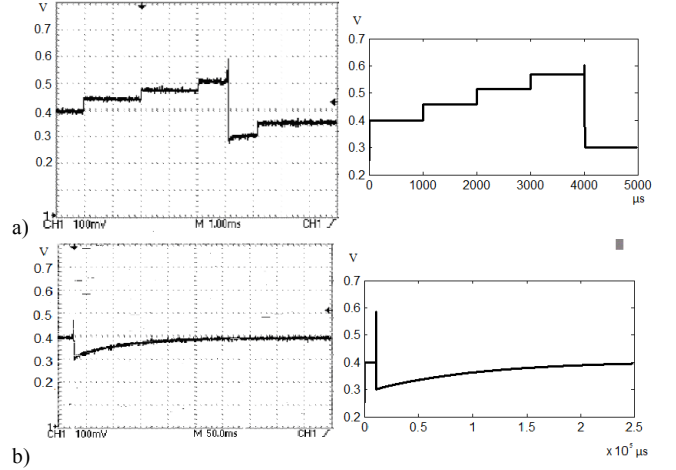


Fig. 7. Electronic neuron operation (left) and mathematical model simulation (right) for: a) temporal integration of the incoming stimulation; b) action potential and refractory period.

The learning mechanisms of the electronic neuron which are simulated in software are presented in figure 8 (a) and (b). To easy understand the mathematical model behavior during synaptic potentiation there are presented the signals which models the input (left) and output (right) of the neuron. During neuron activation the synapse is potentiated by STP and immediately after neuron inactivation the potential is decreased to the pre-firing value. This is demonstrated by variable V_c shown in figure 8 (b) the diagram on the right. Note that due to electronic neuron design, the capacitor voltage increasing means the synaptic weight decreasing. Thus, the synapse is potentiated during the capacitor C discharge as happens in figure 8 (b).

TABLE I
VARIABLES THAT DESCRIBES THE NEURON OPERATION

Var.	Description	Module
V_I	Input potential	Information processing
V_{BE}	Neuron activation threshold	Information processing
V_{TH}	Output of integrator module	Information processing
V_C	Synaptic weight or efficiency	/Learning
V_{CA}	Adjusting potential for STP and LTP.	Learning
V_u	Learning module output / spike generator input	Learning/Spike generator
V_{OE}	Excitatory neuron output	Spike generator

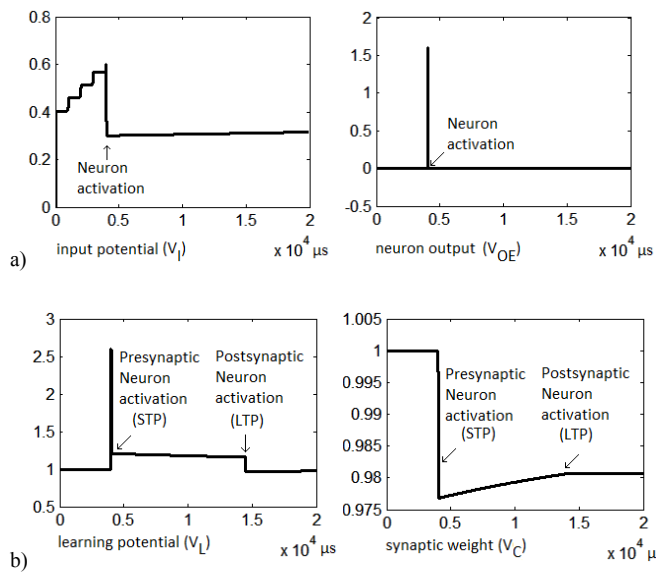


Fig. 8. The variation of the mathematical model variables: a) during neuron activation; b) during STP and LTP

The sudden decrease of the variable V_L which takes place less before 15 ms is caused by the postsynaptic neuron activation which triggers LTP. This mechanism will stop the learning capacitor C recharging and therefore the synaptic weight decreasing.

IV. CONCLUSION

The set of equations presented in this paper represents the mathematical model of a neuron which is hardware implementable. The computer simulations performed in MATLAB shows a good concordance between the electronic neuron behavior and the variables variation. This is encouraging for using the mathematical model in developing in an easier way the synaptic configuration of analogue neural networks. Also, this model can be used for establishing the limits and restrictions on the parameters variation which ensures the proper operation of the electronic neuron. By adjusting some parameters in the set of equations it can be obtained the ideal operation of the artificial neuron which can help in improving the existing model.

REFERENCES

- [1] M. Hulea, "A model of silicon neurons suitable for speech recognition", *CEAI Journal*, Vol 10, No. 4, pp 32-41, 2008.
- [2] M. Hulea, A new method to obtain non-volatile memory for networks of spiking neurons, *Memoirs of Scientific Sections*, Tome XXXIII, , pp. 129-146, 2010.
- [3] M. Hulea, "Method for spoken words recognition using a spiking neuron model of biological inspiration" RO Patent A 2009 00531 (22) / 09.07.2009.

- [4] W. Maass, C.M. Bishop "Pulsed neural networks", Cambridge MA: The MIT Press, , pp. 23-42, November 1998.
- [5] W. Gerstner and W. Kistler, "Spiking neurons models: single neurons, populations, plasticity", Cambridge Univ. Press, , pp. 102 – 111, 2002.
- [6] R. Jolivet, Timothy J. Lewis, and W. Gerstner, "Generalized integrate-and-fire models of neuronal activity approximate spike trains of a detailed model to a high degree of accuracy", *Neurophysiol.*, vol. 92, pp. 959–976, 2004.
- [7] J. J. Lovelace and K. J. Cios, "A very simple spiking neuron model that allows for modeling of large, complex systems", *Neural Comput.*, vol. 20, no. 1, pp. 65-90, January 2008.
- [8] R. Kobayashi, Y. Tsubo and S. Shinomoto, "Made-to-order spiking neuron model equipped with a multi-timescale adaptive threshold" *Front Comput Neurosci.*, vol. 3, no. 9, paper 9, 2009.
- [9] Rossant C, Goodman DFM, Platkiewicz J and Brette R (2010). „Automatic fitting of spiking neuron models to electrophysiological recordings." *Front. Neuroinform.* vol. 4, no. 2, 2010.
- [10] E. M. Izhikevich, "Polychronization: computation with spikes", *Neural Comput.*, vol. 18, issue 2, pp. 245-282, February 2006.
- [11] E. M. Izhikevich "Which model to use for cortical spiking neurons?" *IEEE Transactions on Neural Networks*, vol. 15, pp. 1063-1070, 2004.
- [12] W. Swiercz and K. J. Cios, "A New Synaptic Plasticity Rule for Networks of Spiking Neurons", *IEEE Transaction on Neural Networks*, vol. 17, no. 1, pp. 94-99, Jan. 2006.
- [13] Jh. Shin, et. al. "Recognition of partially occluded and rotated images with a network of spiking neurons", *IEEE Trans Neural Netw.*, vol. 21, no. 10, pp. 1697-1709, Nov. 2010.
- [14] H. Markram and F. Shürmann, "Fully implicit parallel simulation of single neurons", *J. Comput. Neurosci.*, vol. 25, pp. 439–448, 2008.
- [15] W. Maass, T. Natschläger, and H. Markram, "Computational models for generic cortical microcircuits", *Computational Neuroscience: A Comprehensive Approach*, CRC-Press, Eds. Feng, J.M.L. Hines, pp. 575 – 605, 2004
- [16] M. L. Hines, H. Eichner and F. Schürmann, „Neuron spitting in compute-bound parallel network simulations enables runtime scaling with twice as many processors", *J. Comput. Neurosci.*, vol. 25, no. 3, pp. 203-210, 2008.
- [17] R. C. O'Reyly and Y. Munakata, *Computational Explorations in Cognitive Neuroscience.*, The MIT Press, pp. 32-37, 2000.
- [18] G. Bi and M. Poo, "Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type", *J. Neurosci.*, vol. 18, no. 24, pp. 10464-10472, 1998.
- [19] Y. Dan and M. Poo, "Spike Timing Dependent Plasticity of Neural circuits", *Neuron*, vol. 24, pp. 23-30, 2004.
- [20] T. Masquelier, "Competitive-STDP based spike pattern learning", *Neur. Comput.*, vol. 21, no. 5, pp. 1259-1276, 2009.
- [21] V. Jacob, D. J. Brasier, I. Erchova, D. Feldman and D. E. Shulz, "Spike Timing-Dependent Synaptic Depression in the *In Vivo* Barrel Cortex of the Rat", *J. Neurosci.*, vol. 27, no. 6, pp. 1271-1284, 2007.

A Multibackground March Test For All Static Simple Neighborhood Pattern-Sensitive Faults in RAMs

Cristina Huzum, and Petru Cascaval

Abstract— Based on all realistic static two-cell coupling faults presented by Hamdioui, van de Goor, and Rodgers in [1], we have defined a complex model of neighborhood pattern sensitive faults (NPSFs) in $N \times 1$ random-access memories [2]. Thus, the large model of NPSFs we have considered (called all static NPSFs) includes state coupling faults, transition coupling faults, write disturb coupling faults, read destructive coupling faults, deceptive read destructive coupling faults, and incorrect read coupling faults. A new multibackground march test able to cover this large model of NPSFs is presented in this paper.

Index Terms— Memory Testing, Functional Fault Model, Static Faults, Neighborhood Pattern-Sensitive Faults, Multibackground March Test

I. INTRODUCTION

Rapid increase of density in the integrated circuits has an immediate effect upon memory testing. On one hand, the capacity of random-access memories chips enhances, thus increasing the test time and cost; on the other hand, the density of memory circuits grows, therefore more failure modes and faults need to be taken into account in order to obtain a good quality product. Accordingly, there are two conflicting constraints that need to be dealt with when considering a test algorithm: reducing the number of memory operations in order to permit large capacity memories to be tested in an appropriate period of time and covering a larger variety of memory faults [3], [4].

As a result of the increasing coupling effect triggered by the growing density of memory circuits, the pattern-sensitive fault (PSF) is becoming an important fault model [5], [6], [7]. The PSF model is a type of coupling fault, with several aggressor cells (4, 9 etc.) and only one victim cell. In this work, the neighborhood PSF (NPSF) has been considered. This is a particular PSF, in which the aggressor cells are located in the physical neighborhood of the victim cell. The NPSF model was first defined by Hayes in 1980 [5]. He also devised a memory test for this model [5]. Soon after that, Suk and Reddy have proposed a new memory test [8] based on a bipartite method. This test divides the memory cells into two partitions and applies a sequence of transitions to cover all possible victim-aggressor combinations. Unfortunately, for the memory chips currently used, the test proposed by Suk and Reddy needs a long time to perform. In 2002, other more efficient march tests were given by Cheng, Tsai, and Wu, namely: March-100N and CM-79N [9]. In another paper, written in 2008, Julie, Wan Zuha and Sidek use a modified version of

March-100N named March-12N for diagnosis of SRAM [10]. In [2] the model of all static NPSF has been introduced, and a simulation study has been made in order to determine the fault coverage of these faults by the most important published tests dedicated to the NPSF model. Simulation results show that CM-96N [9] with a length of 96N has the best fault coverage for the model of all NPSFs, namely 70,95%. This result shows the fault coverage for all NPSFs is low enough and, consequently, a new memory test dedicated to this model is necessary.

The remainder of this paper is organized as follows. Section II introduces some notations and definitions, and Section III defines the set of fault primitives for the model of all static NPSFs. Section IV presents the new memory test, March-NPSF, able to detect all simple faults of this model of NPSFs. Some conclusions concerning this work are drawn in Section V.

II. NOTATIONS, DEFINITIONS AND FAULT CLASSIFICATIONS

An operation sequence that results in a difference between the observed and the expected memory behaviour is called a *sensitizing operation sequence* (S). The observed memory behaviour that deviates from the expected one is called *faulty behaviour* (F). In order to specify a certain fault, one has to specify the S , together with the corresponding faulty behaviour F , and the read result (R) of S in case it is a read operation. The combination of S , F and R for a given memory failure is called a *Fault Primitive* (FP), and is usually denoted as $\langle S / F / R \rangle$. The concept of FPs allows for establishing a complete framework of all memory faults. Some classifications of FPs can be made based on different and independent factors of S [10].

a) Depending on the number of simultaneous operations required in the S , FPs are classified into *single-port* and *multi-port* faults.

- *Single-ports faults*: These are FPs that require at the most one port in order to sensitize a fault. Note that single-port faults can be sensitized in single-port as well as in multi-port memories.
- *Multi-port faults*: These are FPs that can only sensitize a fault by performing two or more operations simultaneously via different ports.

b) Depending on the number of sequential operations required in the S , FPs are classified into *static faults* and *dynamic faults*. Let $\#O$ be the number of different operations carried out sequentially in a S .

- *Static faults*: These are FPs which sensitize a fault by performing at most one operation in the memory ($\#O=0$ or $\#O=1$);
- *Dynamic faults*: These are FPs that perform more than one operation sequentially in order to sensitize a fault ($\#O > 1$).

Manuscript received: May 30, 2010.

Cristina Huzum is with the Faculty of Automatic Control and Computer Engineering, "Gheorghe Asachi" Technical University of Iași, Romania (e-mail: cristinahuzum@yahoo.com).

Petru Cascaval is with the Faculty of Automatic Control and Computer Engineering, "Gheorghe Asachi" Technical University of Iași, Romania (e-mail: cascaval@cs.tuiasi.ro).

c) Depending on the way FPs manifest themselves, they can be divided into *simple faults* and *linked faults*.

- *Simple faults*: These are faults which cannot be influenced by another fault. That means that the behaviour of a simple fault cannot change the behaviour of another one; therefore masking cannot occur.
- *Linked faults*: These are faults that do influence the behaviour of each other. That means that the behaviour of a certain fault can change the behaviour of another one such that masking can occur. Note that linked faults consist of two or more simple faults.

In this work, single-port, static, simple faults are considered. From here on, the term ‘fault’ refers to a single-port, static, simple fault.

The following notations are usually used to describe operations on RAMs :

- $r0$ ($r1$) denotes a read operation when the expected value is 0 (1);
- $0w0$ ($1w1$) denotes a write 0 (1) operation to a cell which contains a 0 (1) – the state of the cell does not change;
- $0w1$ ($1w0$) denotes an up (down) transition write operation;

III. THE MODEL OF ALL STATIC NEIGHBORHOOD PATTERN-SENSITIVE FAULTS

RAM faults can also be divided into *single-cell* and *multi-cell* faults. Single-cell faults consist of FPs involving a single cell, while multi-cell faults consists of FPs involving more than one cell. In this work, we consider a particular class of multi-cell faults (also called *coupling faults*), namely the pattern sensitive faults (PSF). The PSF is a coupling fault, which affects the content of a memory cell (called the *victim cell* or the *base cell*), or the ability to change its content, when other memory cells (called *aggressor cells*) have certain patterns. It is unnecessary and unrealistic to consider all possible patterns of all the memory cells, therefore simplified models of neighborhood pattern sensitive faults (NPSF) have been introduced. In these models, the aggressor cells are limited to the physical neighborhood of the victim cell. Depending on the number of aggressor cells, NPSF can be divided into several types, but only two of those are more common: Type-1 NPSF that has four aggressor cells and Type-2 NPSF that has eight aggressor cells as illustrated in Fig. 1a and Fig. 1b, respectively [11].

Like in the most previous works, in this paper only Type-1 NPSF is studied, because it is more practical for the type of memory we have considered.

Based on the notations previously defined, a FP for NPSF is presented as

$$< S / F / R > = < S_N ; S_W ; S_E ; S_S ; S_B / F / R > ,$$

where S_N, S_W, S_E, S_S and S_B are the sensitizing operation (or state) sequences performed on the N, W, E and S aggressor cells, and on the B -cell (*base cell*), respectively. N, W, E, S are the cells to which the sensitizing operation (or state) should be applied in order to sensitize the fault, while the B -cell is the cell where the fault appears. Notice that $S_N,$

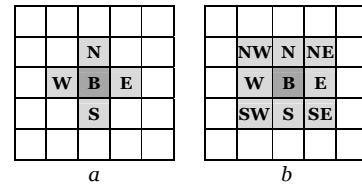


Fig. 1 – Common types of neighborhood pattern sensitive faults:
a – Type-1 NPSF; b – Type-2 NPSF.

$S_W, S_B, S_E, S_S \in \{0, 1, r0, r1, 0w0, 1w1, 0w1, 1w0\}$. There are 576 FPs (see Table I) grouped into the seven FFM as defined in [10], namely:

- state coupling faults ($CFst$) – 32 primitives;
- disturb coupling faults ($CFds$) – 384 primitives;
- transition coupling faults ($CFtr$) – 32 primitives;
- write destructive coupling faults ($CFwd$) – 32 primitives;
- read destructive coupling faults ($CFrd$) – 32 primitives;
- deceptive read destructive coupling faults ($CFdrd$) – 32 primitives;
- incorrect read coupling faults ($CFir$) – 32 primitives.

IV. A NEW MEMORY TEST – MARCH-NPSF

In order to describe the memory test, first some notations regarding the march tests are given. Usually, a complete march test is delimited by ‘{...}’ bracket pair, while a march element is delimited by the ‘(...)’ bracket pair. March elements are separated by semicolons, and the operations within a march element are separated by commas. Note that all operations of a march element are performed at a certain address, before proceeding to the next address. The whole memory is checked homogeneously in either one of two orders: ascending address order ($\Uparrow$) or descending address order ($\Downarrow$). When the address order is not relevant, the symbol $\Updownarrow$ is used.

As stated in [9], if a certain March element is applied to a solid background (all ones or all zeros), when we are reading from or writing to a cell C , then all the cells with higher address than C have the same state, while those with lower address than C also have the same state. So, most of the NPSF faults cannot be activated by applying March elements to these solid backgrounds. Therefore, the necessity of a March test that runs under several different backgrounds arises. These are called multibackground March tests [12],[13].

The test we propose, called March-NPSF, uses sixteen different backgrounds, denoted by $BG_0, BG_1, \dots, BG_{15}$, as presented in Fig. 2. For a certain cell, let a be the value within the background, and b the complement of a . For example, when initializing the memory with BG_1 , a is 0 for the lines 1, 3, 4, 6, 7, 9 etc. of the memory, and 1 for lines 2, 5, 8 etc. These backgrounds were selected for the test in order to create all victim-aggressor combinations necessary for activating all the faults in the NPSF model.

March-NPSF forms an alternating series of March elements and background changes (as Cockburn proposed in [14]) and has the following structure:

TABLE I
LIST OF NPSF PRIMITIVES

Fault type	Fault primitives	
<i>CFst</i>	$\langle x; y; z; t; 0 / 1 / - \rangle$ $\langle x; y; z; t; 1 / 0 / - \rangle$ $\langle r0; x; y; z; 0 / 1 / - \rangle$ $\langle r0; x; y; z; 1 / 0 / - \rangle$ $\langle r1; x; y; z; 0 / 1 / - \rangle$ $\langle r1; x; y; z; 1 / 0 / - \rangle$ $\langle 0w0; x; y; z; 0 / 1 / - \rangle$ $\langle 0w0; x; y; z; 1 / 0 / - \rangle$ $\langle 1w1; x; y; z; 0 / 1 / - \rangle$ $\langle 1w1; x; y; z; 1 / 0 / - \rangle$ $\langle 0w1; x; y; z; 0 / 1 / - \rangle$ $\langle 0w1; x; y; z; 1 / 0 / - \rangle$ $\langle 1w0; x; y; z; 0 / 1 / - \rangle$ $\langle 1w0; x; y; z; 1 / 0 / - \rangle$ $\langle x; r0; y; z; 0 / 1 / - \rangle$ $\langle x; r0; y; z; 1 / 0 / - \rangle$ $\langle x; r1; y; z; 0 / 1 / - \rangle$ $\langle x; r1; y; z; 1 / 0 / - \rangle$ $\langle x; 0w0; y; z; 0 / 1 / - \rangle$ $\langle x; 0w0; y; z; 1 / 0 / - \rangle$ $\langle x; 1w1; y; z; 0 / 1 / - \rangle$ $\langle x; 1w1; y; z; 1 / 0 / - \rangle$ $\langle x; 0w1; y; z; 0 / 1 / - \rangle$ $\langle x; 0w1; y; z; 1 / 0 / - \rangle$ $\langle x; 1w0; y; z; 0 / 1 / - \rangle$ $\langle x; 1w0; y; z; 1 / 0 / - \rangle$ $\langle x; y; r0; z; 0 / 1 / - \rangle$ $\langle x; y; r0; z; 1 / 0 / - \rangle$ $\langle x; y; r1; z; 0 / 1 / - \rangle$ $\langle x; y; r1; z; 1 / 0 / - \rangle$ $\langle x; y; 0w0; z; 0 / 1 / - \rangle$ $\langle x; y; 0w0; z; 1 / 0 / - \rangle$ $\langle x; y; 1w1; z; 0 / 1 / - \rangle$ $\langle x; y; 1w1; z; 1 / 0 / - \rangle$ $\langle x; y; 0w1; z; 0 / 1 / - \rangle$ $\langle x; y; 0w1; z; 1 / 0 / - \rangle$ $\langle x; y; 1w0; z; 0 / 1 / - \rangle$ $\langle x; y; 1w0; z; 1 / 0 / - \rangle$ $\langle x; y; z; r0; 0 / 1 / - \rangle$ $\langle x; y; z; r0; 1 / 0 / - \rangle$ $\langle x; y; z; r1; 0 / 1 / - \rangle$ $\langle x; y; z; r1; 1 / 0 / - \rangle$ $\langle x; y; z; 0w0; 0 / 1 / - \rangle$ $\langle x; y; z; 0w0; 1 / 0 / - \rangle$ $\langle x; y; z; 1w1; 0 / 1 / - \rangle$ $\langle x; y; z; 1w1; 1 / 0 / - \rangle$ $\langle x; y; z; 0w1; 0 / 1 / - \rangle$ $\langle x; y; z; 0w1; 1 / 0 / - \rangle$ $\langle x; y; z; 1w0; 0 / 1 / - \rangle$ $\langle x; y; z; 1w0; 1 / 0 / - \rangle$	$x, y, z, t \in \{0, 1\}$
<i>CFds</i>	$\langle x; y; r0; z; 0 / 1 / - \rangle$ $\langle x; y; r0; z; 1 / 0 / - \rangle$ $\langle x; y; r1; z; 0 / 1 / - \rangle$ $\langle x; y; r1; z; 1 / 0 / - \rangle$ $\langle x; y; 0w0; z; 0 / 1 / - \rangle$ $\langle x; y; 0w0; z; 1 / 0 / - \rangle$ $\langle x; y; 1w1; z; 0 / 1 / - \rangle$ $\langle x; y; 1w1; z; 1 / 0 / - \rangle$ $\langle x; y; 0w1; z; 0 / 1 / - \rangle$ $\langle x; y; 0w1; z; 1 / 0 / - \rangle$ $\langle x; y; 1w0; z; 0 / 1 / - \rangle$ $\langle x; y; 1w0; z; 1 / 0 / - \rangle$ $\langle x; y; z; r0; 0 / 1 / - \rangle$ $\langle x; y; z; r0; 1 / 0 / - \rangle$ $\langle x; y; z; r1; 0 / 1 / - \rangle$ $\langle x; y; z; r1; 1 / 0 / - \rangle$ $\langle x; y; z; 0w0; 0 / 1 / - \rangle$ $\langle x; y; z; 0w0; 1 / 0 / - \rangle$ $\langle x; y; z; 1w1; 0 / 1 / - \rangle$ $\langle x; y; z; 1w1; 1 / 0 / - \rangle$ $\langle x; y; z; 0w1; 0 / 1 / - \rangle$ $\langle x; y; z; 0w1; 1 / 0 / - \rangle$ $\langle x; y; z; 1w0; 0 / 1 / - \rangle$ $\langle x; y; z; 1w0; 1 / 0 / - \rangle$	$x, y, z \in \{0, 1\}$
<i>CFtr</i>	$\langle x; y; z; t; 0w1 / 0 / - \rangle$ $\langle x; y; z; t; 1w0 / 1 / - \rangle$	$x, y, z, t \in \{0, 1\}$
<i>CFwd</i>	$\langle x; y; z; t; 0w0 / 1 / - \rangle$ $\langle x; y; z; t; 1w1 / 0 / - \rangle$	$x, y, z, t \in \{0, 1\}$
<i>CFrd</i>	$\langle x; y; z; t; r0 / 1 / 1 \rangle$ $\langle x; y; z; t; r1 / 0 / 0 \rangle$	$x, y, z, t \in \{0, 1\}$
<i>CFdrd</i>	$\langle x; y; z; t; r0 / 1 / 0 \rangle$ $\langle x; y; z; t; r1 / 0 / 1 \rangle$	$x, y, z, t \in \{0, 1\}$
<i>CFir</i>	$\langle x; y; z; t; r0 / 0 / 1 \rangle$ $\langle x; y; z; t; r1 / 1 / 0 \rangle$	$x, y, z, t \in \{0, 1\}$

$\{\hat{\uparrow}(w0); [\hat{\uparrow}(ra, wb, wb, rb); \hat{\uparrow}(rb, wa, wa, ra); CBG_i],$
 $i=1, 2, \dots, 16\}$, where CBG_i is a sequence for checking
and changing from the current background to the next. More
exactly, the sequence CBG_i has the following meaning:

$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$
BG ₀	BG ₁	BG ₂	BG ₃
$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$
BG ₄	BG ₅	BG ₆	BG ₇
$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$
BG ₈	BG ₉	BG ₁₀	BG ₁₁
$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
BG ₁₂	BG ₁₃	BG ₁₄	BG ₁₅

Fig. 2. The 16 backgrounds for the test March-NPSF

- $CBG_i, i=1, 2, \dots, 15$, changes the background from BG_{i-1} to BG_i . Note that when changing from one background to the next, only the cells that must change states are written. Each write operation is also preceded (for checking) by a read operation. We can observe in Fig. 2 that the backgrounds were ordered such as the difference between every two successive backgrounds consists of three out of the nine cells so that all background changes affect only a trinity of the cells.
- CBG_{16} reads the whole memory for the finally checking ($CBG_{16} = \hat{\uparrow}(ra)$).

Consequently, March-NPSF comprises the following operations:

- N operations for the first initialization;
- $[4N + 4N + 2 \times 3N/9] \times 15 = 130N$ operations, for the first fifteen series of March elements and background changes;
- $4N + 4N + N = 9N$, for the sixteenth series of March elements and final checking.

Therefore, the length of March-NPSF is $N + 130N + 9N = 140N$.

A simulation study has been made in order to determine whether March-NPSF covers all simple faults of this NPSF model. Taking into account that the patterns of the backgrounds used by this test are composed of 3×3 cells, for the simulation study, the memory cells have been divided into nine mutually disjoint subsets, denoted by $S_1, S_2, \dots, S_9$, depending on their location. Let r and c be the row address and the column address, respectively, of a memory cell. The cell belongs to a certain subset according to the following formulas:

$$\text{the cell } (r, c) \in \begin{cases} S_1 & \text{if } c \% 3 = 0 \text{ and } r \% 3 = 0 \\ S_2 & \text{if } c \% 3 = 0 \text{ and } r \% 3 = 1 \\ S_3 & \text{if } c \% 3 = 0 \text{ and } r \% 3 = 2 \\ S_4 & \text{if } c \% 3 = 1 \text{ and } r \% 3 = 0 \\ S_5 & \text{if } c \% 3 = 1 \text{ and } r \% 3 = 1 \\ S_6 & \text{if } c \% 3 = 1 \text{ and } r \% 3 = 2 \\ S_7 & \text{if } c \% 3 = 2 \text{ and } r \% 3 = 0 \\ S_8 & \text{if } c \% 3 = 2 \text{ and } r \% 3 = 1 \\ S_9 & \text{if } c \% 3 = 2 \text{ and } r \% 3 = 2, \end{cases}$$

where “%” denotes the modulo operation.

For a memory array with 8 rows and 8 columns, these nine subsets of cells are illustrated in Fig. 3.

Note that every memory cell that belongs to the same cell subset will support the same operations during the test March-NPSF. Moreover, if two base cells belong to the same subset, their aggressor cells will support, respectively, the same initializations. Hence, for the simulation study, only nine locations (one for each subset) have to be considered for the base cell.

The conclusion of the simulation study is that March-NPSF covers entirely this large model of NPSFs. In addition to the simulation study made in [2], these results indicate that the proposed test is the only algorithm that covers entirely this fault model.

Theorem: March-NPSF is able to cover the model of all static NPSFs.

Proof:

When the memory is initialized with a certain background, this causes the appearance of different bit sequences in neighborhoods that have the base cell in different subsets. For example, when the memory is initialized with BG₂, if the base cell belongs to S₁, its neighborhood contains the bit sequence presented in Fig. 4a, and if the base cell belongs to S₅, its neighborhood consists of the bit sequence presented in Fig. 4b.

Table II presents for each background, the bit sequences that appear in the neighborhood of the base cell when it belongs to each subset. Note that, whichever subset the base cell belongs to, the initialization of the memory cells with all backgrounds will determine the appearance of all 16 possible bit sequences or their complements (denoted by I₁, I₂, ..., I₁₆, and I₁' , I₂' , ... I₁₆' , respectively, as shown in Table III) in a different order.

TABLE II
BIT BIT SEQUENCE FOR EACH BACKGROUND AND FOR EACH SUBSET OF MEMORY CELLS

				S ₁				S ₂				S ₃				S ₄				S ₅				S ₆				S ₇				S ₈				S ₉																																																																																																																																																																																																						
BG ₀	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	I ₁	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	I ₁	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	I ₁	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	I ₁	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	I ₁																																																																																																								
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
BG ₁	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	I ₂	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	I ₁₆	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	I ₃ '	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	I ₂	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	I ₁₆	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	I ₃ '																																																																														
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
BG ₂	<table><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	0	0	I ₃	<table><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	0	0	I ₂ '	<table><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	0	0	I ₁₆ '	<table><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	0	0	I ₃	<table><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	0	0	I ₂ '	<table><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	0	0	0	0	I ₁₆ '																																																																														
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	1	1	1	1	1																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	1	1	1	1	1																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	1	1	1	1	1																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	1	1	1	1	1																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	1	1	1	1	1																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	0	0	0	0	0																																																																																																																																																																																																																																			
1	1	1	1	1	1	1	1																																																																																																																																																																																																																																			
0	0	0	1	0	0	0	0																																																																																																																																																																																																																																			
BG ₃	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₄	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₇ '	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₅ '	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₉ '	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₈	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₁₄	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₁₀	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₁₁ '	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	I ₁₅ '
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
BG ₄	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₅	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₄ '	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₇	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₁₄ '	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₉	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₈	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₁₅	<table><tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₁₁																										
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	1	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
BG ₅	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₆	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₆	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₆	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₁₃ '	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₁₃ '	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₁₃ '	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₁₂	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	0	0	1	0	1	0	I ₁₂																										
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
BG ₆	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₇	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₅	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₄ '	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₈ '	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₁₄ '	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₉	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₁₁	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₁₀ '	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>	1	0	0	0	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	0	1	0	I ₁₀ '
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
0	1	1	0	0	1	0	1																																																																																																																																																																																																																																			
1	0	0	0	1	0	1	0																																																																																																																																																																																																																																			
BG ₇	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₈	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₁₄	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₉ '	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₁₁ '	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₁₅ '	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₁₀	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₇ '	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₄	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	1	0	1	1	0	0	1	0	0	1	1	0	1	1	0	0	1	0	1	1	I ₄
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
0	0	1	0	0	1	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	1	1																																																																																																																																																																																																																																			
BG ₈	<table><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	1	1	0	0	1	0	0	1	I ₉	<table><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	1	1	0	0	1	0	0	1	I ₈ '	<table><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	1	1	0	0	1	0	0	1	I ₁₄ '	<table><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	0	0	1	1	0	0	1																																																																																																																																																				
0	0	1	1	0	0	1	0																																																																																																																																																																																																																																			
0	0	1	1	0	0	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	0	1																																																																																																																																																																																																																																			
0	0	1	1	0	0	1	0																																																																																																																																																																																																																																			
0	0	1	1	0	0	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	0	1																																																																																																																																																																																																																																			
0	0	1	1	0	0	1	0																																																																																																																																																																																																																																			
0	0	1	1	0	0	1	0																																																																																																																																																																																																																																			
1	1	0	0	1	0	0	1																																																																																																																																																																																																																																			
0	0	1	1	0	0	1																																																																																																																																																																																																																																				

	0	1	2	3	4	5	6	7
0	S ₁	S ₄	S ₇	S ₁	S ₄	S ₇	S ₁	S ₄
1	S ₂	S ₅	S ₈	S ₂	S ₅	S ₈	S ₂	S ₅
2	S ₃	S ₆	S ₉	S ₃	S ₆	S ₉	S ₃	S ₆
3	S ₁	S ₄	S ₇	S ₁	S ₄	S ₇	S ₁	S ₄
4	S ₂	S ₅	S ₈	S ₂	S ₅	S ₈	S ₂	S ₅
5	S ₃	S ₆	S ₉	S ₃	S ₆	S ₉	S ₃	S ₆
6	S ₁	S ₄	S ₇	S ₁	S ₄	S ₇	S ₁	S ₄
7	S ₂	S ₅	S ₈	S ₂	S ₅	S ₈	S ₂	S ₅

Fig. 3. The cell subsets for an 8×8 memory chip array.

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	1	1	1	1	1	1
5	0	0	0	0	0	0

a)

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	1	1	1	1	1	1
5	0	0	0	0	0	0

b)

Fig 4. Bits sequences that appear in BG₂: a – Base cell belongs to S₁; b – Base cell belongs to S₅

On the other hand, another simulation study has been made in order to determine for each fault which part of the test sensitizes it and which part detects it. Each of the sixteen series of our test is composed of three test sequences (two March elements and a test sequence for changing the background) identified with a superscript (x), where $x \in \{1, 2, 3\}$, as follows:

$$\{[\hat{1}(ra, wb, wb, rb)^{(1)}; \hat{1}(rb, wa, wa, ra)^{(2)}; CBG_i^{(3)}], i=1, 2, \dots, 16\}$$

Table IV shows the 36 faults detected by applying March-NPSF on the bit sequence I₁. The pair (i, j) denotes the jth operation of element (i); e.g., (1,2) denotes the second operation (i.e. wb) of element (1).

Generalizing, the faults detected by applying the memory test on a certain bit sequence are shown in Table V. Note that, by applying the two March elements (1) and (2) on a bit sequence or on their complement, the same operations are carried out in the memory, and consequently, the same memory faults are detected.

The bit sequence from a neighborhood is written as *xyvzt* and it contains, respectively, the values in the north, west, victim, east, and south cell (as shown in Table III).

TABLE III
BIT SEQUENCES

Notation	Bit sequence (xyvzt)	Notation	Bit sequence (xyvzt)
I ₁	00000	I ₉	11000
I ₂	00001	I ₁₀	01101
I ₃	01111	I ₁₁	00011
I ₄	11010	I ₁₂	00010
I ₅	11011	I ₁₃	10111
I ₆	10101	I ₁₄	11001
I ₇	10100	I ₁₅	01100
I ₈	10110	I ₁₆	01110

TABLE IV
FAULTS DETECTED BY APPLYING MARCH-NPSF ON I₁ BIT SEQUENCE

Fault type	Fault primitive	Sensitization	Detection
CFst	<0; 0; 1; 1; 0 / 1 / - >	(2, 2)	(2, 4)
	<1; 1; 0; 0; 1 / 0 / - >	(1, 2)	(1, 4)
	<r0; 0; 0; 0; 0 / 1 / - >	(1, 1)	(1, 1)
	<r1; 1; 1; 1; 1 / 0 / - >	(2, 1)	(2, 1)
	<0w0; 1; 1; 1; 1 / 0 / - >	(2, 3)	(2, 1)
	<1w1; 0; 0; 0; 0 / 1 / - >	(1, 3)	(1, 1)
	<0w1; 0; 0; 0; 0 / 1 / - >	(1, 2)	(1, 1)
	<1w0; 1; 1; 1; 1 / 0 / - >	(2, 2)	(2, 1)
	<1; r0; 0; 0; 0 / 1 / - >	(1, 1)	(1, 1)
	<0; r1; 1; 1; 1 / 0 / - >	(2, 1)	(2, 1)
	<0; 0w0; 1; 1; 1 / 0 / - >	(2, 3)	(2, 1)
	<1; 1w1; 0; 0; 0 / 1 / - >	(1, 3)	(1, 1)
	<1; 0w1; 0; 0; 0 / 1 / - >	(1, 2)	(1, 1)
	<0; 1w0; 1; 1; 1 / 0 / - >	(2, 2)	(2, 1)
	<1; 1; r0; 0; 1 / 0 / - >	(1, 1)	(2, 1)
	<0; 0; r1; 1; 0 / 1 / - >	(2, 1)	(3, 1)
CFds	<0; 0; 0w0; 1; 0 / 1 / - >	(2, 3)	(3, 1)
	<1; 1; 1w1; 0; 1 / 0 / - >	(1, 3)	(2, 1)
	<1; 1; 0w1; 0; 1 / 0 / - >	(1, 2)	(2, 1)
	<0; 0; 1w0; 1; 0 / 1 / - >	(2, 2)	(3, 1)
	<1; 1; 1; r0; 1 / 0 / - >	(1, 1)	(2, 1)
	<0; 0; 0; r1; 0 / 1 / - >	(2, 1)	(3, 1)
	<0; 0; 0; 0w0; 0 / 1 / - >	(2, 3)	(3, 1)
	<1; 1; 1; 1w1; 1 / 0 / - >	(1, 3)	(2, 1)
	<1; 1; 1; 0w1; 1 / 0 / - >	(1, 2)	(2, 1)
	<0; 0; 0; 1w0; 0 / 1 / - >	(2, 2)	(3, 1)
	<1; 1; 1; r0; 1 / 0 / - >	(1, 1)	(2, 1)
	<0; 0; 0; r1; 0 / 1 / - >	(2, 1)	(3, 1)
	<0; 0; 0; 0w0; 0 / 1 / - >	(2, 3)	(3, 1)
	<1; 1; 1; 1w1; 1 / 0 / - >	(1, 3)	(2, 1)
	<1; 1; 1; 0w1; 1 / 0 / - >	(1, 2)	(2, 1)
	<0; 0; 0; 1w0; 0 / 1 / - >	(2, 2)	(3, 1)
CFtr	<1; 1; 0; 0; 0w1 / 0 / - >	(1, 2); (1, 3)	(1, 4)
	<0; 0; 1; 1; 1w0 / 1 / - >	(2, 2); (2, 3)	(2, 4)
CFwd	<0; 0; 1; 1; 0w0 / 1 / - >	(2, 3)	(2, 4)
	<1; 1; 0; 0; 1w1 / 0 / - >	(1, 3)	(1, 4)
CFrd	<0; 0; 1; 1; r0 / 1 / 1 >	(2, 4)	(2, 4)
	<1; 1; 0; 0; r1 / 0 / 0 >	(1, 4)	(1, 4)
CFdrd	<0; 0; 1; 1; r0 / 1 / 0 >	(2, 4)	(3, 1)
	<1; 1; 0; 0; r1 / 0 / 1 >	(1, 4)	(2, 1)
CFir	<0; 0; 1; 1; r0 / 0 / 1 >	(2, 4)	(2, 4)
	<1; 1; 0; 0; r1 / 1 / 0 >	(1, 4)	(1, 4)

As can be seen, Table V also includes 36 faults from the model of all NPSFs. Considering the significance of the values from the table, it can be concluded that by applying the march test on every of the 16 bit sequences or their complements, 36 unique faults are detected. So, applying the test on all bit sequences or their complements determines the detection of $16 \times 36 = 576$ unique faults, hence all the faults from the considered model.

Note that Table II shows that whichever subset the base cell belongs to, applying all of the 16 backgrounds to the memory will determine the appearance in the neighborhood of the base cell of all 16 bit sequences or their complements in an different order. Therefore, it can be concluded that wherever the base cell is located, March-NPSF is able to detect all simple NPSFs.

V. CONCLUSION

A new multibackground memory test dedicated to a large model of NPSFs is presented in this paper. With a length of 140N, March-NPSF is able to detect all simple faults of this model. On compared with the test CM-96N given by Cheng, Tsai and Wu, March-NPSF is longer but more efficient. Remember that, CM-96N covers only 70,95% simple faults of this large model of NPSF.

TABLE V
FAULTS DETECTED BY APPLYING MARCH-NPSF ON A BIT
SEQUENCE OF FORM $x; y; z; t$

Fault type	Fault primitive	Sensitization	Detection
CFst	$\langle x; y; \bar{z}; \bar{t}; v / \bar{v} / - \rangle$	(2, 2)	(2, 4)
	$\langle \bar{x}; \bar{y}; z; t; \bar{v} / v / - \rangle$	(1, 2)	(1, 4)
	$\langle rx; y; z; t; v / \bar{v} / - \rangle$	(1, 1)	(1, 1)
	$\langle r\bar{x}; \bar{y}; \bar{z}; \bar{t}; \bar{v} / v / - \rangle$	(2, 1)	(2, 1)
	$\langle xwx; \bar{y}; \bar{z}; \bar{t}; \bar{v} / v / - \rangle$	(2, 3)	(2, 1)
	$\langle \bar{x}w\bar{x}; y; z; t; v / \bar{v} / - \rangle$	(1, 3)	(1, 1)
	$\langle xw\bar{x}; y; z; t; v / \bar{v} / - \rangle$	(1, 2)	(1, 1)
	$\langle \bar{x}wx; \bar{y}; \bar{z}; \bar{t}; \bar{v} / v / - \rangle$	(2, 2)	(2, 1)
	$\langle \bar{x}; ry; z; t; v / \bar{v} / - \rangle$	(1, 1)	(1, 1)
	$\langle x; r\bar{y}; \bar{z}; \bar{t}; \bar{v} / v / - \rangle$	(2, 1)	(2, 1)
	$\langle x; ywy; \bar{z}; \bar{t}; \bar{v} / v / - \rangle$	(2, 3)	(2, 1)
	$\langle \bar{x}; \bar{y}w\bar{y}; z; t; v / \bar{v} / - \rangle$	(1, 3)	(1, 1)
	$\langle \bar{x}; yw\bar{y}; z; t; v / \bar{v} / - \rangle$	(1, 2)	(1, 1)
	$\langle x; \bar{y}w\bar{y}; \bar{z}; \bar{t}; \bar{v} / v / - \rangle$	(2, 2)	(2, 1)
	$\langle \bar{x}; \bar{y}; rz; t; \bar{v} / v / - \rangle$	(1, 1)	(2, 1)
	$\langle x; y; r\bar{z}; \bar{t}; \bar{v} / \bar{v} / - \rangle$	(2, 1)	(3, 1)
CFds	$\langle x; y; zwz; \bar{t}; v / \bar{v} / - \rangle$	(2, 3)	(3, 1)
	$\langle \bar{x}; \bar{y}; \bar{z}w\bar{z}; t; \bar{v} / v / - \rangle$	(1, 3)	(2, 1)
	$\langle \bar{x}; \bar{y}; zw\bar{z}; t; \bar{v} / v / - \rangle$	(1, 2)	(2, 1)
	$\langle x; y; \bar{z}wz; \bar{t}; v / \bar{v} / - \rangle$	(2, 2)	(3, 1)
	$\langle \bar{x}; \bar{y}; \bar{z}; rt; \bar{v} / v / - \rangle$	(1, 1)	(2, 1)
	$\langle x; y; z; r\bar{t}; v / \bar{v} / - \rangle$	(2, 1)	(3, 1)
	$\langle x; y; z; twt; v / \bar{v} / - \rangle$	(2, 3)	(3, 1)
	$\langle \bar{x}; \bar{y}; \bar{z}; \bar{t}w\bar{t}; \bar{v} / v / - \rangle$	(1, 3)	(2, 1)
	$\langle \bar{x}; \bar{y}; \bar{z}; tw\bar{t}; \bar{v} / v / - \rangle$	(1, 2)	(2, 1)
	$\langle x; y; z; \bar{t}w\bar{t}; v / \bar{v} / - \rangle$	(2, 2)	(3, 1)
	$\langle \bar{x}; \bar{y}; z; t; vw\bar{v} / v / - \rangle$	(1, 2); (1, 3)	(1, 4)
	$\langle x; y; \bar{z}; \bar{t}; \bar{v}wv / \bar{v} / - \rangle$	(2, 2); (2, 3)	(2, 4)
	$\langle x; y; \bar{z}; \bar{t}; v\bar{w}v / \bar{v} / - \rangle$	(2, 3)	(2, 4)
	$\langle \bar{x}; \bar{y}; z; t; \bar{v}w\bar{v} / v / - \rangle$	(1, 3)	(1, 4)
	$\langle x; y; \bar{z}; \bar{t}; rv / \bar{v} / \bar{v} \rangle$	(2, 4)	(2, 4)
	$\langle \bar{x}; \bar{y}; z; t; r\bar{v} / v / v \rangle$	(1, 4)	(1, 4)
CFdrd	$\langle x; y; \bar{z}; \bar{t}; rv / \bar{v} / v \rangle$	(2, 4)	(3, 1)
	$\langle \bar{x}; \bar{y}; z; t; r\bar{v} / v / \bar{v} \rangle$	(1, 4)	(2, 1)
CFir	$\langle x; y; \bar{z}; \bar{t}; rv / v / \bar{v} \rangle$	(2, 4)	(2, 4)
	$\langle \bar{x}; \bar{y}; z; t; r\bar{v} / \bar{v} / v \rangle$	(1, 4)	(1, 4)

REFERENCES

- [1] S. Hamdioui, A. J. van de Goor, M. Rodgers, "March SS: A test for all static simple RAM faults", in *Proc. of IEEE Int'l Workshop on Memory Technology, Design and Testing*, pp.95–100, 2002.
- [2] C. Huzum, P. Cascaval, "A Fault Primitive Based Model For All Static Neighbourhood Pattern-Sensitive Faults In Random Access Memories " in *Bulet. Instit. Politehn. Iasi, Tom LV (LIX), Fasc. 3*, 2009.
- [3] R. D. Adams, "High performance memory testing: design principles, fault modelling and self-test", Ed. Kluwer Academic Publishers, USA, 2003, pp 150–183.
- [4] S. Hamdioui, "Testing static random access memories: defects, fault models and test patterns", Ed. Kluwer Academic Publishers, The Netherlands, 2004, pp 54 – 106.
- [5] J. P. Hayes, "Testing memories for single-cell pattern-sensitive fault" in *IEEE Trans. Comput.*, Vol. 29, pp 249–254, 1980.
- [6] B. F. Cockburn, "Deterministic tests for detecting scrambled pattern-sensitive faults in RAMs" in *Proc. IEEE Int. Workshop Memory Technology, Design and Testing (MTDT)*, pp.117-122, 1995.
- [7] D. C. Kang, S. B. Cho, "An efficient build-in self-test algorithm for neighborhood pattern sensitive faults in high-density memories", in *Proc. 4th Korea-Russia Int. Symp. Science and Technology*, vol. 2, pp. 218–223, 2000.

- [8] D. S. Suk, M. Reddy, "Test procedures for a class of pattern-sensitive faults in semiconductor random-access memories", in *IEEE Trans. Comput.*, vol. 29, pp.419–429, 1980.
- [9] K. L. Cheng, M. F. Tsai, C. W. Wu, "Neighborhood pattern-sensitive fault testing and diagnostics for random-access memories", in *IEEE Trans. On CAD*, vol. 21, no. 11, pp.1328–1336, 2002.
- [10] R. R. Julie, W. H. Wan Zuha, R. M. Sidek, "12N test procedure for NPSF testing and diagnosis for SRAMs", in *Proc. IEEE Int. Conf. on Semiconductor Electronics*, pp.430–435, 2008.
- [11] A. J. van de Goor, "Testing semiconductor memories: theory and practice", John Wiley & Sons Inc, pp. 465–490, 1991.
- [12] V. Yarmolik, Y. Klimets, S. Demidenko, "March PS(23N) test for DRAM pattern-sensitive faults" in *Proc. Seventh IEEE Asian Test Symp. (ATS)*, pp.354–357, 1998.
- [13] K. L. Cheng, M. F. Tsai, C. W. Wu, "Efficient neighborhood pattern-sensitive fault test algorithms for semiconductor memories" in *Proc. IEEE VLSI Test Symp. (VTS)*, pp.225–237, 2001.
- [14] B. F. Cockburn, "Deterministic tests for detecting single V-coupling faults in RAMs" in *J. Electronic Testing. Theory and Applications*, vol. 5(1), pp. 91–113, 1994.

Modeling and Control of Continuous-Review Perishable Inventory Systems with Multiple Supply Alternatives

P. Ignaciuk and A. Bartoszewicz

Abstract—In this paper, the problem of efficient control of continuous-review perishable inventory systems is addressed from the control-theoretic perspective. In the considered systems, the goods at a distribution center used to fulfill unknown, variable demand are subject to exponential decay. The stock is replenished from multiple supply sources characterized by different latency in procuring orders. A new supply policy which incorporates the Smith predictor to counteract the adverse effects of delay is developed. The proposed policy guarantees that the assigned storage space at the distribution center is never exceeded. Moreover, we show that with appropriately chosen controller parameters all of the demand imposed at the distribution center is realized from the readily available resources. As a result, the maximum service level is ensured.

I. INTRODUCTION

IT FOLLOWS from the extensive review papers documenting the research work in the past [1]–[5] that certain areas of inventory control are not sufficiently addressed at the formal design level. This concerns in particular an important class of problems related to the management of perishable commodities (food, drugs, gasoline, etc.). The main difficulty in developing control schemes for perishable inventories is the necessity of conducting a thorough analysis of product lifetimes. The design problem poses a serious challenge in the situation when the demand is subject to significant uncertainty and inventories are replenished with nonnegligible delay, possibly from multiple supply sources [6]. For such systems, typically only heuristic solutions are available [7].

In this paper, a formal approach to the inventory policy design is applied. In contrast to our earlier results devoted exclusively to periodic-review inventory systems with non-deteriorating stock [8]–[10], in this work continuous-review systems with finite lifetime of the stored goods is analyzed. We consider a system in which the stock used to fulfill uncertain, variable demand imposed on a goods distribution center, decays exponentially. The demand can follow any statistical distribution, possibly a nonstationary one. The

stock is replenished with delay from multiple supply sources. In order to solve the stability problems related to nonnegligible delay (see e.g. [11] for a discussion of the influence of delay on the dynamics of the traditional inventory systems), we propose to apply the Smith predictor [12]. The standard form of the Smith predictor is adapted for the system with multiple delays in the feedback loop analyzed in this work.

The designed control strategy is demonstrated to establish nonnegative and bounded ordering signal, which is a crucial requirement for the practical implementation of any replenishment rule. It is also shown that in the inventory system governed by the proposed policy the stock level never exceeds the assigned warehouse capacity, which means that the potential necessity for expensive emergency storage outside the company premises is eliminated. At the same time, it is demonstrated that the stock is never depleted, which implies full satisfaction of unknown demand from the readily available resources and maximum service level. It is noted, however, that the designed controller may generate overshoots at the output in response to abrupt demand changes. Therefore, in order to overcome the problems related to increased storage space due to the stock level overshoots, we propose a modified controller. The improved policy retains all the favorable properties of the original scheme, yet avoids excessive stock growth in the situation of sudden changes in the market trend.

II. PROBLEM FORMULATION

We consider an inventory system where the goods at a distribution center used to fulfill the customers' (or retailers) demand are acquired with delay from multiple supply sources. In such setting, illustrated in Fig. 1, a common point (distribution center) is used to provide goods for another production stage or a distribution network. The task is to design a control strategy which, on one hand, will minimize the holding and shortage costs, and, on the other hand, will ensure smooth flow of goods despite unpredictable changes in market conditions.

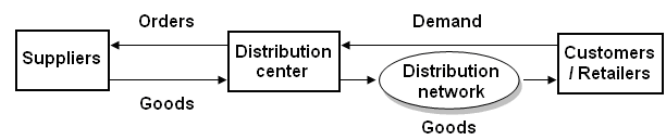


Fig. 1. Flow of goods and information.

This work was financed by the Polish State budget in the years 2010–2012 as a research project N N514 108638 “Application of regulation theory methods to the control of logistic processes”. The first author is a scholarship holder of the project entitled “Innovative Education without Limits – Integrated Progress of the Technical University of Łódź” supported by the European Social Fund.

P. Ignaciuk (e-mail: przemyslaw.ignaciuk@p.lodz.pl) is with Institute of Information Technology, 215 Wólczajska St. and A. Bartoszewicz (e-mail: andrzej.bartoszewicz@p.lodz.pl) is with Institute of Automatic Control, 18/22 Stefanowskiego St., Technical University of Łódź, 90-924 Łódź, Poland.

A. System Model

The imposed demand (the number of items requested from the distribution center) is modeled as an *a priori* unknown, bounded function of time $d(t)$, where t denotes time. We assume that the demand can follow any statistical distribution as long as $0 \leq d(t) \leq d_{\max}$, where $d_{\max}$ is a positive constant. If there is a sufficient number of items at the distribution center to satisfy the imposed demand, then the actually met demand $h(t)$ (the number of items sold to the customers or sent to the retailers in the distribution network) will be equal to the requested one. Otherwise, the imposed demand is satisfied only from the arriving shipments, and the additional demand is lost (we assume that the sales are not backordered, and the excessive demand is equivalent to a missed business opportunity). Thus, we may write the following relation

$$0 \leq h(t) \leq d(t) \leq d_{\max}. \quad (1)$$

Note that since $d(t)$ is *a priori* unknown, also the function representing the realized sales cannot be determined *a priori*, and is treated as a disturbance in the considered system.

The on-hand stock used to fulfill the demand deteriorates when kept in the distribution center warehouse at rate σ , $0 \leq \sigma < 1$. It is replenished with delay from (possibly) several supply sources. Denoting the overall order quantity generated at time t by $u(t)$, and the share of supplier p ($p = 1, 2, \dots, m$) by λ_p , we have the order assigned for supplier p equal to $u_p(t) = \lambda_p u(t)$. The share λ_p is a real number from the interval $[0, 1]$ satisfying $\sum_{p=1}^m \lambda_p = 1$. Typically, the values of λ_p will be obtained from a cost optimization algorithm [13]. In the limit case, when $\lambda_p = 1$, a single supplier is selected to deliver the whole order quantity, while $\lambda_p = 0$ implies no replenishment from option p . Each non-zero order placed at supplier p is realized with lead-time delay $L_p > 0$. Without the loss of generality we may order the supply alternatives according to their lead time in the following way

$$L_1 \leq L_2 \leq \dots \leq L_{m-1} \leq L_m. \quad (2)$$

Denoting the shipment received from supplier p by $u_R^p(t)$, we have

$$u_R^p(t) = u_p(t - L_p) = \lambda_p u(t - L_p). \quad (3)$$

Consequently, the stock balance equation in the considered system with multiple supply alternatives can be written in the following way

$$\dot{y} = -\sigma y(t) + \sum_{p=1}^m \lambda_p u(t - L_p) - h(t). \quad (4)$$

According to the stock balance equation, the on-hand stock decreases due to the realized sales represented by function $h(\cdot)$, and the decay characterized by factor σ . It is refilled

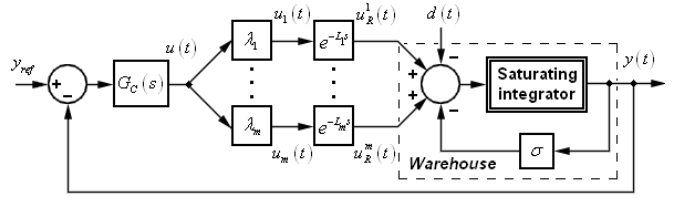


Fig. 2. System model.

from the goods acquired from the suppliers $\sum_{p=1}^m u_R^p(t)$. For the sake of further analysis it is convenient to represent (4) in an integral form. We assume that initially the warehouse is empty, i.e. $y(0) = 0$, and the first orders are placed at $t = 0$, i.e. $u(t) = 0$ for $t < 0$. Solving (4) for $y(\cdot)$, we obtain

$$\begin{aligned} y(t) &= \sum_{p=1}^m \int_0^t e^{-\sigma(t-\tau)} \lambda_p u(\tau - L_p) d\tau - \int_0^t e^{-\sigma(t-\tau)} h(\tau) d\tau \\ &= \sum_{p=1}^m \lambda_p \int_0^{t-L_p} e^{-\sigma(t-L_p-\tau)} u(\tau) d\tau - \int_0^t e^{-\sigma(t-\tau)} h(\tau) d\tau. \end{aligned} \quad (5)$$

Note that in order to adequately model the stock accumulation of perishable goods, a saturating integrator should be applied (since due to the absence of backorders the stock level must not become negative). From the modeling perspective, this makes the considered system nonlinear. However, if one can ensure that the control signal is nonnegative for arbitrary t , then by introducing the function representing the actually realized sales, $h(t) \leq d(t)$, the stock dynamics can be reduced to the linear equation (5). In the further part of the paper, we will design a control law which will be shown to satisfy the conditions $u(t) \geq 0$ and $y(t) > 0$. If the stock level remains positive after serving the current demand, then the realized sales $h(t) = d(t)$. As a result, the inventory system will stay in the linear region of operation for the whole range of the disturbance $0 \leq d(t) \leq d_{\max}$.

B. Transfer Function Representation

The linear part of the model of the considered inventory system with perishable goods can be described using transfer functions. The system block diagram is shown in Fig. 2. The saturating integrator in the internal loop represents accumulating the stock of perishables characterized by decay factor σ . The controller, with transfer function $G_C(s)$, is supposed to steer the on-hand stock level $y(t)$ towards the reference value y_{ref} such that high level of demand satisfaction is achieved.

III. PROPOSED CONTROL STRATEGY

The primary obstacle in providing efficient control in the considered class of systems is the latency in procuring orders. In order to satisfactorily counteract the adverse effects of delay in the analyzed system with perishable goods, it is not sufficient to introduce inventory position variables (constituting the sum of on-hand stock and open orders), or the notion of work-in-progress, as it is usually done in the traditional inventory systems [4]. This is due to the fact that the pure sum of open orders (or the work-in-progress) does not account for the stock degradation within lead time. To over-

come the delay problem, in this work we propose to apply the Smith predictor [12], which proved a successful method of dead-time compensation in many engineering areas.

A. Principal Control Strategy

The proposed control strategy employing the (generalized) Smith predictor for dead-time compensation in the system with multiple delays is illustrated in Fig. 3.

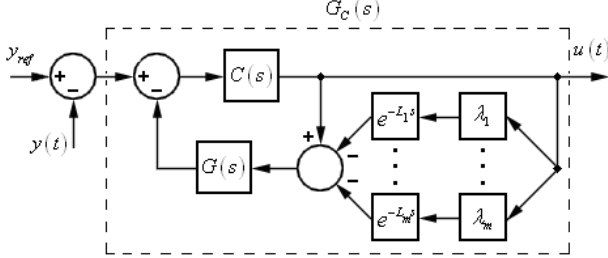


Fig. 3. Smith predictor based controller.

The control structure consists of the primary plant controller $C(s)$ and the Smith predictor with the linearized model of the plant $G(s) = 1/(s + \sigma)$. With the primary controller selected as the proportional one $C(s) = K$, where K is a positive constant, we obtain the transfer function of the overall control structure $G_C(s)$,

$$G_C(s) = \frac{K}{1 + KG(s)(1 - \sum_{p=1}^m \lambda_p e^{-L_p s})}. \quad (6)$$

Taking into account the initial conditions, we obtain in time-domain

$$u(t) = K[y_{ref} - y(t)] - K \left[\int_0^t e^{-\sigma(t-\tau)} u(\tau) d\tau - \sum_{p=1}^m \lambda_p \int_0^{t-L_p} e^{-\sigma(t-L_p-\tau)} u(\tau) d\tau \right]. \quad (7)$$

This control law can be interpreted as to generate orders in proportion to the difference between the current on-hand stock and its reference value $K[y_{ref} - y(t)]$ decreased by the amount of open orders quantified by the rate of deterioration within the last lead time.

B. Properties of the Principal Strategy

The properties of the proposed control strategy will be given in three theorems, and strictly proved. The first proposition shows that the ordering signal generated by controller (7) is always nonnegative and bounded, which is a crucial prerequisite for the implementation of any cost-efficient inventory policy. The second theorem specifies the upper bound of the on-hand stock, which constitutes the smallest warehouse capacity required to store all the incoming shipments. Finally, the third proposition shows how to select the stock reference value in order to guarantee that all of the imposed demand will be fulfilled from the readily available resources at the distribution center. As a result, the maximum service level is ensured.

Theorem 1: For any time instant $t \geq 0$ the ordering signal generated by controller (7) applied to system (4) satisfies the following inequalities

$$\sigma K y_{ref} / (\sigma + K) \leq u(t) \leq K y_{ref}. \quad (8)$$

Moreover, there exists an instant t_0 such that for any $t \geq t_0$

$$u(t) \leq K(\sigma y_{ref} + d_{\max}) / (\sigma + K). \quad (9)$$

Proof: Substituting (5) into (7) we get

$$u(t) = K \left[y_{ref} - \int_0^t e^{-\sigma(t-\tau)} u(\tau) d\tau + \int_0^t e^{-\sigma(t-\tau)} h(\tau) d\tau \right]. \quad (10)$$

Consequently,

$$\begin{aligned} \dot{u} &= -K \frac{d}{dt} \left\{ e^{-\sigma t} \int_0^t e^{\sigma \tau} [u(\tau) - h(\tau)] d\tau \right\} \\ &= \sigma K \int_0^t e^{-\sigma(t-\tau)} [u(\tau) - h(\tau)] d\tau - K[u(t) - h(t)]. \end{aligned} \quad (11)$$

It follows from (10) that

$$\sigma K \int_0^t e^{-\sigma(t-\tau)} [u(\tau) - h(\tau)] d\tau = \sigma [K y_{ref} - u(t)]. \quad (12)$$

Hence, equation (11) may be rewritten as

$$\dot{u} = \sigma K y_{ref} - (\sigma + K)u(t) + K h(t). \quad (13)$$

Investigating, $\dot{u} = 0$ we get

$$u(t) = K[\sigma y_{ref} + h(t)] / (\sigma + K). \quad (14)$$

According to constraint (1) the minimum satisfied demand equals zero. At the initial time $u(0) = K y_{ref} > 0$. Therefore, since $0 \leq \sigma < 1$ and $h(\cdot) \geq 0$, we get from (13) and (14) that the control signal decreases ($\dot{u} < 0$) as long as it is bigger than $K[\sigma y_{ref} + h(\cdot)] / (\sigma + K)$, and it never falls below $K \sigma y_{ref} / (\sigma + K)$. Moreover, there exists a time instant t_0 when $u(\cdot)$ reaches the level of $K[\sigma y_{ref} + d_{\max}] / (\sigma + K)$ for the first time. Since $h(\cdot) \leq d_{\max}$, we get from (14) that for all $t \geq t_0$

$$u(t) \leq K(\sigma y_{ref} + d_{\max}) / (\sigma + K).$$

This conclusion ends the proof. $\square$

Theorem 2: If policy (7) is applied to control the flow of goods in system (4), then the on-hand stock at the distribution center never exceeds the level of y_{ref} for $\sigma = 0$ and

$$y_{\max} = \frac{K}{\sigma + K} \left[y_{ref} + \frac{d_{\max}}{\sigma} \left(1 - \sum_{p=1}^m \lambda_p e^{-\sigma L_p} \right) \right] \quad \text{for } \sigma > 0. \quad (15)$$

Proof: Applying (10) to (4), we get

$$\dot{y} = -\sigma y(t) + \sum_{p=1}^m \lambda_p K y_{ref} - h(t) - \sum_{p=1}^m \lambda_p K \left[\int_0^{t-L_p} e^{-\sigma(t-L_p-\tau)} u(\tau) d\tau - \int_0^{t-L_p} e^{-\sigma(t-L_p-\tau)} h(\tau) d\tau \right]. \quad (16)$$

Since $\sum_{p=1}^m \lambda_p = 1$, using (5) we may modify (16) as follows

$$\begin{aligned} \dot{y} &= K y_{ref} - (\sigma + K) y(t) - h(t) \\ &\quad + K \sum_{p=1}^m \lambda_p (e^{\sigma L_p} - 1) \int_0^{t-L_p} e^{-\sigma(t-\tau)} h(\tau) d\tau \\ &\quad - K \sum_{p=1}^m \lambda_p \int_{t-L_p}^t e^{-\sigma(t-\tau)} h(\tau) d\tau. \end{aligned} \quad (17)$$

Investigating $\dot{y} = 0$ leads to

$$\begin{aligned} y(t) &= \frac{K y_{ref}}{\sigma + K} + \frac{K}{\sigma + K} \sum_{p=1}^m \lambda_p (e^{\sigma L_p} - 1) \int_0^{t-L_p} e^{-\sigma(t-\tau)} h(\tau) d\tau \\ &\quad - \frac{K}{\sigma + K} \left[\sum_{p=1}^m \lambda_p \int_{t-L_p}^t e^{-\sigma(t-\tau)} h(\tau) d\tau + h(t) / K \right]. \end{aligned} \quad (18)$$

Since $h(\cdot) \geq 0$, we get immediately from (18) that for $\sigma = 0$ (the case of nondeteriorating stock) $y(t) \leq y_{ref}$. Moreover, since $K > 0$, the biggest value of $y(\cdot)$ when $\sigma > 0$ is expected when the demand realized from the shipments from supplier p is maximum in the interval $(L_p, t - L_p]$, and zero in the interval $(t - L_p, t]$. Evaluating the first integral in (18) for $\sigma > 0$ and $h(t) = d_{\max}$ we obtain

$$\int_0^{t-L_p} e^{-\sigma(t-\tau)} h(\tau) d\tau \leq d_{\max} \int_0^{t-L_p} e^{-\sigma(t-\tau)} d\tau \leq d_{\max} \sigma^{-1} e^{-\sigma L_p}. \quad (19)$$

Consequently, applying (19) to (18), we arrive at

$$\begin{aligned} y(t) &\leq \frac{K y_{ref}}{\sigma + K} + \frac{K}{\sigma + K} \sum_{p=1}^m \lambda_p (e^{\sigma L_p} - 1) \frac{d_{\max}}{\sigma} e^{-\sigma L_p} \\ &= \frac{K}{\sigma + K} \left[y_{ref} + \frac{d_{\max}}{\sigma} \sum_{p=1}^m \lambda_p (1 - e^{-\sigma L_p}) \right]. \end{aligned} \quad (20)$$

From (15) we get $y(t) \leq y_{\max}$, which ends the proof. $\square$

Theorem 3: If policy (7) is applied to system (4), and the reference stock level is selected as

$$y_{ref} > d_{\max} \left(\sum_{p=1}^m \lambda_p L_p + 1 / K \right) \quad \text{for } \sigma = 0, \quad (21)$$

$$y_{ref} > d_{\max} \left[\left(1 - \sum_{p=1}^m \lambda_p e^{-\sigma L_p} \right) / \sigma + 1 / K \right] \quad \text{for } \sigma > 0, \quad (22)$$

then the on-hand stock level at the distribution center is strictly positive for any $t > L_m$.

Proof: Note that $e^{\sigma L_p} - 1 \geq 0$. Hence, considering (1) and (18), we can expect the smallest on-hand stock level in the circumstances when the demand realized from the shipments from supplier p is zero for $\tau \leq t - L_p$ and maximum for τ from the interval $(t - L_p, t]$. In the case of the system with nondeteriorating stock ($\sigma = 0$) we get immediately from (18)

$$y(t) \geq y_{ref} - d_{\max} \left(\sum_{p=1}^m \lambda_p L_p + 1 / K \right). \quad (23)$$

Thus, using assumption (21) we have $y(t) > 0$ for $\sigma = 0$. Evaluating the second integral in (18) for $t > L_p$ when $h(t) = d_{\max}$ and $\sigma > 0$, we obtain

$$\int_{t-L_p}^t e^{-\sigma(t-\tau)} h(\tau) d\tau \leq d_{\max} \sigma^{-1} (1 - e^{-\sigma L_p}). \quad (24)$$

Applying (24) to (18), we get the estimate of the on-hand stock level $y(\cdot)$ at the instant when it is minimum

$$y(t) \geq \frac{K}{\sigma + K} \left[y_{ref} - \sum_{p=1}^m \lambda_p \frac{d_{\max}}{\sigma} (1 - e^{-\sigma L_p}) - \frac{d_{\max}}{K} \right]. \quad (25)$$

If the reference stock level is adjusted according to (22), then using (25) one may conclude that $y(t) > 0$ for the case of $\sigma > 0$. This completes the proof. $\square$

C. Modified Control Strategy

The Smith predictor based controller (7) establishes a smooth, nonoscillatory ordering signal. However, an overshoot in the output variable (the stock level) may be generated for an abrupt change in the market trend. This leads to increased warehouse capacity required to accommodate the resulting spike in $y(\cdot)$. Below, we formulate an enhanced control law, which allows for maintaining all the favorable properties of the original strategy, yet with an overshoot-free output and thus reduced warehouse capacity.

We propose to apply the following control law

$$u(t) = K \left[y_{ref} - \Lambda y(t) - \Lambda \sum_{p=1}^m \lambda_p \int_{t-L_p}^t e^{-\sigma(t-L_p-\tau)} u(\tau) d\tau \right], \quad (26)$$

where $\Lambda = 1 / \sum_{p=1}^m \lambda_p e^{\sigma L_p}$. Therefore, in the modified scheme, the order quantity is generated in proportion to the reference stock level minus the scaled output (the term $\Lambda y(t)$), decreased by the amount of goods on-route quantified by the rate of deterioration within lead-time (the integral in (26)). The structure of the improved controller is illustrated in Fig. 4. We can notice from Fig. 4 that the signals in the delay compensating loop are multiplied by factor Λ .

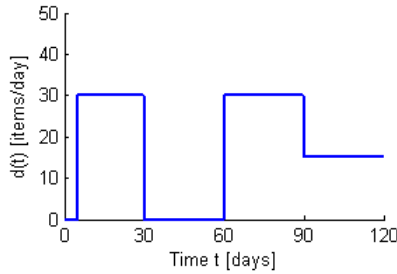


Fig. 5. Demand at the distribution center.

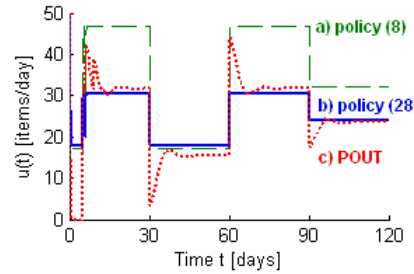


Fig. 6. Generated orders:
a) policy (7), b) policy (26), c) policy (34).

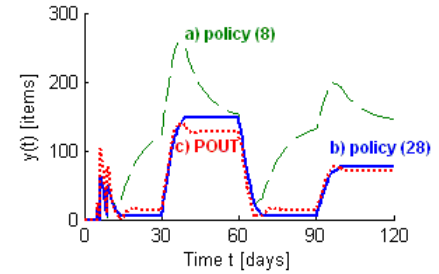


Fig. 7. On-hand stock:
a) policy (7), b) policy (26), c) policy (34).

as $y_{ref} = 150 > 146$ items. This results in the required storage space: for policy (7) determined from Theorem 2 as 281 items, and for policy (26) obtained from Theorem 5 as 149 items. The order-up-to level for the POUT policy is adjusted so that it generates the same holding costs as the improved policy (34). We set $y_{OUT} = 236$ items.

The orders generated by the controllers are shown in Fig. 6, and the resulting on-hand stock level in Fig. 7: policy (7) – curve a), policy (26) – curve b), and policy (34) – curve c). We can see from the graphs depicted in Fig. 6 that the proposed controllers (7) and (26) quickly respond to the sudden changes in the demand trend without overshoots or overshoots in the ordering signal. Policy (26) establishes smaller order quantities than controller (7), which leads to smaller purchase costs while maintaining a given service level. The ordering signal generated by the POUT policy is contaminated by overshoots and oscillations, thus being more difficult to follow and requiring bigger safety stock from the suppliers. We can see from Fig. 7 that the stock level resulting from the application of the classical inventory policy also exhibits small-degree overshoots and oscillations, and occasionally drops to zero (day 14) incurring lost-sales costs. The stock level resulting from the use of policies (7) and (26) does not increase beyond the maximum level calculated from Theorems 2 and 5, which means that the assigned warehouse capacity is sufficient to store the goods at the distribution center at all times. Moreover, the on-hand stock never falls to zero after the initial phase which implies full demand satisfaction and the maximum service level. The $y(t)$ curve obtained from policy (7) exhibits large overshoots after sudden changes in the demand trend, which lead to increased holding and warehouse maintenance costs. The improved policy (26) is free from overshoots, and it offers the least costly solution of the three investigated controllers.

V. CONCLUSIONS

In this work, a new supply policy employing the Smith predictor for continuous-review perishable inventory systems was proposed. The ordering signal established by the controller smoothly adapts to the demand changes, and thus it is easy to follow by the suppliers. The stock level does not increase beyond the precisely determined warehouse capacity, which eliminates the need for costly emergency storage and facilitates capacity planning at the goods distribution

center. It is shown how to select the controller parameters to achieve full satisfaction of unknown market demand. Since the basic Smith predictor based strategy may lead to overshoots in the stock level in response to abrupt demand changes, we introduce a modified controller. The improved control law retains all the favorable properties of the original scheme but ensures exponential convergence of the stock level to steady state without overshoots and oscillations. The improved policy outperforms the classical order-up-to replenishment rule in counteracting the bullwhip effect, and in terms of holding and lost-sales costs.

REFERENCES

- [1] S. Nahmias, "Perishable inventory theory: a review," *Oper. Res.*, vol. 30, no. 4, pp. 680–708, 1982.
- [2] F. Raftaat, "Survey of literature on continuously deteriorating inventory models," *J. Oper. Res. Soc.*, vol. 42, no. 1, pp. 27–37, 1991.
- [3] S. K. Goyal and B. C. Giri, "Recent trends in modeling of deteriorating inventory," *Eur. J. Oper. Res.*, vol. 134, no. 1, pp. 1–16, 2001.
- [4] H. Sarimveis, P. Patrinos, C. D. Tarantilis, and C. T. Kiranoudis, "Dynamic modeling and control of supply chain systems: a review," *Comput. Oper. Res.*, vol. 35, no. 11, pp. 3530–3561, 2008.
- [5] I. Karaesmen I, A. Scheller-Wolf, and B. Deniz, "Managing perishable and aging inventories: review and future research directions," In: K. Kempf, P. Keskinocak, and R. Uzsoy (eds.) *Handbook of production planning*. Dordrecht: Kluwer, 2008.
- [6] S. Minner, "Multiple-supplier inventory models in supply chain management: a review," *Int. J. Prod. Econ.*, vol. 81–82, no. 1, pp. 265–279, 2003.
- [7] R. A. C. M. Broekmeulen and K. H. van Donselaar, "A heuristic to manage perishable inventory with batch ordering, positive lead times, and time-varying demand," *Comput. Oper. Res.*, vol. 36, no. 11, pp. 3013–3018, 2009.
- [8] P. Ignaciuk and A. Bartoszewicz, "LQ optimal sliding mode supply policy for periodic review inventory systems," *IEEE Trans. Autom. Control*, vol. 55, no. 1, pp. 269–274, Jan. 2010.
- [9] P. Ignaciuk, A. Bartoszewicz, "Linear-quadratic optimal control strategy for periodic-review inventory systems," *Automatica*, vol. 46, no. 12, pp. 1982–1993, 2010.
- [10] P. Ignaciuk, A. Bartoszewicz, "Linear-quadratic optimal control strategy for periodic-review inventory systems," *Int. J. Syst. Sci.*, vol. 42, 2011 (in press).
- [11] K. Hoberg, J. R. Bradley, and U. W. Thonemann, "Analyzing the effect of the inventory policy on order and inventory variability with linear control theory," *Eur. J. Oper. Res.*, vol. 176, no. 3, pp. 1620–1642, 2007.
- [12] O. J. C. Smith, "A controller to overcome dead time," *ISA J.*, vol. 6, no. 2, pp. 28–33, 1959.
- [13] W. Dullaert, B. Vernimmen, B. Raa, and F. Witlox, "A hybrid approach to designing inbound-resupply strategies," *IEEE Intel. Syst.*, vol. 20, no. 4, pp. 31–35, Jul./Aug. 2005.
- [14] J. Dejonckheere, S. M. Disney, M. R. Lambrecht, and D. R. Towill, "Measuring and avoiding the bullwhip effect: a control theoretic approach," *Eur. J. Oper. Res.*, vol. 147, no. 3, pp. 567–590, 2003.

GPU optimized computation of the artificial compressibility method

Lucian Mihai Itu, Constantin Suciu, Florin Moldoveanu and Adrian Postelnicu

Abstract – The paper introduces an optimized GPU based implementation of the artificial compressibility method, which is used to solve the incompressible Navier-Stokes equations. A finite difference approach has been chosen for the numerical solution of the continuity and momentum equations. The simulations have been performed on a two dimensional backward facing step problem discretized on a Marker and Cell grid, which is used in order to improve the stability of the solution. The computationally intensive parts of the algorithm are performed on the GPU, i.e. the computation of the velocities and of the pressure values at the grid nodes. Due to the lack of communication between the blocks of the GPU grid, the computations have been included in two separate kernels, and the outer loop which iterates through the time steps, has been kept on the CPU. Several optimization strategies have been applied for the two kernels and have lead to an incremental increase in performance. The opposite nature of the two kernels has lead to different optimum versions. The final versions of the kernels have been then used to perform a comparison between the CPU and the GPU version of the algorithms on three different grained grids. The results indicate a speed-up which varies from just under one order of magnitude for the coarsest grid up to two orders of magnitude for the finest grid.

I. INTRODUCTION

An alternative to CPU based solutions has been raised from Graphics Processing Unit (GPU) based implementations. The GPUs were introduced initially as graphical accelerators with focus, mainly, to image processing applications. A GPU is a stream processor, specifically designed to perform a very large number of floating point operations (Flops) in parallel by simultaneously using multiple computational units [1]. The relative recent introduction of non-graphics application programming interfaces (API) for GPUs brought a new perspective on these, transforming them into general purpose units (GPGPU) [2].

The GPU's highly parallel device structure is the main reason for its good results [3]. When a GPU is programmed through CUDA (Compute Unified Device Architecture), the GPU is viewed as a compute device which is able to run a large number of threads in parallel. A kernel is a function, written in C language, which is executed on the GPU by several threads [4]. The GPU contains several streaming

multiprocessors, each of them containing eight cores.

Practically, the vast majority of GPU implementations are done in a CPU-GPU tandem manner [5]. The CPU (called host) is responsible for launching the main application and initializing the used data (e.g. acquired from a database or a process). This data has to be transferred to the GPU for processing. The GPU (usually also called device) contains a certain amount of global memory to/from which the CPU or host thread can write/read, being accessible to all multiprocessors.

A major parameter, indicating whether it is worth to move the computational intensive part of a program to a GPU, is the execution time compared to the situation when the code is executed entirely on the host. In order to have a complete picture of the comparison, when the execution time for the GPU is determined, the execution times required by the data manipulation between host and device have to be taken into consideration.

Computational fluid dynamics (CFD) is one of the most demanding branches of science from a computational point of view. When the appearance of the CUDA architecture was announced, leading CFD experts have identified its potential to accelerate computational fluid dynamics to unprecedented levels [6]. GPU-powered personal workstations are already able to provide levels of performance comparable to those of supercomputers. Experimental results can be analyzed within seconds with the help of CUDA based implementations. This streamlines the feedback process on which researchers rely in order to achieve breakthroughs.

The goal of this article has been to obtain an optimized GPU based implementation of the incompressible Navier-Stokes using the artificial compressibility method compared to reference work [7], which is the only paper describing the GPU implementation of the artificial compressibility method. A finite difference method (FDM) approach [8] is used, which, in order to improve the stability of the results, will be implemented on a staggered grid (also called Marker and Cell - MAC - grid [9]). Section two will provide a detailed description of the incompressible Navier-Stokes equations, of the artificial compressibility method and of the MAC grid on which they will be implemented. Section three focuses on the GPU implementation and on the code optimization activities. Section four presents the results of our research activity and shows the performance improvements brought by the GPU version. Section 5 is dedicated to conclusions related to this work.

L. M. Itu is with the Transilvania University of Brasov, 500036 Romania (phone: 0040741029888; e-mail: lucian.itu@unitbv.ro).

C. Suciu is with Siemens Corporate Technology and Transilvania University of Brasov, 500036 Romania (e-mail: constantin.suciu@siemens.com).

F. Moldoveanu is with the Transilvania University of Brasov, 500036 Romania (e-mail: moldof@unitbv.ro).

A. Postelnicu is with the Transilvania University of Brasov, 500036 Romania (e-mail: adip@unitbv.ro).

II. THE INCOMPRESSIBLE NAVIER-STOKES EQUATIONS

The incompressible (viscous) Navier-Stokes equations describe a wide range of problems in fluid dynamics. The further description will be based on two dimensional flow problems. As a result, the equations are composed of a mass conservation equation and two momentum conservation equations, one for each Cartesian velocity component [10], [11]. The dependent variables, which will be determined numerically, are the pressure p and the velocity components u and v in the x and y directions respectively (for simplicity only the isothermal case will be considered).

When dealing with incompressible flows, there are two different approaches: primitive variable methods and vortex methods. The former includes the artificial compressibility (ACM) [12], and the pressure correction methods (PCM) including the marker and cell (MAC) methods [9]. The major difficulty in incompressible flows is to determine an accurate solution of the pressure. This is why the vortex methods remove the pressure terms from the momentum equations (by solving the vorticity transport equations) [13].

The mass conservation equation:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (2.1)$$

The momentum equations on the two axes (2.2-2.3):

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial}{\partial x}(u^2 + p) + \frac{\partial}{\partial y}(u \cdot v) &= \frac{1}{\text{Re}} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \\ \frac{\partial v}{\partial t} + \frac{\partial}{\partial x}(u \cdot v) + \frac{\partial}{\partial y}(v^2 + p) &= \frac{1}{\text{Re}} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \end{aligned}$$

where u and v represent the velocities on the x and y directions, p is the pressure and Re is the Reynolds number. The variables have been non-dimensionalized using the free stream velocity V_∞ , density ρ_∞ , viscosity μ_∞ and the length scale L . The incompressible Navier Stokes equations are difficult to solve for the pressure because there is no time dependent term in the continuity equation. One of the methods used to solve this issue, is to include an artificial compressibility term in the continuity equation.

$$\frac{\partial p}{\partial t} + a^2 \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) = 0 \quad (2.4)$$

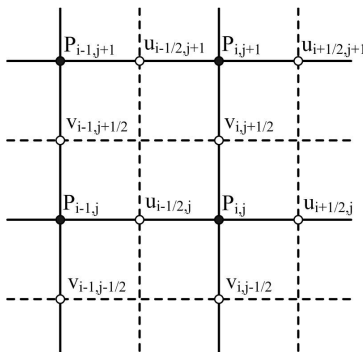


Figure 1. The MAC grid

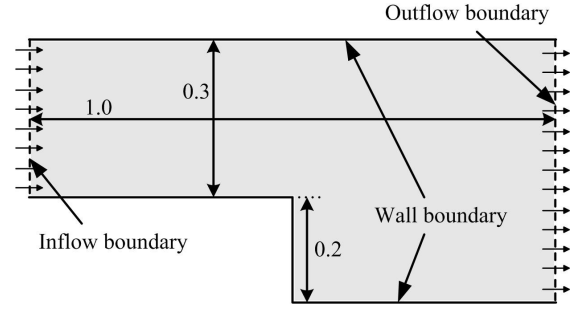


Figure 2. The MAC grid

The term related to the pressure will actually vanish as the scheme progresses to a steady state solution. Consequently, at steady state, the original continuity equation will be recovered.

There are various numerical methods which can be used to solve these equations, like forward time-central space (FTCS), Dufort Frankel, MacCormack, Crank Nicholson, etc. An explicit numerical scheme [14] has been chosen to be implemented on a so called MAC grid (or staggered grid) for improving the stability of the incompressible flow. Fig. 1 shows a detail of the grid. The MAC grid provides stronger coupling between pressure and velocity variables, thus eliminating the spurious oscillations caused by the central space discretizations.

A primary (solid lines) and a secondary grid (dashed lines) can be distinguished on this MAC grid. The pressure is defined on the primary nodes and the velocities on the secondary grid.

A first order difference in time and a second order central difference in space have been used in the discrete domain equations (2.2-2.4).

In order to determine the performance of the algorithm and to conduct the optimizations, the incompressible Navier-Stokes equations have been solved for a backward facing step problem, which is very popular in benchmarking activities. Fig. 2 displays the domain of the flow.

Due to the use of an explicit formulation of the finite difference method, the stability condition, also known as Courant–Friedrichs–Lewy (CFL) condition, has to be satisfied [15]:

$$\Delta t \leq \frac{2 \cdot \Delta \min}{N^{1/2} \cdot a \cdot (1 + \sqrt{5})} \quad (2.5)$$

where $\Delta \min$ is the smallest value between Δx and Δy , N is the number of dimensions in the domain (2) and a is the local speed of sound, which is determined by the maximum velocity.

III. IMPLEMENTATION OF THE INCOMPRESSIBLE NAVIER-STOKES EQUATIONS

A. GPU based implementation

The use of the MAC grid improves the stability of the solution but from an implementation point of view it leads to

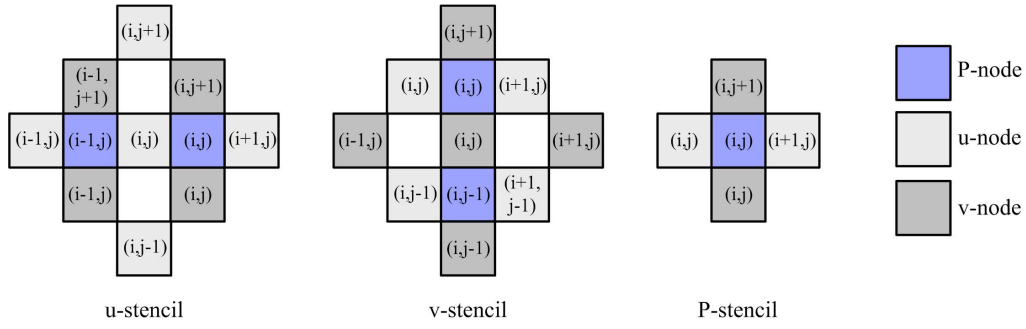


Figure 3. The MAC grid Memory stencils for the three dependent variables

some difficulties when compared with a non-staggered version where all dependent variables are defined at every grid node. A stencil describes or displays in CFD the neighboring grid points/variables which are involved in the equation of a certain grid point (equations (2.2-2.4) will be solved for every point in the grid). These stencils are more difficult to be represented in a MAC grid because the dependent variables are defined at different nodes (on the primary or on the secondary grid). Fig. 3 displays the stencils for the three dependent variables, which have to be determined at every time step. The memory stencil contains eleven different values for the velocities and five different values for the pressure.

The computation has to take place in two different stages at every time step. In the first stage the two velocity components are determined and in the second stage the pressure values are determined. It is not possible to complete the computations in a single stage because the pressure value at time $n+1$ is based on the velocity components at moment $n+1$ and as a result all surrounding velocity components have to be known when the pressure values are computed.

Fig. 4 displays the general work flow used to solve the flow problem on a CPU (left) and on a GPU (right). The discretized versions of eq. (2.3) and (2.4) have a single unknown variable. Consequently, the computations of the values of a dependent variable corresponding to all grid nodes can take place in parallel. Hence, the two double for loops situated inside the outer while loop can be replaced by two kernels. Fig. 5 displays the GPU grid corresponding to the backward facing step problem. Two dimensional blocks of 256 threads (one such block is displayed on the right hand side) are used. The grid structure contains a two dimensional arrangement of blocks (4 blocks in the x direction and 7 blocks in the y direction). The x and y directions were swapped, compared to the equations (2.2-2.4). This shows that the computational grid is completely independent from the kernel grid. Fig. 5 corresponds to a coarse grid containing only 101x51 points. Actually, the backward facing problem is solved on three different grids (a coarse, a medium and a fine grid) in order to show how the performance on the CPU and GPU varies according to the grid size.

For the GPU version of the algorithm the CPU still plays

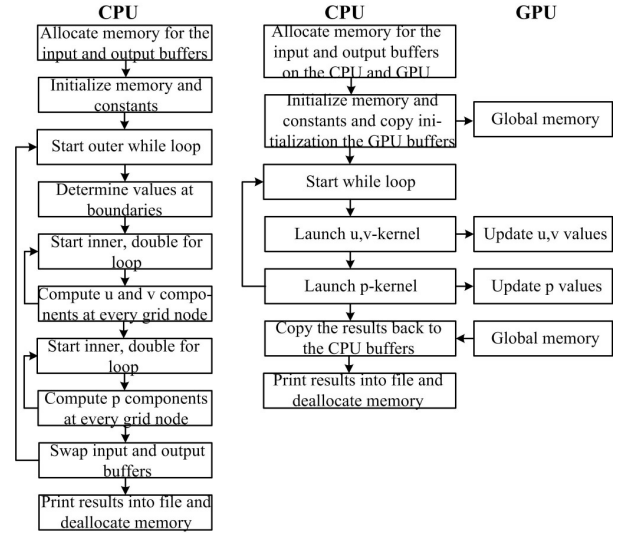


Figure 4. Work flow for the CPU version (left hand side) and the CPU-GPU version (right hand side)

an important role. It allocates the memory for the CPU and the GPU buffers, it initializes the data and then repeatedly calls the two kernels in order to compute the new values of the dependent variables. Consequently, every kernel takes as input the values from the previous time step and computes the values of the next time step.

The reason for not implementing the while loop inside the kernels is given by the impossibility of the GPU to set up a communication between the blocks of the grids [16]. It is not possible to control which blocks get executed first and there is also no possibility to perform synchronization between the blocks. This property is determined by the structure of the GPU which consists of several streaming multiprocessors, on which the blocks are scheduled for execution. The need to

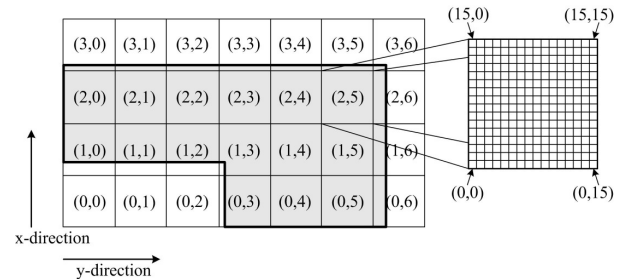


Figure 5. GPU grid corresponding to a coarse grid

establish communication between the blocks would have been given by two reasons:

- To communicate the values of the grid nodes lying at the block boundaries;
- To synchronize before starting the computation of the pressure values and also before a new iteration is started.

B. Optimization activities

The operations described until now have lead to a basic version of the GPU implementation. In order to improve the performance further, various optimization activities have been conducted.

The kernel - eq. (2.2-2.3) - which computes the velocities, has to perform much more computations. The memory stencil required to compute one value of u and one value of v corresponding to a certain node (i,j) of the grid contains 15 values (this number is obtained by combining the first two stencils in fig. 3). Consequently, the first kernel is computationally but also more bandwidth intensive than the second one. The nvcc compiler reports a usage of 23 registers for the first kernel and of only 7 registers for the second one. The CUDA profiler indicates that six times more instructions are being executed for the first kernel than for the second one. By analyzing the summary results provided by the profiler, the limitation factor of each kernel can be determined: the velocities kernel is compute limited and the pressure kernel is bandwidth limited.

First the input and output buffers corresponding to the three dependent variables have been padded on the upper side of the domain so that all of the global memory reads and writes performed by the half-warps are sequential and aligned (fig. 6 shows how unpadded memory can lead to two read operations instead of a single one, thus leading to wasted global memory bandwidth) [17]. For the coarsest grid, this has lead to an increase of 25,5% in memory usage but this disadvantage is easily compensated by the time gained through the reduced number of global memory reads and writes. Then the copy operations between the input and output buffers at the end of each iteration have been eliminated. Instead, the order of the buffers in the parameter lists of the kernel calls is dynamically changed at each iteration. Another change refers to the computation of the

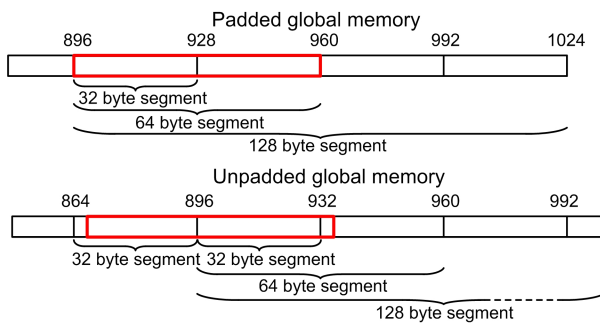


Figure 6. Comparison between coalesced and uncoalesced memory read

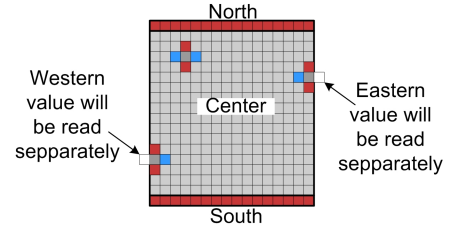


Figure 7. Optimized shared memory pattern

boundary values of the domain, which need special treatment, computation which has also been included in the kernels. Then the usage of shared memory has been considered. Shared memory can reduce the global memory transactions (especially the read operations). Separate tests have been conducted for the two kernels, which have showed that shared memory usage does not improve the performance of the kernel which computes the velocities (because it is compute limited) but it improves the performance of the pressure kernel (because it is bandwidth limited). Fig. 7 shows the optimum shared memory usage pattern for a simplified stencil containing only five points. The idea is to use a shared memory buffer containing also the northern and southern part of the domain, but not the western and eastern parts. This choice has lead to the best performance because the western and eastern parts lead to an increased number of divergent branches inside the kernel which has a negative effect on the overall performance. On the other side the upper and lower parts of the buffer can be populated through only one memory read operation for each segment. During the last optimization activity the optimum thread granularity has been sought. In the basic version of the algorithms each thread computes the value of a single node. Generally one thread may compute several values. Again, because the velocity kernel is compute limited, the best choice is to use one thread to compute one value of u and v . For the pressure kernel, the best choice has been to let one thread compute four p values.

IV. RESULTS

The main goal of the paper has been to obtain an optimized GPU version for the implementation of the artificial compressibility method. This section is focused on displaying the results of this iterative performance improvement. Finally the optimized GPU is compared against a CPU version in order to evaluate the speed-up obtained through our approach.

The tests have been conducted on a NVIDIA GTX260 GPU which consists of 27 streaming multiprocessors. Each such multiprocessor can simultaneously execute the threads of up to eight blocks containing a total of up to 1024 threads, and allows the usage of up to 16K of registers. It has been mentioned in the previous section that the kernel, which computes the values of the velocities, uses 23 registers. This means that simultaneously only two blocks can reside on a multiprocessor (two blocks use almost 12K of registers, three

TABLE I
INITIAL OCCUPANCIES OF THE KERNELS

Kernel	Coarse Grid	Medium Grid	Fine Grid
u,v - kernel	0.518	1.685	6.26
p - kernel	0.259	0.842	3.13

blocks would exceed the hardware limit). On the other hand the kernel which computes the pressure values uses only 7 registers and consequently 4 blocks can simultaneously reside on a multiprocessor. These facts are very important when assessing the potential of the GPU.

The performances of the two versions have been measured for three different grids: a coarse grid (101x51 points, 4000 cells, 28 blocks), a medium grid (201x101 points, 16000 cells, 91 blocks) and a fine grid (401x201 points, 64000 cells, 338 blocks). The cell numbers take into account that only 80% of the domain is used (because of the set up of the backward facing step problem, see fig. 2).

A general performance guideline states that it is necessary to use a grid that ensures a full initial occupancy of the multiprocessors in order to obtain a performance as close as possible to the maximum achievable on the GPU. At least 54 blocks in the grid should be used for the u,v -kernel and at least 108 blocks for the p -kernel. Hence, the initial occupancies of the three grids are the ones displayed in table 1. This shows that only the finest grid guarantees full initial occupancy for both kernels. These facts let us conclude that the GPU performance should increase in terms of Mcells computed per second as the grid becomes finer.

The execution times (expressed in μs) of the various versions of the velocity and pressure kernels are displayed in fig. 8 (for the fine grid). As already stated earlier, shared memory usage and multiple values computed per thread improve the execution time of the pressure kernel but not of

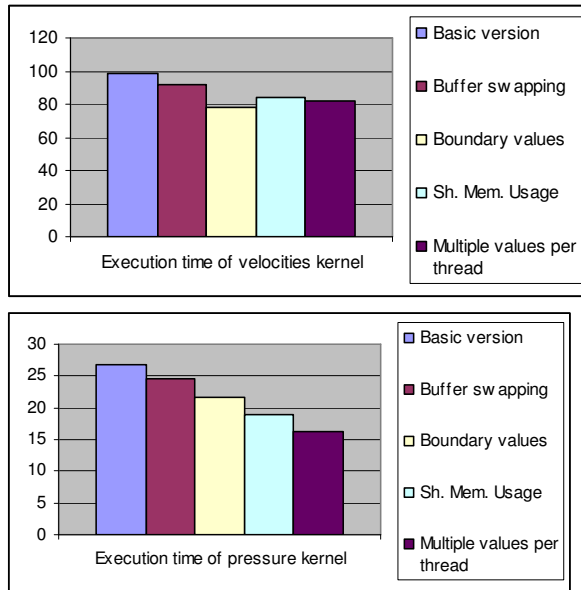


Figure 8. Execution times of the velocities kernel versions (top) and pressure kernel versions (bottom)

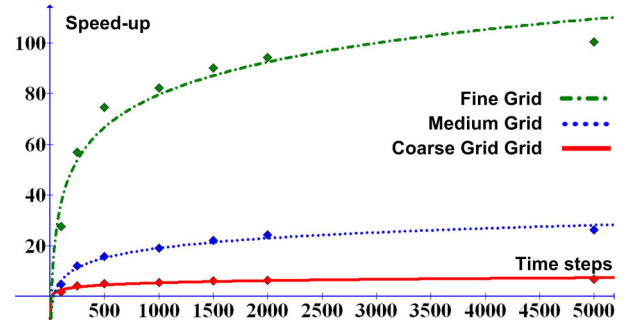


Figure 9. Speed-up obtained by using the GPU version

TABLE II
COMPARISON BETWEEN CPU AND GPU ON THE FINE GRID

Time steps	GPU execution time [ms]	CPU execution time [ms]	Speed-up
100	44.55	1232	27.65
250	71.87	4081	56.78
500	117.33	8741	74.51
1000	208.38	17149	82.29
1500	299.48	27013	90.22
2000	390.61	36834	94.28
5000	937.77	94202	100.45

the velocities kernel. Thus the total improvement is of 20.17% for the velocities kernel and of 39.92% for the pressure kernel. Compared to reference work [7], the improvement is of 14.09% for the velocities kernel and of 34.57% for the pressure kernel. Having obtained this optimized GPU implementation, we have then compared it against a CPU version of the same algorithm on three different grained grids. The evolution of the speed-up value is displayed in fig. 9 for different time step values for the three grids. The speed-up factor increases from under one order of magnitude for the coarsest grid up to exactly two orders of magnitude for the finest grid. The difference between the CPU and the GPU version becomes greater when the grid becomes finer. This is mainly due to the greater parallelism of the finer grid and to the greater initial occupancy. Table 2 displays the data corresponding to the finest grid. As one can see significant performance improvements can be achieved by the usage of the GPU. The more time steps are executed, the greater becomes the improvement. This is caused by the fact that the GPU execution time also includes the memory copies performed at the start and at the end of the computations. When more time steps are executed these memory copies represent a smaller fraction of the total execution time.

Table 3 displays the average execution times of the two kernels on the different grids. Before analyzing these values, it has to be mentioned that the number of cells in the grid is multiplied by four when the switch from a coarser grid to a finer one (coarse to medium or medium to fine) is made. The values in brackets represent the ratio between the execution time on the current grid and the execution time on the next coarser grid. Theoretically, these values should be around

TABLE III
AVERAGE EXECUTION TIMES [μ s] OF THE KERNELS ON THE THREE
GRIDS

Kernel	Coarse Grid	Medium Grid	Fine Grid
u,v - kernel	12.63	29.06 (2.3)	78.45 (2.7)
p - kernel	4.59	7.57 (1.65)	16.12 (2.13)

four because there are four times more cells to compute when the grid is switched. Nevertheless, the values in the table show a smaller increase, which clearly indicates that on coarser grids the GPU does not work at full capacity. Another interesting remark is that the increase in execution time is greater for the u,v - kernel than for the p - kernel. This fact can be explained by recalling the values displayed in table 1 which indicate that the p - kernel reaches a full initial occupancy on finer grained grids than the u,v - kernel.

Finally a comparison of the Mcells/second values for the three grids and for different values of time steps is displayed in fig. 10. The values seem to asymptotically reach a limit as the weight of the memory copy operations goes to zero. As already indicated by the values in table 3, the finer the grid the more Mcells can be computed per second.

V. CONCLUSIONS

The research activity described in this paper has been focused on an optimized GPU implementation for the solution of the incompressible Navier-Stokes equations based on the artificial compressibility method. The GPU computations have been included in two kernels: the velocities kernel and the pressure kernel. Several optimization activities have lead to important performance improvements for the two kernels. The different nature of the two kernels (the first one is compute limited and the second one is bandwidth limited) has lead to different results: the performance of the pressure kernel has been improved through the usage of shared memory and by letting one thread compute several values, whilst the performance of the velocities kernel could not be improved through this techniques. Hence, compared to [7], the performance of the first kernel has been improved by 14.09 % and the performance of the second kernel by 34.57%.

The comparison with the CPU version of the algorithm

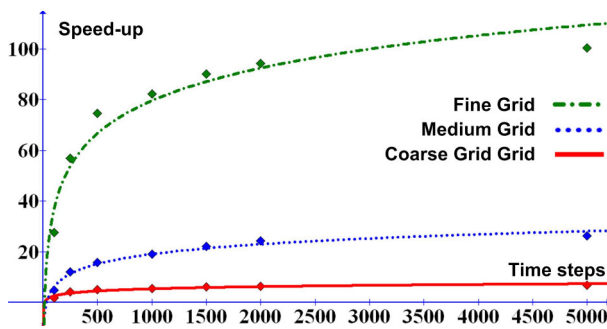


Figure 10. Performance comparison of the optimized GPU version for the three grids

has been conducted on three different grained grids and has indicated a speed up which varies from around one order of magnitude (on the coarsest grid) to up to two orders of magnitude (on the finest grid). This difference can be explained through the poor occupancy obtained on the GPU for the coarser grids.

ACKNOWLEDGMENT

This paper is supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the contract number POSDRU/88/1.5/S/76945.

REFERENCES

- [1] D. Kirk, and W.M. Hwu, *Programming Massively Parallel Processors: A Hands-on Approach*, London: Elsevier, 2010.
- [2] J.D. Owens, M. Houston, D. Luebke, S. Green, J.E. Stone, and J.C. Phillips, "GPU Computing", *Proc. of the IEEE JRPOC*, Vol. 96, pp. 879-884, May 2008.
- [3] G. Chen, G. Li, S. Pei, and B. Wu, "High Performance Computing Via a GPU", *Inter. Conf. on Information Science and Engineering*, pp. 238-241, Dec. 2009.
- [4] C. Zou, C. Xia, and G. Zhao, "Numerical Parallel Processing Based on GPU with CUDA Architecture", *Inter. Conf. on Wireless Networks and Information Systems*, pp. 93-96, Dec. 2009.
- [5] S. Ryoo, C.I. Rodrigues, S.S. Stone, J.A. Stratton, S.Z. Ueng, and S.S. Bagsorkhi, "Program Optimization Carving for GPU Computing", *Journal of Parallel and Distributed Computing*, Vol. 68, No.10, pp. 1389-1401, Oct. 2008.
- [6] M. Blazewicz, K. Kurowski, B. Ludwiczak, and K. Napierala, "Problems Related to Parallelization of CFD Algorithms on GPU, Multi-GPU and Hybrid Architectures", *AIP Conference Proceedings*, vol.1281 pp. 1301-1304, May 2009.
- [7] I. Tanno, K. Morinishi, N., Satofukac, and Y. Watanabea, "Calculation by artificial compressibility method and virtual flux method on GPU", *Computers & Fluids*, Vol. 45, pp. 162-167, June 2011.
- [8] N. Takada, T. Shimobaba, N. Masuda, and T. Ito, "High-Speed FDTD Simulation Algorithm for GPU with Compute Unified Device Architecture", *IEEE Antennas and Propagation Society International Symposium*, pp. 1-4, July 2009.
- [9] F. H. Harlow and J. E. Welch, "Numerical calculations of time dependent viscous incompressible flow of fluid with a free surface", *Phys. Fluids*, vol. 8, pp. 2182-2192, 1965.
- [10] J.F. Wendt, *Computational Fluid Dynamics: An Introduction*, 3rd ed., Berlin: Springer, 2009.
- [11] T.J. Chung, *Computational Fluid Dynamics*, Cambridge University Press: Cambridge, 2002.
- [12] A.J. Chorin, "The Numerical Solution of the Navier-Stokes Equations for an Incompressible Fluid," *AEC Research and Development Report*, New York Univ., Rept. NYO-1480-82, New York, 1967.
- [13] J. Blazek, *Computational Fluid Dynamics: Principles and Applications*, 2nd ed., London: Elsevier, 2007.
- [14] J. Qiwei, D.B. Thomas, and W. Luk, "Exploring Reconfigurable Architectures For Explicit Finite Difference Option Pricing Models", *Inter. Conf. on Field Programmable Logic and Applications*, pp. 73-78 Aug. 2009.
- [15] K.A. Hoffmann, and S.T. Chiang, *Computational Fluid Dynamics*, Vol. 1, Engineering Education System: Wichita, 1998.
- [16] NVIDIA Corporation, *CUDA, Compute Unified Device Architecture Programming Guide v3.1*, 2010.
- [17] NVIDIA Corporation, *CUDA, Compute Unified Device Architecture Best Practices Guide v3.1*, 2010.

Metadata harvesting learning resources – an agent-oriented approach

Mirjana Ivanović*, Dejan Mitrović*, Zoran Budimac*, Milan Vidaković**

* Department of Mathematics and
Informatics, Faculty of Sciences,
University of Novi Sad, Serbia
Email: {mira, dejan, zjb}@dmi.uns.ac.rs

** Faculty of Technical Sciences,
University of Novi Sad, Serbia
Email: minja@uns.ac.rs

Abstract—The paper presents initial ideas and architecture for harvesting learning contents available on the web, with intention to support more qualitative usage and reusability of learning resources. Commonly-known web technologies are considered as good instruments to be used for harvesting and mining embedded metadata. Metadata harvesting/mining through different learning resources is by its nature distributed, and represents a good basis for employing agent technology. Based on previously implemented agent architectures at our university and extensive usage of e-learning tools we propose agent architecture for metadata harvesting in the context of learning resources.

I. INTRODUCTION

THE e-learning market has already been cluttered with different tools and platforms intended to be used within learning communities. These tools can usually be used individually by learners, or by mash-ups, or as systems which involve a number of separate tools. As a consequence several challenging issues encourage further research [2] and we slightly modified some of them for our purposes and research plans:

- **Reusability**: Learning content is going to be split up and reused in different contexts.
- **Interoperability**: Multiple learning systems/tools can communicate with each other in order to share, mix and propose some parts of existing learning material as additional useful resources.
- **Accessibility**: Learning material can be accessed by several different learning tools. Besides that, appropriateness of content must be ensured.

Nowadays, since there are a lot of electronic versions of educational material in private and organized repositories (e.g. Open Educational Resources – OER, <http://www.oercommons.org/community>), it is a strong

suggestion to make the content, as well as the accompanying metadata, widely available for harvesting. Currently, there are a number of organizations operating in the area of eLearning standards: Dublin Core Metadata Initiative, the IEEE, the IMS Global Learning Consortium, the Alliance of Remote Instructional Authoring and Distribution Networks for Europe, the Aviation Industry CBT Committee, the Advanced Distributed Learning Initiative (SCORM - Sharable Content Object Reference Model as a collection of standards and specifications for web-based e-learning), etc. Therefore, it is important to use some of them in order to prepare e-learning material and make it useful for wider audience.

Current educational metadata standards and concepts focus on content-centered approaches. To perform any kind of activities (prepare learning material, implement and use automatic/semi-automatic tools for retrieving existing learning repositories) it is necessary to have the following generally recognized suggestions in mind:

- For the content: there are a lot of possibilities, such as HTML / Microsoft Office / OpenOffice / PDF documents, MPEG video clips, MP3 sound files, etc. Having in mind *open* educational resources, it is wiser to publish the content in as open format as possible.
- For the metadata: there is a wide range of possibilities as well, but recently it seems that Learning Object Metadata (LOM) (defines approximately 70 elements for describing general, technical, educational and other aspects) or Dublin Core (DC) (more generic in nature) are prevalent. Some authors also use other standards such as MPEG (for audio-visual material) or some other special purpose formats. In such cases, a mapping to LOM or DC is strongly suggested to enable interoperability with learning specific tools/infrastructures. Metadata are

The work is partially supported by Ministry of Education and Science of the Republic of Serbia, through project no. OI174023: "Intelligent techniques and their integration into wide-spectrum decision support"

typically expressed in a machine-readable format, most commonly in XML - the Extensible Markup Language, and RDF - the Resource Description Framework. They allow describing relationships and properties about resources.

Standards focus on search and re-use of learning material, frequently called content items, learning objects or other learning components, and a lot of them are focused on and use LOM. The main motivation for the creators of educational material is to use appropriate standards and metadata in order to make existing huge learning repositories useful and re-usable. Having these factors in mind, our architecture is designed to facilitate software agents that automatically and dynamically recognize and propose possible useful learning objects as additional learning material for a student. In this way, our architecture is intended to perform the scaffolding way of teaching. In future, we surely have to have in mind the composition of harvested learning components in form of additional lessons, personalized as much as possible.

The important concept we have incorporated in our proposed architecture is *harvesting*. Harvesting allows a software agent to collect different resources (content or metadata) from remote repositories. For example, the *OAI-PMH* harvesting protocol (based on a specific query) enables the requesting agent to retrieve precise set of metadata or content within the *Networked Digital Library of Theses and Dissertations* (NDLTD) [18]. After the harvesting process is finished, the harvester has all the metadata or content available, so there is no further need to contact the harvested repositories. The key concern is that this approach allows third parties to collect the metadata and content from a repository.

The other side of the problem is that, rather frequently, the distant knowledge acquisition process can cause worse effectiveness than traditional teaching. One of main reasons for this phenomenon is students' inability to select essential information from "informational noise" and/or the fact that educational materials have been prepared in an inappropriate way.

Open educational resources (OERs) are "digitized materials offered freely and openly for educators, students and self-learners to use and reuse for teaching, learning and research." [12]. The problem of being able to find an appropriate OER on the Internet is evidently still an open and an important concern. Recently, research groups all over the world have offered new search solutions and new tools specifically developed for search and discovery of appropriate educational materials.

To overcome the wide range of problems connected to the preparation, administration, innovation, and reusability of learning repositories and courses, a solution to improve efficiency of knowledge acquisition process, could be application of an intelligent multi-agent system. In this paper we propose an agent-based architecture as a support for

automatic collection and suggestion of additional appropriate teaching material for students. We designed our own architecture based on our previous research in the domains of agent languages [13], mobile agents [6] and tutoring systems [1]. Such proposed system will offer us better possibility to tailor it to our local educational needs and further improvements.

The rest of the paper is organized as follows. Section 2 brings remarks on related work. In section 3 architecture of our system is presented. Section 4 concludes paper and gives possible directions for future work.

II. RELATED WORK

Over the years, the agent technology has been suggested as a promising tool for extending intelligent tutoring and course recommendation systems; see, for example, EducMAS [10], MathTutor [11], and ABITS [17]. Instead of using harvesting agents, however, these systems usually offer metadata authoring tools to teachers, who then themselves prepare and organize the course material. Nonetheless, some of these concepts, combined with our own experience in developing tutoring systems [1] will be used for future improvements of the proposed architecture.

One of the most fundamental examples of using agents for metadata harvesting was demonstrated in [20]. Their system employs crawler agents in order to parallelize the process of extracting metadata from web-based content, such as *PDF* documents. The results of this process are stored in XML format and offered to conventional search engines, trying to improve their effectiveness.

Agent Based Search System (ABSS) [21] and the AgCAT system [5] represent more complete agent-based frameworks for harvesting learning objects. Both systems operate by using agents that track changes in remote LO repositories (addition of new LOs, deletion of existing LOs, and so forth), and update their local repositories in response. LOs stored in these local repositories can then be accessed by using sophisticated, embedded search engines. Out of the two systems, ABSS seems to be more powerful and feature-complete. For example, it uses *Personal Agents* to offer personalized search capabilities, customized presentation of the available material, etc.

The general architecture, as well as the types and roles of agents used in our system are similar in nature to ABSS and AgCAT. However, while metadata harvesting agents in ABSS and AgCAT constantly monitor the changes in remote repositories, harvesting agents in our system are dispatched to remote LO repositories in response to an automatically detected decline in student's performance. That is, our system includes intelligent, BDI-style agents that monitor the student's progress through a course, and obtain new learning material as needed. This newly harvested material is processed using intelligent data mining techniques and then *pushed* to the student, rather than the student having to *pull* it using a search engine.

BRENHET [8] is another example of an agent-based LO search engine. It utilizes several types of agents, such as *Statistical*, *Sort*, and *Optimize*, layered on top of each other in order to provide the best LO results based on the user's search query. The general conclusion of the authors of BRENHET is "that the agent-based architecture is ideal for resolving the problem of federated searches in heterogeneous repositories, due to the adaptation and learning facilities of the actual agents" [8].

Obviously, there is a lot of ongoing research on the subject of agent oriented e-learning systems. Although the existing systems do include sophisticated metadata harvesting approaches, they do not seem to incorporate agents in both the metadata harvesting, and the course recommendation levels. This is the goal of our proposed architecture.

III. SYSTEM ARCHITECTURE

The agent technology has been evolving rapidly during the last two decades, employing and incorporating agents in different environments and real-world domains. During this period, different types of agents have been defined and recognized [4]. Among many other things, agents are capable of acting as virtual assistants for their (human) users, collecting specific data and presenting it in accordance to the user's preferences, realize complex tasks on behalf of their users, and adjust their own behavior to suit the user's needs. When incorporated in distant or e-learning systems, agents can significantly reduce duties of the human supervisor and help in selecting the appropriate course content [16].

Considering the proven usefulness of different kinds of software agents, as well as our previous experiences in developing and implementing agent languages and frameworks, we are going to unify our efforts and propose an agent-based architecture for harvesting learning materials. Our goal is to support scaffolding learning and to develop a system that can offer appropriate additional learning material for students in automatic and semi-intelligent way. Before we propose the architecture, we will shortly introduce the types of agents which will be used in our system.

A. Agent technology used

The proposed system will be completely built using 2 of our own agent development environments: EXtensible Java-based Agent Framework (XJAF) [7], and Adaptable Java Agents (AJA) [13].

XJAF is a FIPA-compliant [9] multi-agent system. It consists of a set of loosely-coupled modules, called managers, each of which is responsible for handling a distinct part of the overall agent management process. Managers can be added dynamically (as plug-ins) and are recognized solely by their interfaces, allowing custom implementations. Multiple distributed instances of XJAF can

be connected to each other, forming a tree-like network structure, with the primary instance representing the root of the tree [14].

AJA is a tool for programming of multi-agent systems developed at our Department [13]. The main goal of the research leading to the development of AJA was to create an agent-development tool that introduces new agent concepts at the programming language level. An additional goal was to use AJA in implementation of enterprise-scaled, real-world multi-agent systems. In order to fulfill these goals, the original approach taken in AJA was to combine Java with higher-level agent constructs and AI-components.

Regarding the agent classifications [4], AJA agents can be classified as: software, static, middle- and big-sized agents, agents that can learn, with hybrid architecture, and for both cooperative and competitive systems. An AJA agent consists of the following parts:

- *Beliefs*: primitive values, data structures, and the values generated by AI components. Beliefs define agent's internal state and represent the information an agent has about the world.
- *Actions*: blocks of code that can be seen as atomic agent actions. AJA provides the synchronization mechanism for the parallel actions execution.
- *Reflexes*: condition-action(s) pairs. Reflex is a reactive component. The first part of AJA reflex is activation condition, whose value determines whether the second part of the reflex should be executed or not. The second part is a sequence of action invocations.
- *Negotiations*: represent the inter-agent communication:
 - *Requesting negotiation*: used when agent initiates the communication.
 - *Responding negotiation*: used when agent responds to the request.
 - *WWW negotiation*: used for user-agent communication via Internet. Agent initiates the communication with other agent(s) with a requesting negotiation. An agent uses a responding negotiation to respond.
- *Initialization part*: actions that execute after the agent has been activated.

AJA can be extended and improved by including new AI constructs, resulting in more intelligent AJA agents. These extension mechanisms can include a case-based reasoning component, a knowledge base and an inference engine component, a fuzzy logic component, a planner component, etc. For each new component, corresponding language constructs have to be defined and implemented.

B. System architecture

Some authors have recognized that intelligent agents can

be a vehicle for AI-related technologies. If intelligent agents and agent-based systems become a mainstream, then AI can indirectly also become a mainstream. But, even under such circumstances that average software developer starts to think in agent terms, they still miss AI background. A programmer cannot program planning component of an agent, if he/she does not know how to do it. Because of the fact that pluggable AI components are still not generally available, an alternative approach is to embed AI techniques into agent-programming language. For example, a language could contain different constructs that support reasoning and inference so a programmer can use them in his/her agents to reach intelligent behavior. Such approach is used in AJA (Adaptable Java Agents) programming environment, mentioned in previous section.

For the purpose of our architecture, AJA has to be enhanced by a knowledge base component and an inference engine component. These extensions will help in intelligent and precise selection of appropriate learning objects, and allow harvesting agents to collect additional learning material that is based of the student's learning preferences achievements.

According to abovementioned, as well as our previous research in the domain of agent technology, our proposition for harvesting agent-based learning architecture is shown in Figure 1. A more detailed description of major components of the system is given next.

SPM – Student Performance Module – constantly monitors student's learning activities and collects all data from the learning process. It records necessary data about the student's progress through the course, the quality and quantity of acquired knowledge, weaknesses, drawbacks and

advancements. SPM is then able to use this data and to start the process of harvesting when appropriate, in order to obtain additional learning material for the student. SPM can start the harvesting process either on explicitly expressed student's wishes/needs, or automatically, under specific conditions and circumstances, based on the results of the learning progress analysis. It prepares and sends the appropriate information and requests for additional learning objects for further analysis to Central Intelligent Agent.

HA – Harvesting agent – performs the metadata harvesting and lecture indexing, in response to the LO metadata harvesting request sent by the Central Intelligent Agent. HA collects appropriate metadata from a particular remote repository, selects appropriate learning objects, and reports its results back to CIA. By default, a HA relies on Simple Query Interface (SQI) for interoperability with different OERs. SQI was designed as a query language for learning object repositories. It is a tool that facilitates interoperability between open educational resources and the applications that use them [19]. The SQI specification has largely facilitated the interoperability between resources, as well as between them and search engines. For a LO repository that does not offer an SQI, HAs can be customized to use the protocol supported by the given repository. In order to get better performances and quality of harvested resources, several harvesting agents can be employed at the same time; each of them would be given a clear specification of what to harvest, e.g. in order to avoid duplication of harvested material.

CIA – Central Intelligent Agent – the most important component of the system. CIA assembles the metadata and learning objects received from harvesting agents, and, in cooperation with SPM, completes final intelligent sieving of them. It makes decisive selection of most suitable and appropriate harvested components into additional teaching material and sends the results back to the student.

When it receives an explicit request from SPM, CIA initiates the harvesting procedure by sending appropriate requests to HAs. These requests consist of a precise specification necessary for HAs to find additional learning objects. After the harvesting process is finished, CIA collects gathered information from dispatched HA(s) and compares and consolidates possibly huge amount of data about LOs that can be proposed to the student. Based on quality, similarity, and appropriateness to original learning material the student has used, it selects the most suitable learning components, does more intelligent assessment and final selection. Also, when CIA completes final intelligent sieving of learning components and makes ultimate selection of the harvested material it sends propositions for additional learning objects back to the student. The analysis of collected data allows for general evaluation and assigning to one of three levels (e.g. basic, intermediate, and advanced). CIA can follow paths through the learning material as proposed by the SCORM 4 [3] standard, if it is present.

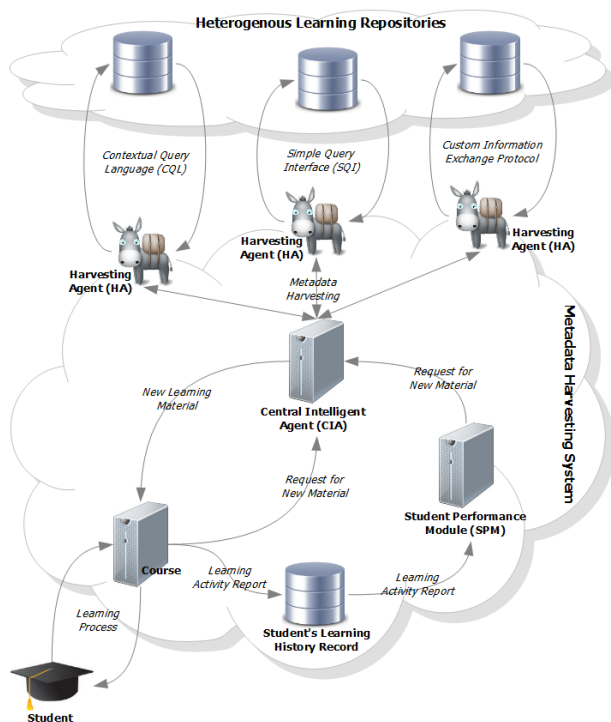


Figure 1. Agent-based harvesting architecture.

This agent will be implemented as AJA agent, which will be enhanced by appropriate knowledge base and an inference engine component. Such component will be able to compare different harvested learning objects and make final decision and selection of most appropriate for further student's learning activities.

Each HA can use several specialized services to access LO metadata and courses – *LOMService*, *PDFService* and *PPTService*. These services are used to harvest metadata (if existing), or to index course material contained in *PPT* or *PDF* files. The main difference between a search engine (i.e. Google) and a HA is that HA can obtain identity in order to access web material that is not public. HAs reside in XJAF nodes and are activated by the CIA by receiving the request for harvesting. Sequence diagram for the proposed architecture is shown in Figure 2.

C. Possible applications of the proposed architecture

During last several years at our Department we have been developing several e-learning environments and tools and have been using and testing them in different courses [22]. *Mag* is one of such system, designed to implement distance education with experiences from traditional way of learning programming languages. Architecture of the *Mag* system contains several modules which are standard for such kind of learning environments:

- **Student Module** is used for student modeling. It stores information that is specific to each individual learner. At a minimum, such module tracks how well a student has mastered the material being taught. A possible addition to this is also to record misconceptions. Since the purpose of the student module is to provide data for the pedagogical module of the system, all of the information gathered should be available for

the pedagogical module.

- **Pedagogical Module**. This component provides a model of the teaching process, by controlling when to review, when to present a new topic, and which topic to present. Student module should be used as input to this component, so the system's pedagogical decisions reflect the different needs of each student.
- **Domain Knowledge**. This component contains knowledge from the domain of teaching. Generally, it requires significant knowledge engineering to represent a domain so that other parts of the pedagogical module can access it.
- **Communications Module**. Interactions with the learner, including the dialogue and the screen layouts, are controlled by this component.
- **Expert Module**. This module is similar to the domain module. However, it is more than just a representation of the data; it is an appropriate way of presenting knowledge to the student. Most commonly, it is a part of the system that is capable of solving problems from the domain. By using an expert module, the tutor can compare the student's solution to the expected solution, pinpointing the subject in which the learner had difficulties.

Mag can provide the student with a more convenient and efficient way to advance through a Java programming course using prepared LOs and test his/her knowledge using various types of questions and tasks. This functionality provides ways to constantly build, improve, and extend the student module (in the architecture proposed in this paper, **Student Module** corresponds to **Student Performance Module**). Appropriate student module makes easier

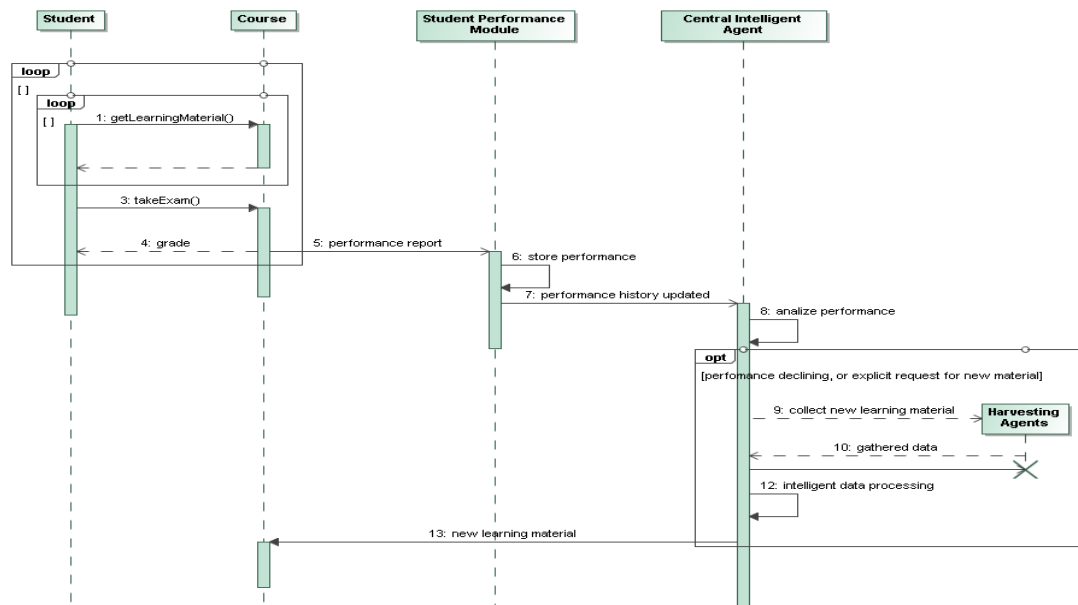


Figure 2. Sequence diagram of the harvesting architecture.

adaptation of the remaining process of teaching and testing to every particular student. The applied system architecture brings several benefits to practical use and it provides possibilities for easier integration of *Mag* system into wider learning environment as well as other components in *Mag* environment. These possible ways of integration can bring additional potential in order to obtain multifunctional learning management system.

Harvesting agent-based learning architecture we proposed in previous subsection could be rather smoothly incorporated in the *Mag* environment and used to increase student's learning performances by obtaining more qualitative and appropriate additional learning objects available through a network.

IV. CONCLUSION AND FUTURE WORK

Nowadays, there are a lot of e-learning environments and applications of such systems in everyday life. However, it is necessary to upgrade such systems with additional functionalities, such as personalization, recommendation, and more productive usage of teaching materials available all over the Internet. Existence of various repositories of educational material must be additional requirement and challenge for research community in order to re-use such learning resources in more qualitative ways.

According to that, our proposed architecture has to facilitate software agents to automatically and dynamically recognize, collect and suggest additional learning material for a student. Harvesting agents represent an important concept that has been incorporated into the proposed architecture. They have the task of exploring heterogeneous repositories and suggesting the most adequate learning material to students. When compared to conventional search engines, harvesting software agents offer the following immediate advantages:

- The ability to access protected resources by, for example, using the student's virtual identity.
- The ability to follow paths through the learning material as proposed by the SCORM 4 standard (if supported by the target system).

These features significantly improve the quality of the learning process and enable efficient re-use of existing learning material.

The proposed architecture is based on our own experience. Our previously developed technologies offer the required functionality for realizing the metadata harvesting system. In the long run, this approach will also allow us to integrate new, possibly third-party, components and functionalities in a straightforward fashion.

REFERENCES

- [1] A. Klačnja-Milićević, B. Vesin, M. Ivanović, and Z. Budimac, "Integration of recommendations and adaptive hypermedia into Java tutoring system". *ComSIS Journal*, 8(1):211-224, January 2011.
- [2] A. Soyulu, S. Kuru, F. Wild, and F. Mödritscher, "E-Learning and microformats: a learning object harvesting model and a sample application". In: *1st International Workshop on Mashup Personal Learning Environments (MUPPLE08) at The 3rd European Conference on Technology Enhanced Learning (EC-TEL 2008)*, Maastricht, The Netherlands, 2008.
- [3] ADLnet – Advanced Distributed Learning Homepage, <http://www.adlnet.gov>
- [4] C. Badica, Z. Budimac, D. Burkhard, and M. Ivanović, "Software agents: languages, tools, platforms", *ComSIS Journal*, 8(2), May 2011, (in print).
- [5] C. F. Barcelos, and J. C. Gluz, "An agent-based federated learning object search service". *Interdisciplinary journal of e-learning and learning objects*, 7:37–54, 2011.
- [6] D. Pešović, "A high-level language for defining business processes". *PhD Thesis*, University of Novi Sad, Novi Sad, 2007.
- [7] D. Pešović, M. Vidaković, M. Ivanović, Z. Budimac, and J. Vidaković, "Usage of agents in document management". *ComSIS Journal*, 8(1):193-210, January 2011.
- [8] F. de la Prieta, and A. B. Gil, "A multi-agent system that searches for learning objects in heterogeneous repositories". In *PAAMS special sessions and workshops*, 71:355-362, 2010.
- [9] FIPA Homepage, <http://www.fipa.org>
- [10] I. S. Gago, V. M. Werneck, and R. M. Costa, "Modeling and educational multi-agent system in MaSE". In *Proceedings of the 5th international conference on active media technology, AMT'09*, pp. 335-346, 2009.
- [11] J. Cardoso, B. Guilherme, L. B. Frigo, and E. Pozzebon, "MathTutor: a multi-agent intelligent tutoring system". In *First IFIP conference on artificial intelligence applications and innovations*, pp. 231-242, 2004.
- [12] J. Hylén, "Giving knowledge for free: the emergence of open educational resources". *Paris, France: OECD Publishing*, p. 30, DOI: 10.1787/9789264032125-en. <http://www.sourceoecd.org/education/9789264031746>. Retrieved 2010-12-03
- [13] M. Badonski, M. Ivanović, and Z. Budimac, "Adaptable Java Agents (AJA): a tool for programming of multi-agent systems". *SIGPLAN Notices*, 40(2):17-26, February 2005.
- [14] M. Vidaković, B. Milosavljević, Z. Konjović, and G. Sladić, "Extensible Java EE-based agent framework and its application on distributed library catalogues". *ComSIS Journal*, 6(2), UDC 004.428, DOI: 10.2298/csis0902001V, ISSN: 1820-0214, December 2009.
- [15] M. Vidaković, G. Sladić, and M. Zarić, "Metadata harvesting using agent technology". In *Proceedings of the 8th IASTED International conference on software engineering and applications (SEA 2004)*, Cambridge, USA, pp. 489-493, November 9-11, 2004.
- [16] M. Woda, and M. Piotr, "Distance learning system: multi-agent approach". *Journal of Digital Information Management*, 3(3):198-201, September 2005.
- [17] N. Capuano, M. Marsella, S. Salerno, "ABITS: an agent based intelligent tutoring system for distance learning". In *Proceedings of the International Workshop in adaptive and intelligent web-based educational systems*, pp. 17-28, 2000.
- [18] NDLTD Homepage, <http://www.ndltd.org/>
- [19] S. Otón, J. R. Hiler, E. García, A. García, L. de-Marcos, A. Ortiz, J. A. Gutiérrez, J. J. Martínez, J. M. Gutiérrez, and R. Barchino, "A proposal to improve the Simple Query Interface (SQI) of learning objects repositories". In *10th IEEE International Conference on Advanced Learning Technologies, ICALT*, pp. 64-66, 2010.
- [20] S. Sharma, and J. P. Gupta, "A novel architecture of agent based crawling for OAI resources". *International journal of computer science and engineering, IJCSE*, 2(4):1190–1195, 2010.
- [21] T. Orzechowski, "The use of multi-agents' systems in e-learning platforms". In *Siberian conference on control and communications, SIBCON '07*, ISBN: 1-4244-0346-4, pp. 64-71, 2007.
- [22] Z. Budimac, M. Ivanović, Z. Putnik, Ž. Komlenov, I. Pribela, A. Klačnja-Milićević, and B. Vesin, "E-Learning at the Department of mathematics and informatics – from the first steps to elements of Web 2.0". In *Proceedings of the conference "E-Learning, Way Towards Society of Knowledge"*, Belgrade, Serbia, pp. 94-98, 2010.

Model Predictive Control of the Interleaved DC-DC Boost Converter

Petros Karamanakos, *Student Member, IEEE*, Georgios Papafotiou, *Member, IEEE*, and Stefanos N. Manias, *Fellow, IEEE*

Abstract—This paper extends the recently introduced approach to the modeling and control design in the framework of model predictive control of the dc-dc boost converter to the dc-dc parallel interleaved boost converter. Based on the converter's model a constrained optimal control problem is formulated and solved. This allows the controller to achieve (a) the regulation of the output voltage to a predefined reference value, despite changes in the input voltage and the load, and (b) the load current balancing to the converter's individual legs, by regulating the currents of the circuit's inductors to proper references, set by an outer loop based on an observer. Simulation results are provided to illustrate the merits of the proposed control scheme.

I. INTRODUCTION

DC-DC converters are widespread power electronics circuits with advantageous features, such as small size, light weight, and high efficiency. They are used in applications where dc-dc conversion between different voltage levels is needed, like dc power supplies and dc motor drive systems [1]. Despite the fact that the switch-mode dc-dc conversion is a mature technology, the problems associated with these applications and their closed-loop controlled performance still pose theoretical and practical challenges. The emergence of new applications with more demanding performance requirements, coupled with the development of novel computational control techniques, made possible by the increased computational power of the available control platforms, leads to new approaches to the control problem.

In this paper we aim to tackle the problem of the control of the dc-dc parallel interleaved boost converter from the perspective of the model predictive control (MPC). MPC has successfully established itself as a systematic control method with numerous applications, from the automotive [2], power systems operation [3] and fuel cell control [4] to power electronics industry. In the latter the MPC has been implemented for several topologies [5]–[11]. Specifically, in the field of dc-dc converters a number of publications have appeared [12]–[14] where the method demonstrated a significant potential in achieving important performance improvements. The main advantage of this method is its straightforward design procedure. It is based on the discrete-time predicted model of the system, enhanced by an objective function composed of terms that penalize the deviation of the predicted system behavior, and a set of inputs which

are derived from the assigned control objectives. In each sampling instant, the input resulting from the minimization of the assigned objective function that yields the best predicted performance of the system is chosen. This input is called optimal.

Regarding the material presented in this paper, the chosen topology is the dc-dc parallel interleaved boost converter. The converter is consisted of two parallel legs, in respect to the power supply and the common load. The benefits arising from the parallelization are many [15]. To name the most important are the cost reduction and the higher reliability. For the first, the components used could be less expensive, since the load current is equally distributed to the individual converter legs, which means that the power stages can handle lower currents. For the latter, since the failure of a parallel leg does not disrupt the converter's operation, this increases the fault tolerance of the topology. However, in reality the parallel legs cannot be identical due to several reasons, such as components mismatching, both in the inductance and in the power semiconductors, and parameter variations caused by the temperature increase. Thus, the power switches should be turned on and off in such a way that the load current will be shared equally to both legs. This is the first control objective. Furthermore, the output voltage of the converter should be regulated to a desired value despite changes in the input voltage and the load. This is the second control objective.

The controller is designed by taking into consideration the control objectives mentioned above. The control method introduced here comprises the two loops that are commonly used in the control operation of the dc-dc converters; an inner loop which drives the currents of the individual legs to specified references, and an outer loop which calculates these references based on the regulation of the output voltage to a desired voltage level. However, as proposed in [16], since the present work extends the research done there for the dc-dc boost converter, the difference in the approach of how these loops are employed is the following: the inner current control regulation problem is posed in the MPC framework, while in the outer loop the inductor current references are derived from the converter's power balance expression, where the estimated load current is used. In order to go into more detail, as shown in [12]–[14], and [17] the inner current control loop is formulated using hysteresis bounds for the parallel legs inductor currents, in the proposed MPC modeling, using soft constraints.

The control method presented here has several benefits. The most prominent are the very fast current dynamics that

P. Karamanakos and S. N. Manias are with the Electrical Engineering Department, National Technical University of Athens, Athens 10682, Greece (e-mail: petkar@central.ntua.gr, manias@central.ntua.gr).

G. Papafotiou is with ABB Corporate Research, Baden-Dättwil 5405, Switzerland (e-mail: georgios.papafotiou@ch.abb.com).

MPC can deliver, combined with the guaranteed robustness and stability of the method, since the closed-loop performance is independent from the operating point. Moreover, excessive iterations and tuning are avoided due to its straightforward implementation, since the control objectives are expressed in the objective function of the optimal controller. In addition, the proposed approach rejects disturbances in the input voltage and the load. Furthermore, due to the model-based nature of the controller and its design simplicity, it can be easily extended to interleaved topologies with more individual legs. On the other hand, the proposed control scheme shows some inherent drawbacks. The use of longer prediction horizon increases exponentially the computational power needed, while the absence of a modulator and the direct manipulation of the converter switches imply a variable switching frequency.

This paper is organized as follows. In Section II, the topology of the considered converter is presented, showing the nonlinear continuous-time state space model, and the derivation of the discrete-time model of the system that is suitable for the controller. In Section III, an optimal control problem incorporating the appropriate control objectives is formulated. The way is explained in which the load estimation setup is employed. Section IV provides simulation results to show the performance of the proposed control scheme. Finally, the paper is summarized in Section V, where conclusions are also drawn.

II. MATHEMATICAL MODEL

A. Continuous-Time Model

The dc-dc parallel interleaved boost converter (Fig. 1) consists of two legs, each of which includes one inductor, L_n , and the corresponding internal resistor R_{L_n} , and the power semiconductors; the controllable switch, S_n , and the dually operated diode, D_n , with $n = 1, 2$. The mathematical representation of the converter in the continuous-time state space is given by the following equations [18]

$$\frac{dx(t)}{dt} = (A_1 + A_2 \cdot u(t)) \cdot x(t) + B \cdot w(t) \quad (1a)$$

$$y(t) = C \cdot x(t) \quad (1b)$$

where

$$x(t) = [i_{L_1}(t) \ i_{L_2}(t) \ v_o(t)]^T, \quad w(t) = [v_i(t) \ i_o(t)]^T,$$

$$A_1 = \begin{bmatrix} -\frac{R_{L_1}}{L_1} & 0 & -\frac{1}{L_1} \\ 0 & -\frac{R_{L_2}}{L_2} & -\frac{1}{L_2} \\ \frac{1}{C_o} & \frac{1}{C_o} & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} \frac{1}{L_1} & 0 & 0 \\ 0 & \frac{1}{L_2} & 0 \\ 0 & 0 & -\frac{1}{C_o} \end{bmatrix},$$

$$B = \begin{bmatrix} \frac{1}{L_1} & 0 \\ \frac{1}{L_2} & 0 \\ 0 & -\frac{1}{C_o} \end{bmatrix}, \quad C = [0 \ 0 \ 1], \quad \text{and } u = \begin{bmatrix} 0 & 0 & u_1 \\ 0 & 0 & u_2 \\ u_1 & u_2 & 0 \end{bmatrix}$$

where C_o is the filter capacitance, $i_{L_n}(t)$ the n -inductor current, $i_o(t)$ the load current, $v_o(t)$ the output voltage, and $v_i(t)$ the input voltage. Finally the variable u_n defines the switching position, i.e. $u_n = 1$, when the switch S_n is on,

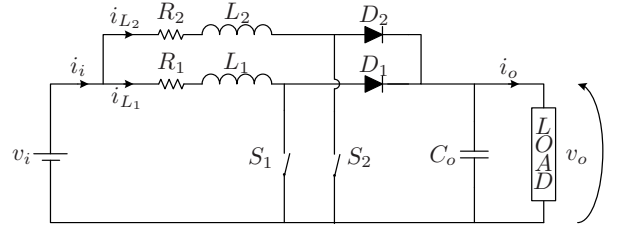


Fig. 1: Topology of the dc-dc parallel interleaved boost converter.

and $u_n = 0$, when the switch S_n is off. Thus, the converter features 2^n operation modes with equal affine equations.

B. Discrete-Time Model

Starting from the continuous-time state space model developed in the previous section, the goal of this section is the derivation of a mathematical model of the converter suitable for the controller. Since MPC is a discrete-time controller the model is discretized using first-order approximation. This leads to the following discrete-time state space model, defined as

$$x(k+1) = (A_{1,d} + A_{2,d} \cdot u(k)) \cdot x(k) + B_d \cdot w(k) \quad (2a)$$

$$y(k) = C_d \cdot x(k) \quad (2b)$$

where

$$x(k) = [I_{L_1}(k) \ I_{L_2}(k) \ V_o(k)]^T, \quad w(k) = [V_i(k) \ I_o(k)]^T,$$

$$A_{1,d} = (I + A_1 \cdot T_s), \quad A_{2,d} = A_2 \cdot T_s, \quad B_d = B \cdot T_s,$$

$$C_d = C, \quad \text{and } u(k) = u$$

where capital letters denote the discrete time-varying physical quantities, and T_s the sampling time.

III. CONTROLLER DESIGN

In this section the design of the nominal controller is presented. The controller design is done in three steps. Firstly, an objective function has to be defined. Hence, the state variables under consideration are predicted in each sampling time, from the affine mathematical models of the converter, derived from the respective switching states. Then, these predicted variables are compared to their reference values, and the calculated errors, bounded by soft constraints, are the optimization variables of the objective function. Once the objective function is formulated, the switching state that minimizes it, is applied to the next sampling period. Finally, a second order Luenberger observer is designed in order to estimate the load current under any kind of circumstances.

A. Control Objectives

The main control objective of the parallel dc-dc boost converter is to command the switching semiconductors so as to achieve an output voltage equal to its desired value, despite measurable changes in the input voltage and unmeasurable variations in the load. Given the input voltage $v_i(t)$, the dc component of the output $v_o(t)$ should be regulated at its reference value v_o^* with suitable choice of the control action, which means by appropriate switching. Moreover, the controller's performance in terms of current sharing is also

taken into consideration. Thus, the controller should act on the switches in such a way that the average input current is distributed equally to both legs of the converter.

B. Constrained Optimal Control

MPC is considered as a controller that has been implemented successfully in the process industry. In a few words, the procedure of the MPC is the following: An objective function is defined, based on the system's dynamics and the problem's constraints. This function is subjected to hard and/or soft constraints, depending on the configuration of the topology, and the performance of the controller that the designer wants to achieve. In a second step, the control action is obtained by minimizing this objective function at each step, over a finite prediction horizon. Such horizon recedes by one sampling interval respectively (receding horizon strategy) [17].

In the control method introduced here, the inner current control regulation problem is formulated and solved as an MPC problem. In order to penalize the deviation of the inductor currents, hysteresis bounds are introduced that are used as soft constraints. These bounds are defined by the maximum and minimum values of the reference inductor currents. In order to describe the degree of violation of the predefined constraints, which are weighted with p_{a_n} and $p_{b_n} \in \mathbb{R}^+$, a slack variable $e_n(k)$ is introduced [19].

$$e_n(k) = \begin{cases} p_{a_n} \cdot (I_{L_n}(k) - I_{L_n,\max}) & \text{if } I_{L_n}(k) \geq I_{L_n,\max} \\ p_{a_n} \cdot (I_{L_n,\min} - I_{L_n}(k)) & \text{if } I_{L_n}(k) \leq I_{L_n,\min} \\ p_{b_n} \cdot |I_{L_n}(k) - I_{L_n}^*| & \text{else} \end{cases} \quad (3)$$

where the terms $I_{L_n,\max}$ and $I_{L_n,\min}$ are specified as a percentage of the reference $I_{L_n}^*$.

Subsequently an objective function is formulated based on the error variable in (3). Thus, the following function is introduced

$$J(k) = \sum_{\ell=1}^N \left(\|e_1(k + \ell|k)\|_1 + \|e_2(k + \ell|k)\|_1 \right) \quad (4)$$

which penalizes the error's evolution, $e_n(k + \ell|k)$, over the finite horizon N using the 1-norm (see [12]–[14]).

In order to get the optimal control input, an optimization problem is formulated, which is commonly referred to as constrained finite time optimal control (CFTOC) problem. Due to this method, the control input is achieved by minimizing the objective function (4) subjected to constraints in each sampling period

$$u^*(k) = \arg \min J(k) \quad (5)$$

subject to eq. (2), and (3)

The optimal value is applied to the n -legs of the converter, and the procedure is repeated based on new measurements acquired on the following sampling instance.

As already mentioned, the outer loop sets the current reference in such a way that the output voltage is regulated to a desired value. The power balance mathematical expression

is used, i.e. $P_{in} = P_{out}$, in order to manipulate the reference inductor currents, and assuming that the power switches are ideal. Furthermore, since one of the goals of the controller is the balancing of the inductor currents in each leg of the converter, the corresponding reference currents must be equal. This means that the reference inductor currents are derived from the input reference current I_i^* , and are equal to $I_{L_1}^* = I_{L_2}^* = I_i^*/2$. This results to the following expression for the reference currents

$$I_{L_1}^* = I_{L_2}^* = \frac{V_i}{R_{L_1} + R_{L_2}} - \sqrt{\left(\frac{V_i}{R_{L_1} + R_{L_2}} \right)^2 - \frac{V_o^* \cdot I_o}{R_{L_1} + R_{L_2}}} \quad (6)$$

C. Load Uncertainty

In the vast majority of the applications the load cannot be modeled by a simple resistor. It can be a variable resistor, or in most of the cases, a nonlinear load, such as a power converter. In order to eliminate the undesired steady state error at the output voltage caused from unknown variations in the load, an observer is employed: the load current, which is used in the calculation of the reference inductor currents, is estimated by a second order Luenberger observer [20]. By this the reduction of the hardware cost is achieved, apart from the successful decoupling of the design of the controller from the type and the value of the load.

As shown in [16] the design of a second order Luenberger observer is given by

$$\hat{x}_L(k+1) = A_L \cdot \hat{x}_L(k) + B_L \cdot \bar{u}(k) + H \cdot \bar{y}_L(k) \quad (7a)$$

$$\hat{y}_L(k) = C_L \cdot \hat{x}_L(k) \quad (7b)$$

where

$$\begin{aligned} \hat{x}_L(k) &= [\hat{I}_o(k) \ \hat{V}_o(k)]^T, \ H = [h_1 \ h_2]^T, \ \bar{y}_L(k) = y_L(k) - \hat{y}_L(k) \\ &= C_L \cdot (x_L(k) - \hat{x}_L(k)) = C_L \cdot \bar{x}_L(k), \\ A_L &= \begin{bmatrix} 1 & 0 \\ -\frac{T_s}{C_o} & 1 \end{bmatrix}, \ B_L = \begin{bmatrix} 0 & 0 \\ \frac{T_s \cdot I_{L_1}}{C_o} & \frac{T_s \cdot I_{L_2}}{C_o} \end{bmatrix}, \ C_L = [0 \ 1] \end{aligned}$$

where $\hat{x}_L(k)$ is the estimated state. The observer gain which has to be tuned is given by the matrix $H = [h_1 \ h_2]^T$. Finally, the terms $\bar{x}_L(k)$ and $\bar{y}_L(k)$ represent the errors between the measured and the observed values of the state and the output of the system, respectively.

The error between the state variables and the observed ones at instant $(k+1) \cdot T_s$ is

$$\bar{x}_L(k+1) = (A_L - H \cdot C_L) \cdot \bar{x}_L(k) \quad (8)$$

The above error converges to zero if the observer is stable, at a rate determined by the eigenvalues of the matrix $[A_L - H \cdot C_L]$. If the pair (A_L, C_L) is observable, then the gain H exists which places the observer eigenvalues at these locations, given an arbitrary set of locations in the complex plane.

Considering the gain matrix H , the estimated variables

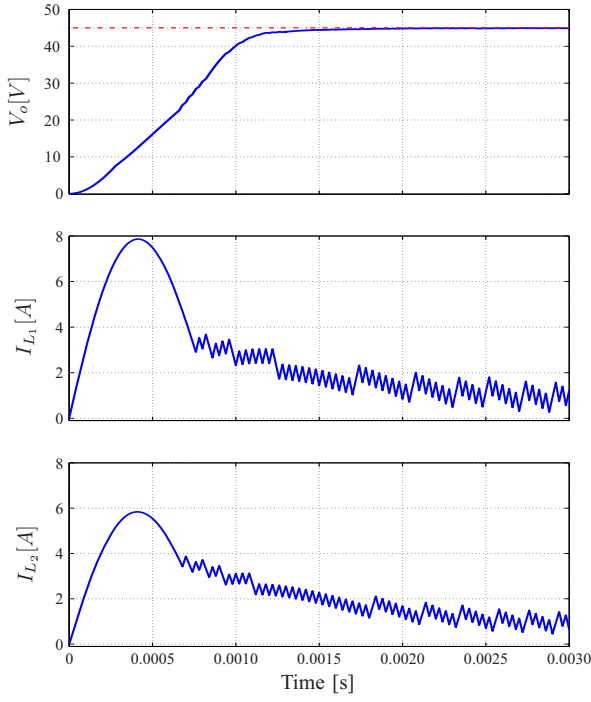


Fig. 2: Simulation results for nominal startup ($V_i = 20$ V, $V_o^* = 45$ V, $R = 75$ Ohm) from initial condition zero (from top to bottom): output voltage (blue solid line) and output voltage reference (light red dashed line), current of the first inductor, current of the second inductor.

$\hat{I}_o(k+1)$, $\hat{V}_o(k+1)$ are given by the following equations

$$\hat{I}_o(k+1) = \hat{I}_o(k) + h_1 \cdot (V_o(k) - \hat{V}_o(k)) \quad (9a)$$

$$\begin{aligned} \hat{V}_o(k+1) = & (1 - h_2) \cdot \hat{V}_o(k) - \frac{T_s}{C_o} \cdot \hat{I}_o(k) + h_2 \cdot V_o(k) + \\ & \frac{T_s}{C_o} \cdot [I_{L1}(k) \cdot (1 - u_1(k)) + I_{L2}(k) \cdot (1 - u_2(k))] \end{aligned} \quad (9b)$$

From now on, the estimated current $\hat{I}_o$ will be used instead the actual current I_o , based on equations (9b) and (9a).

D. Inductor Currents Uncertainty

In (6), the internal resistances of the inductors are considered as constants. However, in reality the values of the resistances divert from its nominal values due to the temperature increment during the power supply's operation. Consequently, there exists an error between the calculated reference values and the actual ones [21] that cannot be regarded as trifling for small loads. A proposal to compensate the deviation in the calculation of the reference currents is to augment the state-feedback controller with a different type of external estimation loop, such as a discrete-time Kalman filter [22], which would be able to model the effect of the changing inductor resistances on the inductor currents.

IV. SIMULATION RESULTS

In this section simulation results are presented in order to demonstrate the performance of the proposed controller under several operating conditions. The dynamic performance is examined for the startup. Also the response to step changes

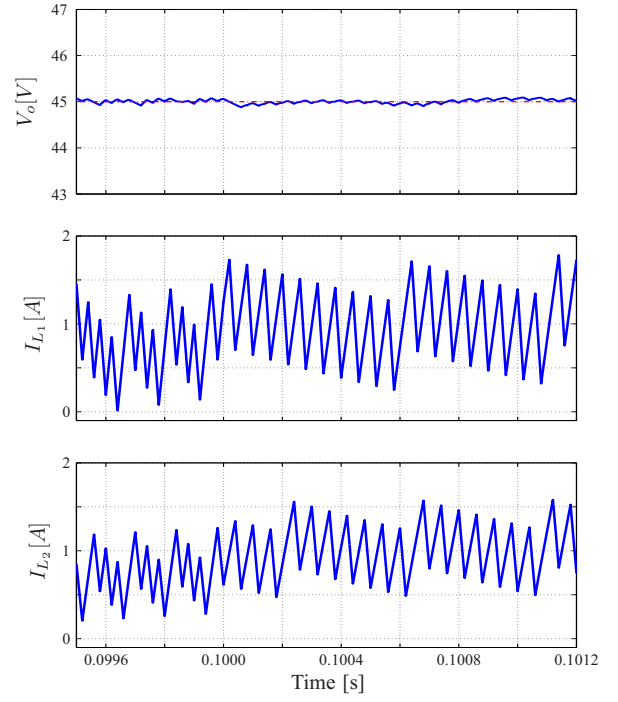


Fig. 3: Simulation results for a step-down change in the input voltage from $V_i = 20$ V to $V_i = 15$ V ($V_o^* = 45$ V, $R = 75$ Ohm) at time $t = 0.1$ s (from top to bottom): output voltage (blue solid line) and output voltage reference (light red dashed line), current of the first inductor, current of the second inductor.

in the desired output voltage level, the input voltage and the load resistance are illustrated. Furthermore, the steady state behavior is investigated in terms of current sharing in the parallel legs of the converter. The parameters of the setup are $L_1 = 0.6$ mH, $L_2 = 1.1$ mH, $R_{L1} = 0.35$ Ohm, $R_{L2} = 0.6$ Ohm, $C_o = 220$ μ F. Initially the input voltage is $V_i = 20$ V, the load resistance $R = 75$ Ohm, while the reference output voltage is set equal to $V_o^* = 45$ V. Finally, the sampling time $T_s = 20$ μ s.

Regarding the optimal control model, the weighting factors of the constraints are defined as $p_{a1} = 0.5$, $p_{a2} = 0.4$, $p_{b1} = 0.01$, and $p_{b2} = 0.01$. The bounds of the errors $I_{L1,max} = 1.10 \cdot I_{L1}^*$, $I_{L2,max} = 1.05 \cdot I_{L2}^*$, $I_{L1,min} = 0.90 \cdot I_{L1}^*$, $I_{L2,min} = 0.95 \cdot I_{L2}^*$. The prediction horizon is $N = 5$.

A. Startup

The first case to be examined is the startup from zero as initial condition. The converter operates under nominal conditions ($V_i = 20$ V, $V_o^* = 45$ V, $R = 75$ Ohm). As can be seen in Fig. 2, the inductor currents are increased until the capacitor is charged to the desired voltage level. The output voltage reaches its reference value in about 1.8 ms, while no overshoots are observed. Once the transient phenomenon has occurred, the inductor currents reach their nominal values; the output voltage remains constant to the desired level.

B. Input Voltage Step Change

During the previously attained steady-state point of operation, at time $t = 0.1$ s, a step change to the input voltage

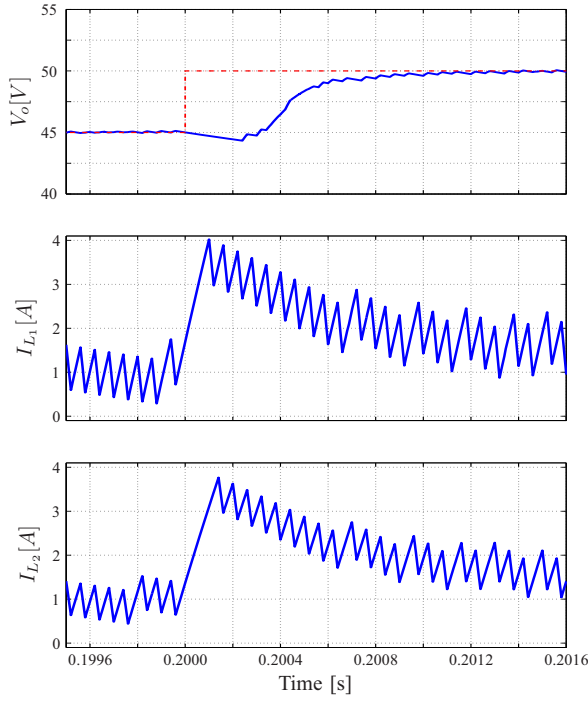


Fig. 4: Simulation results for a step-up change in the output voltage reference from $V_o^* = 45$ V to $V_o^* = 50$ V ($V_i = 15$ V, $R = 75$ Ohm) at time $t = 0.2$ s (from top to bottom): output voltage (blue solid line) and output voltage reference (light red dashed line), current of the first inductor, current of the second inductor.

is applied, from $V_i = 20$ V to $V_i = 15$ V. The response of the converter is depicted in Fig. 3. The output voltage remains practically unaffected, while the controller settles very quickly at the new steady state operating point.

C. Output Reference Voltage Step Change

The performance of the controller is further tested in the presence of a step variation in the output reference voltage. In this case a step-up change of the output reference voltage is considered, at time $t = 0.2$ s, from $V_o^* = 45$ V to $V_o^* = 50$ V. Again, the currents are increased until the capacitor is charged to the new reference voltage level. The controller exhibits satisfactory transient behavior, since it reaches the new output voltage in about 1.4 ms, with no overshoots. The segment of interest in this test is illustrated in Fig. 4.

D. Load Resistance Step Change

The last case examined is that of a step change in the load resistance occurring at $t = 0.3$ s. Starting from the nominal load $R = 75$ Ohm, a step-down change to $R = 50$ Ohm is applied. As shown in Fig. 5, the controller manages to adjust to the non-nominal operating conditions, while the system reaches the desired operating point quickly (in about 0.4 ms). The output voltage remains close to its reference value irrespective to the load step.

E. Current Balancing

Since one of the control objectives of the controller is the equal current distribution to the individual converter's legs,

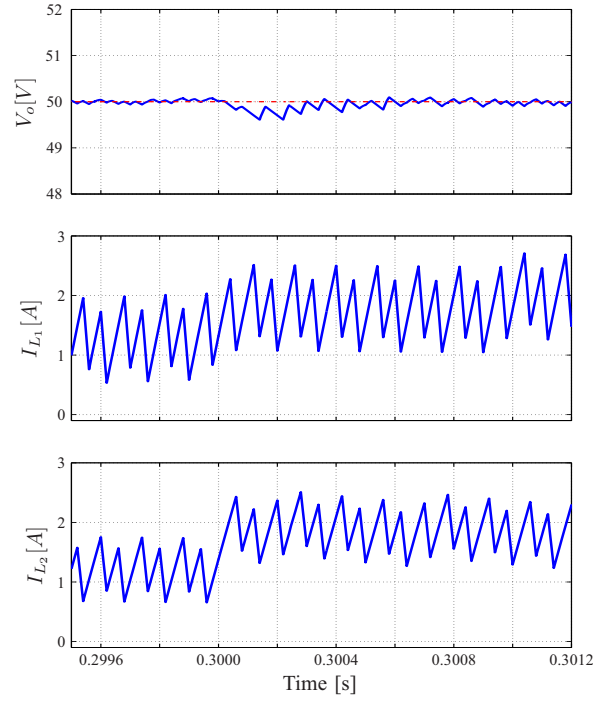


Fig. 5: Simulation results for a step-down change in the load resistance from $R = 75$ Ohm to $R = 50$ Ohm ($V_i = 15$ V, $V_o^* = 50$ V) at time $t = 0.3$ s (from top to bottom): output voltage (blue solid line) and output voltage reference (light red dashed line), current of the first inductor, current of the second inductor.

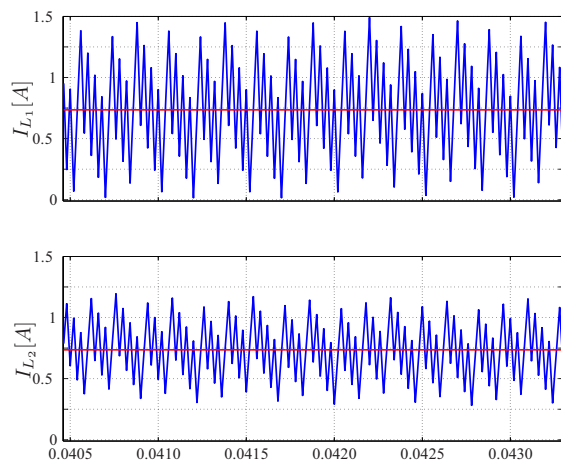
the steady-state performance is examined. In order to validate the current balancing, the average inductor currents should be calculated. However, the converter operates under variable switching frequency, which means that the integration of the current in one sampling period, T_s , is not accurate. Thus, if the currents are integrated over a large time "window", then we can claim that the outcome of the calculation is precise. The average currents are given by

$$I_{L_n,dc}(t) = \frac{\int_0^{T_{sw}} I_{L_n}(t) dt}{T_{sw}} \quad (10)$$

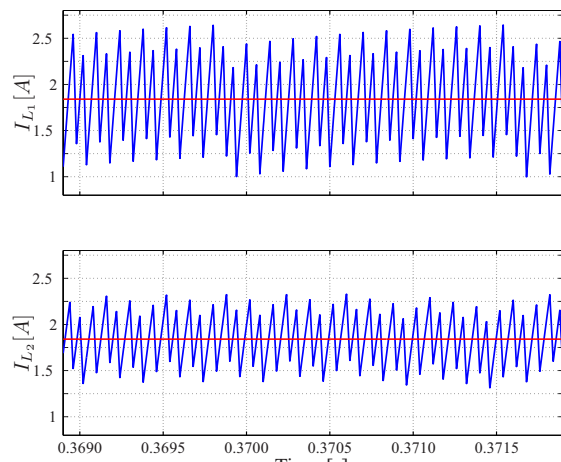
where $I_{L_n,dc}$ is the average current of the n individual inductor, I_{L_n} is the respective actual current, and T_{sw} is the time "window" with $T_{sw} \gg T_s$. Fig. 6 shows the current balancing under steady-state operation of the converter for two different operating points. Fig. 6(a) corresponds to the operating point where $V_i = 20$ V, $V_o^* = 45$ V, and $R = 75$ Ohm, while Fig. 6(b) to the operating point where $V_i = 15$ V, $V_o^* = 50$ V, and $R = 50$ Ohm.

V. CONCLUSION

In this paper, the MPC approach, which was recently introduced for the dc-dc boost converter, is extended to the dc-dc parallel interleaved boost topology that features two individual legs. The proposed control scheme leads to favorable dynamical and robustness properties, in particular during startup and transients. Furthermore, since the con-



(a) Operating point: $V_i = 20$ V, $V_o^* = 45$ V, $R = 75$ Ohm.



(b) Operating point: $V_i = 15$ V, $V_o^* = 50$ V, $R = 50$ Ohm.

Fig. 6: Simulation results for the average and actual inductor currents for two different operating points (from top to bottom for both (a) and (b)): current of the first inductor (blue line) and corresponding average current (red line), current of the second inductor (blue line) and corresponding average current (red line).

troller achieves equal current distribution to the converter legs benefits are derived in terms of cost and reliability. These advantages overshadow the inherent drawbacks of the method, such as the computational complexity and the variable switching frequency. Its straightforward implementation allows the scheme to be easily extended to the interleaved topology with more legs. Simulation results validate the high performance of the proposed method both in steady-state and in dynamic operating conditions.

REFERENCES

- [1] N. Mohan, T. M. Undeland, and W. P. Robbins, *Power Electronics: Converters, Applications and Design*, Wiley, 1989.
- [2] R. Möbus, M. Baotic, and M. Morari, "Multi-Object Adaptive Cruise Control," *Hybrid Systems: Computation and Control*, vol. 2623, pp. 359–374, Lecture Notes in Computer Science, 2003.
- [3] M. Larsson, and D. Karlsson, "Coordinated System Protection Scheme Against Voltage Collapse Using Heuristic Search and Predictive Control," *IEEE Transactions on Power Systems*, vol. 18, no. 3, pp. 1001–1006, Aug. 2003.

- [4] A. Vahidi, A. Stefanopoulou, and Huei Peng, "Current Management in a Hybrid Fuel Cell Power System: A Model-Predictive Control Approach," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 6, pp. 1047–1057, Nov. 2006.
- [5] S. Buso, and P. Mattavelli, *Digital Control in Power Electronics (Synthesis Lectures on Power Electronics)*, Morgan and Claypool publishers, 2006.
- [6] J. Chen, A. Prodic, R. W. Erickson, and D. Maksimovic, "Predictive Digital Current Programmed Control," *IEEE Transactions on Power Electronics*, vol. 18, no. 1, pp. 411–419, Jan. 2003.
- [7] Y. A. -R. I. Mohamed, and E. F. El-Saadany, "Robust High Bandwidth Discrete-Time Predictive Current Control with Predictive Internal Model—A Unified Approach for Voltage-Source PWM Converters," *IEEE Transactions on Power Electronics*, vol. 23, no. 1, pp. 126–136, Jan. 2008.
- [8] P. Mattavelli, G. Spiazzi, and P. Tenti, "Predictive Digital Control of Power Factor Preregulators With Input Voltage Estimation Using Disturbance Observers," *IEEE Transactions on Power Electronics*, vol. 20, no. 1, pp. 140–147, Jan. 2005.
- [9] G. H. Bode, P. C. Loh, M. J. Newman, and D. G. Holmes, "An Improved Robust Predictive Current Regulation Algorithm," *IEEE Transactions on Industry Applications*, vol. 41, no. 6, pp. 1720–1733, Nov. 2005.
- [10] P. Cortés, J. Rodríguez, P. Antoniewicz, and M. Kazmierkowski, "Direct Power Control of an AFE Using Predictive Control," *IEEE Transactions on Power Electronics*, vol. 23, no. 5, pp. 2516–2523, Sep. 2008.
- [11] J. Rodríguez, J. Pontt, C. A. Silva, P. Correa, P. Lezana, P. Cortés, and U. Ammann, "Predictive Current Control of a Voltage Source Inverter," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 1, pp. 495–503, Feb. 2007.
- [12] T. Geyer, G. Papafotiou, R. Frasca, and M. Morari, "Constrained Optimal Control of the Step-Down DC-DC Converter," *IEEE Transactions on Power Electronics*, vol. 23, no. 5, pp. 2454–2464, Sep. 2008.
- [13] A. G. Beccuti, G. Papafotiou, R. Frasca, and M. Morari, "Explicit Hybrid Model Predictive Control of the dc-dc Boost Converter," in *Proc. IEEE Power Electronics Specialist Conference (PESC)*, Orlando, Florida, USA, pp. 2503–2509, June 2007.
- [14] A. G. Beccuti, G. Papafotiou, and M. Morari, "Optimal Control of the Parallel Interleaved Buck dc-dc Converter," in *Proc. IEEE Workshops on Computers in Power Electronics (COMPEL)*, NY, USA, pp. 57–63, July 2006.
- [15] L. G. Balogh, "Paralleling Power - Choosing and Applying the Best Technique for Load Sharing," Unitrode Design Seminar, 2003, <http://focus.ti.com/lit/ml/slup207/slup207.pdf>.
- [16] P. Karamanakos, G. Papafotiou, and S. Manias, "Model Predictive Control Strategies for DC-DC Boost Voltage Conversion," in *Proc. European Conference on Power Electronics and Applications (EPE)*, Birmingham, UK, pp. 1–9, Aug./Sep. 2011.
- [17] J. M. Maciejowski, *Predictive Control with Constraints*, Prentice Hall publications, 2002.
- [18] R. W. Erickson, and D. Maksimovic, *Fundamentals of Power Electronics*, Kluwer Academic Publishers, 2nd edition, 2000.
- [19] T. Geyer, G. Papafotiou, and M. Morari, "A hybrid model predictive control approach to the direct torque control problem of induction motors," *International Journal of Robust and Nonlinear Control*, vol. 17, pp. 1572–1589, May 2007.
- [20] C. T. Chen, *Linear System Theory and Design*, Oxford University Press, 3rd edition, 1998.
- [21] G. T. Kostakis, S. N. Manias, and N. I. Margaris, "A Generalized Method for Calculating the RMS Values of Switching Power Converters," *IEEE Transactions on Power Electronics*, vol. 15, no. 4, pp. 616–625, July 2000.
- [22] G. Pannocchia, and J. B. Rawlings, "Disturbance models for offset-free model-predictive control," *AIChE Journal*, vol. 49, no. 2, pp. 426–437, Feb. 2003.

BRDF approximation and estimation for Augmented Reality

Patrick Kührtreiber

Martin Knecht

Christoph Traxler

Institute of Computer
Graphics and Algorithms
Vienna University of
Technology

Institute of Computer
Graphics and Algorithms
Vienna University of
Technology

Abstract—In Augmented Reality applications it is important to have a good description of the surfaces of real objects if a consistent shading between real and virtual object is required. If such a description of a surface is not available it has to be estimated or approximated.

In this paper several methods are presented that deal with the bi-directional reflectance distribution function (BRDF) approximation in Augmented Reality. Of course an important thing to discuss is whether the applications we present work in real-time and compute real looking results.

Different methods can be used to achieve these goals. All of the methods presented work via image based lighting. Some require a 3D polygonal mesh representation of the object, for which the BRDF shall be approximated. Some methods estimate the BRDF parameters via error values and provide results at each iteration.

I. INTRODUCTION

This paper provides a state of the art report about BRDF approximation and estimation in respect to Augmented Reality applications. We looked at several different approaches and emphasized whether the presented methods are applicable in AR environments.

Several papers deal with the problem of BRDF approximation. If BRDFs for certain objects are not known they have to be approximated as closely as possible. In Augmented Reality applications it would be desirable to do this during run-time.

We need a good representation of the reflection behaviours of the surfaces and, what makes it even more difficult, we need them in real-time in order to avoid a pre-modeling step.

The most of the presented methods are rather similar, as all use image based lighting. Still there are differences in performance and photorealism which we will point out at the end of each section. A complete summary and conclusion are given in sections 7 and 8.

The papers presented are:

- Image-Based Rendering of Diffuse, Specular and Glossy Surfaces from a Single Image, [1]

patrick.kuehtreiber@chello.at
martin.knecht@cg.tuwien.ac.at
traxler@cg.tuwien.ac.at

- Recovery of material under complex illumination conditions, [2]
- A Framework for Automatically Recovering Object Shape, Reflectance and Light Sources from Calibrated Images, [3]
- Recovering surface reflectance and multiple light locations and intensities from image data, [4]
- On-line estimation of diffuse materials, [5]

II. IMAGE-BASED RENDERING OF DIFFUSE, SPECULAR AND GLOSSY SURFACES FROM A SINGLE IMAGE

This approach, developed by Boivin and Gagalowicz, addresses the problem in an iterative way [1]. Every iteration step adds another level of detail to the rendered scene until the result - rendered with Ward's BRDF reflectance model [6] - looks good enough.

The input for this algorithm is a simple photograph and a geometric model of the scene which includes camera position and light sources that are not exact but approximated. For the Ward reflectance model five parameters are needed: the diffuse reflectance ρ_d , the specular reflectance ρ_s , the anisotropy direction $\vec{x}$ and the anisotropic roughness parameters α_x and α_y [1].

As stated before the algorithm works iteratively and each iteration step is refined several times. Each subsequent state refines the previous one based on an error picture between the (offscreen) rendered picture and the original photograph. The authors used a global illumination method to compute the rendered picture. The algorithm runs through a couple of assumptions where the next step is only computed if the error value from the previous picture is over a certain threshold.

A. Perfect diffuse

The error ε is computed as the ratio between the average of the radiances from an object (more exactly its a group of objects) in the real picture and in the synthetic one. The diffuse reflectance ρ_d can be corrected iteratively now with help of the error value. The steady refinement of ρ_d can be seen in Fig. 1.

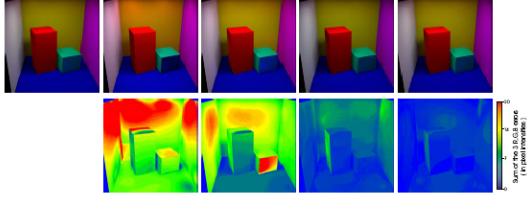


Fig. 1: In the top row are from left to right the refined images based on the error pictures below. Picture from Boivin and Galgalowicz [1].

If the perfectly diffuse (i.e. no highlights in the image) assumption fails the second assumption is tried (perfectly specular = all objects are mirrors). This is easy to accomplish as you simply have to set the diffuse reflectance $\rho_d = 0$ and the specular reflectance $\rho_s = 1$ and replace all the ρ_d with ρ_s .

If there are still objects that have a high error value, they are now considered to have diffuse and specular reflectance properties. To get good approximations to these surface reflectances the diffuse reflectance ρ_d and the specular ρ_s are computed together in a two dimensional system of linear equations.

B. Isotropic surfaces

In this step the surfaces are assumed to have a certain roughness factor α . This roughness factor is considered in Ward's BRDF Model [6]. Now ρ_d , ρ_s and α have to be minimized. Boivin and Galgalowicz did this with the downhill simplex method [7]. As the accuracy does not have to be that high (10^{-2} for ρ_d and ρ_s and 10^{-4} for α respectively), because the visual difference would become imperceptible, the parameters can be found within two minutes [1].

C. Anisotropic surfaces

In this section five parameters of Ward's BRDF model have to be taken into account: The diffuse and specular reflectances ρ_d and ρ_s , the anisotropy direction $\vec{x}$ and the roughness factors α_x and α_y . As the rather independent ρ_d and ρ_s have already been computed, they do not have to be calculated anew.

$\vec{x}$ is determined from the original picture as follows. First the authors consider the anisotropic surfaces as perfect mirrors, compute a synthetic image and estimate the difference between this image and the real one. Next they compute the index buffer for this mirror of all surfaces visible through it, then they look at a surface which has the biggest reflection area and which is closest to the mirror. Then they compute a number of vectors where each one determines a direction to traverse the error image and then they compute the average of the standard error derivation. Finally they select the direction for which the average error becomes smallest [1].

D. Textured surfaces

This is their final assumption. As the textures of the objects in the real image are already illuminated by the light sources they can not simply take those textures. So they introduce the notion *radiosity texture* that balances the extracted texture with an intermediate texture in order to minimize the error between the real and the synthetic image [1].

To increase the speed of this method they propose several alterations of the algorithm. For example, if the error after the specular assumption is higher than 50% the algorithm goes directly to the textured case as the isotropic and anisotropic cases last the longest (almost 4 hours for the anisotropic case).



Fig. 2: Result of the algorithm presented by Boivin and Galgalowicz. These images show the usability of their approach for Augmented Reality applications as there have been made some changes from the original image. The image on the upper left side shows the original image with some removed objects. The other ones show the same scene under novel lighting conditions, new viewpoints or with new objects. Picture from Boivin and Galgalowicz [1].

E. Discussion

The discussions at the end of each section is divided into discussion about running time, input data and illumination model.

1) *Running time:* The recovery of the image in the anisotropic case took more than four hours. With the enhancements they made the algorithm does not have to try the isotropic and anisotropic cases as they do not promise to deliver better results. So the two most time consuming parts of their algorithm can be skipped if the error after the specular assumption is too high.

The rendering of Fig. 2 took about half an hour. The pre-processing was of course more time consuming, the inverse algorithm took them 4 hours and 40 minutes, where 4 hours alone have been spent to recover the anisotropy parameters for the aluminium surface [1].

2) *Input data:* The input data for this algorithm was a simple photograph of the scene, where the position of the light sources have to be known and a geometric model of the

scene. If both are available then the approach from Boivin and Gagalowicz provides good looking results and is suitable for AR applications.

3) *Illumination model*: The rendering was done via a global illumination algorithm, because they had to compare the whole scene with the photograph. The BRDF parameters were retrieved for Ward's BRDF model because they took more advanced parameters like roughness into account.

III. RECOVERY OF MATERIAL UNDER COMPLEX ILLUMINATION CONDITIONS

Wu et al. presented a method on how to recover the BRDF for RADIANCE's low parameter reflectance model [8], which also uses Ward's model, for a real homogenous object under complex lighting conditions from a high dynamic range (HDR) photograph of the object and one of the environment to find the light sources [2].

Again their aim is to recover the BRDF ρ , where all other variables are known. The parameters they need for their modified RADIANCE reflectance model are specular, diffuse and directional diffuse reflectance and transmission.

The BRDF model they used can be expressed as

$$f_r = \max(0, \vec{q} \cdot \vec{n}_p) \left[\frac{\rho_d}{\pi} + \rho_s \right] + \max(0, -\vec{q} \cdot \vec{n}_p) \left[\frac{\tau_d}{\pi} + \tau_s \right] \quad (1)$$

where

$$\begin{aligned} \rho_d &= pC(1 - a_4) \\ \rho_s &= r_s \frac{f_s(\vec{q})}{\sqrt{(\vec{q} \cdot \vec{n}_p)(-\vec{v} \cdot \vec{n}_p)}} \\ \tau_d &= a_6(1 - r_s)(1 - a_7)pC \\ \tau_s &= a_6a_7(1 - r_s) \frac{g_s(\vec{q})}{\sqrt{(-\vec{q} \cdot \vec{n}_p)(-\vec{v} \cdot \vec{n}_p)}} \\ r_s &= \begin{cases} a_4 & \text{plastic} \\ \{a_1a_4, a_2a_4, a_3a_4\} & \text{metal} \end{cases} \end{aligned}$$

where $\vec{q}$ is the direction from the surface point to a light source sample, $\vec{v}$ is the viewing vector, $\vec{n}_p$ is the surface normal at the point p , C is the surface colour, p is the material pattern and $a_i (i = 1, \dots, 7)$ are parameters. The ρ -parameters define the reflection, the τ -parameters the transmission coefficients [2].

After the acquisition of the illumination maps they now proceed to recover the wanted materials for the object which is illuminated by known lighting that is represented as an illumination field. The recovery of the material parameters is done, similarly to the method in the section before, via the minimization of a difference value between the real (I_r) and a synthetic rendered (I_o) image of the object. The difference in the mean of least squares is

$$\chi^2 = \|I_o - I_r\|^2 \quad (2)$$

This is a nonlinear optimization problem that Wu et al. solved with a simulated annealing algorithm. The algorithm works with a set of initial parameter values to optimize them and to reduce χ^2 step by step until a global minimum is found. So we see that they also used a global illumination algorithm to detect the differences between the original photograph and the rendered scene. These calculated parameters are then used to render the object with ray tracing. A result of their work can be seen in Fig. 3.

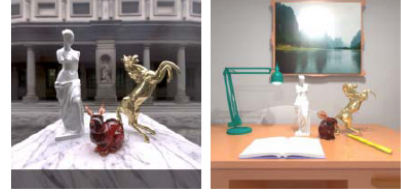


Fig. 3: Left: Virtual objects rendered into a real (Augmented Reality) scene. Right: Virtual objects rendered into a virtual scene. Picture by Wu et al. [2].

A. Discussion

1) *Running time*: The recovery of the materials took them about 2 to 3 hours on a Dell Dimension 4100 with a 667 MHz CPU and with 128 MB of working memory [2].

2) *Input data*: The method is similar to the method by Boivin and Gagalowicz [1] but they do not need a geometrical model of the scene which makes the pre-computations more efficient. On the other hand they needed several low dynamic range (LDR) pictures to derive a high dynamic range image and high dynamic range environment maps. Whereas the light sources - other than in the approach described before - do not have to be known, the estimation of the positions of the light sources takes some time.

3) *Illumination model*: Again the whole scene had to be rendered with a global illumination algorithm in order to be able to compare the images. The object's surface was reconstructed using a model which is similar to Ward's reflectance model.

IV. A FRAMEWORK FOR AUTOMATICALLY RECOVERING OBJECT SHAPE, REFLECTANCE AND LIGHT SOURCES FROM CALIBRATED IMAGES

Mercier et al. [3] present a method for recovering object shapes, reflectance properties (for the modified Phong model [9]) and the position of light sources from a set of images. We will only focus on the surface and reflectance recovery. The modified Phong model is expressed as

$$L_r = \frac{L_s k_d}{\pi r^2} \cos \theta + \frac{(n+2)L_s k_s}{2\pi r^2} \cos \theta \cos^n \phi \quad (3)$$

where L_r is the radiance reflected, L_s is the radiance emitted by a light source S arriving at P , r is the distance between

S and P , θ is the angle between the surface normal and the direction of the light source, ϕ is the angle between the mirror reflection direction and the actual reflection direction and again k_s and k_d are the specular and diffuse parameters and n is the specular exponent.

For each of the images in the image-set the position and orientation of the camera have to be known.

They made different images for different purposes. They made an overexposed image for segmentation of the surfaces and a second image for reflectance and for the estimation of the light sources.

Their first step is to acquire the object shape from these images using a shape from silhouette approach. They used a marching cubes algorithm that considers image pixels to extract the polygonal surface and the surface normals that are later needed to estimate BRDF parameters.

After they have acquired the polygonal mesh of the object they now proceed to the estimation of the light source directions [3].

During the estimation of the light sources they also present an identification algorithm to provide the BRDF coefficients again using an error function E_a .

They split the estimation of light sources into two classes. The first one applies to diffuse - the second to glossy surfaces (i.e. surfaces that reflect highlights). To find the appropriate class of surface (diffuse or glossy) a variation coefficient V^{class} is computed from the radiance samples. Now for a perfectly diffuse surface the variation coefficient equals zero. V^{class} increases directly proportional to the specular aspect of the surface i.e., it is higher the glossier a surface is. Mercier et al. applied an identification algorithm with the help of a gradient descent method in order to minimize the error E_a and hereby find the BRDF parameters. The parameters are chosen so that E_a becomes as low as possible and the type of surface is known.

A. Discussion

1) *Running time:* They used a Dual Intel Xeon 2.4 GHz processor with 1 GB of working memory. The BRDF estimation of a small object took 6 minutes and 30 seconds pre-computation time. The approach by Mercier et al. has certain limitations. For example it is hard to acquire the surface properties if the object possesses a variety of textures [3].

2) *Input data:* As input they needed an overexposed image for the estimation of the position of the light sources and several images from the scene. In all of these images the camera orientation and position have to be known. From these images they estimate the mesh of the object.

3) *Illumination model:* Mercier et al. do not apply a global illumination algorithm, therefore their method works faster but only recovers a BRDF of a single object. The BRDF parameters are retrieved for the modified Phong model.

V. RECOVERING SURFACE REFLECTANCE AND MULTIPLE LIGHT LOCATIONS AND INTENSITIES FROM IMAGE DATA

Xu and Wallace presented a method to recover reflectance properties from multiple objects using two intensity images and one depth image. Their approach provides the diffuse and constant specular reflectance parameters from object images [4]. They also recover the light source parameters.

To get the surface geometry they used an active 3D scanner and a stereo pair of CCD cameras. They split their approach into two steps. The first step is to get the light source parameters and the specular reflection for the Phong-Blinn reflection model with the simplified formula for the specular irradiance which also considers the light intensity.

$$I_{spec} = k_s L(P) (H \cdot N)^n \quad (4)$$

where $L(P)$ is the light intensity at point P , N is the normal at that point, n is the specular exponent and H is the halfway vector which calculates as

$$H = \frac{l + V}{|l + V|},$$

where l is the normalized vector pointing to the lightsource and V is the viewing vector [4], [10].

A. Obtaining the specular reflectance

As said before they used two cameras. They assume that both cameras have the same radiometric constant. The first step is to calculate the difference between both camera images. This way the part in the formula of the image brightness for the diffuse reflectance disappears as the diffuse reflectance is the same for each viewpoint and only the specular reflectance differs. Now Xu and Wallace compute the difference ϵ_r between the measured and the predicted values.

To solve this minimization function $\mathbf{f}$ Xu and Wallace used a gradient descent least-square optimization procedure on $\mathbf{f}$ using the Levenberg-Marquardt method. To compute the initial values for the specular coefficient k_{s_j} (the j parameter is for multiple light sources) they solved a linear system which is obtained by approximating the measured image brightness difference ΔI_m [4].

B. Obtaining the diffuse reflectance

Following the method to gain the specular reflectance Xu and Wallace now use the calculated difference between the two camera pictures to estimate the diffuse parameter for each

point. They again calculated the difference between the two camera images and erased the points which are visible to the relevant camera or light.

A result of Xu and Wallace's work can be seen in Fig. 4. The difference between the estimated specular parameters and the ground truth was (at 1% additive noise) 4.76% for the specular coefficient and 1.1% for the specular exponent. This image had only one point light source. The error goes up as the number of light sources increases.

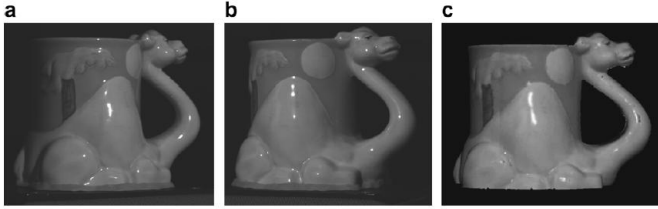


Fig. 4: a) and b) are the two stereo images of the camel cup, c) is the rendered image with the estimated parameters from a new viewpoint [4].

C. Discussion

1) *Input data:* As input they took several images to estimate the positions of the light sources and hereby detecting the specular and diffuse attributes. They do not need a polygonal mesh of the object as they used a 3D scanner and a pair of CCD cameras to get the surface geometry.

2) *Illumination model:* Like the method presented by Mercier et al., discussed in section 4, they do not use a global illumination algorithm as their algorithm just detects BRDF parameters for one single object. As illumination model they used the Phong-Blinn reflection model.

VI. ONLINE ESTIMATION OF DIFFUSE MATERIALS

Ritschel and Grosch presented a way to get diffuse parts of BRDFs at run-time from digital photographs [5]. They do not really work in real-time environments but expect their approach to also work in real-time. The only difference really is the parametrization of the model (which has to be done automatically at real-time). They used two HDR video cameras to get the diffuse materials. The equation for the outgoing radiance on a photograph at a surface point is

$$L_0 = \int_{2\pi} f_r(\omega_i, \omega_0) L_i(\omega_i) \cos(\theta_i) d\omega_i \quad (5)$$

where f_r is the BRDF of the surface, L_i is the incoming radiance from direction ω_i and $\cos(\theta_i)$ is the cosine of the angle between incoming direction and surface normal.

The previously mentioned HDR cameras are positioned as follows. One is observing the object whose BRDF shall be approximated and the second is filming the light source. This *light-camera* is at a fixed position and records the whole

environment illumination with a fisheye lens. The *object-camera* should be moved as close to the object as possible so we get the same illumination on the virtual object as there is on the real object.

A marker is placed besides the object to track camera position and orientation. This happens via optical tracking with ARToolkit [11].

The images captured with this camera run through a couple of processes which are out of the scope of this paper. The software side of their procedure is divided into two steps: *Inverse Texturing* (storing the camera images to a texture) and *Inverse Lighting* (processing these textures to one final reflectance texture).

As their approach only works for diffuse lighting, they propose to factorize the software steps into orthogonal components to get the specular part as well. A result of their work can be seen in Fig. 5), where a real object gets duplicated. The BRDF of the duplicated object is approximated so that it matches the different alignment.



Fig. 5: A cloned donkey from Ritschel et al. [5]

A. Discussion

1) *Running time:* The algorithm presented by Ritschel and Grosch has a performance of 5 fps with a model that has 100 facets and a resolution of 320x240. The most time consuming process was the inverse lighting (70 ms). Of course the performance always depends on the number of pixels and texels.

2) *Input data:* As input data they used a polygonal mesh of the object whose BRDF shall be approximated but they state that an algorithm which approximates a surface model of a given object on-line would also work.

3) *Illumination model:* Ritschel and Grosch approximate a BRDF of one single object and give as output BRDF parameters for the equation mentioned earlier (5).

VII. COMPARISON OF THE PRESENTED METHODS

A. Running time

If a fast solution is needed, the approach from Boivin and Gagalowicz [1] is the best choice even though a fast approximation provides an image which is not the correct one.

If you change the threshold and the error value accordingly you get a rough approximation after a very short time (if you skip the isotropic and anisotropic assumptions it will always be fast).

The only method which estimates a BRDF of a single object on-line is the one presented by Ritschel and Grosch [5]

B. Input data

Several presented papers used image based methods to estimate the BRDF parameters. Most need an HDR, or couple of LDR pictures of the scene and a 3D model of the scene/of an object. Some papers need a polygonal mesh as input [1]. The other ones include or reference to a parametrization method which is of course also costly.

So if a 3D polygonal mesh is known the Method from Boivin and Gagalowicz [1] can be applied to approximate or estimate the BRDF. Additionally the method from Boivin and Gagalowicz [1] provides very good looking results with an algorithm that is not hard to implement.

If the 3D polygonal mesh is not known one of the more modern approaches should be used as some of them find the shape of the objects just from images [3], [2], [5]. Xu and Wallace [4] used an active 3D scanner and a pair of CCD cameras which does not appear to be a low budget solution.

C. Illumination model

Some papers ([3], [4]) used the Phong illumination model or the Phong-Blinn model respectively. While Phong's model is widely used and simple it is still not physically plausible. Other papers try to make their objects look more realistic and adapt different illumination models. Two papers approximated and estimated parameters for Ward's BRDF model [12], [1], others for RADIANCE [2] or a completely different illumination algorithm [5].

The realism of the papers was generally rather good.

D. Suitable for Augmented Reality

The best solution for Augmented Reality applications was presented by Boivin and Gagalowicz who also presented some ways to deal with their method in Augmented Reality (setting of a novel viewpoint, changing illumination conditions, adding and removing objects). Most of the other papers dealt only with the recovery of BRDF (or shape) of a certain model as they assumed that the geometry is not known. A summary of the comparison can be seen in table 1.

VIII. CONCLUSION

In our paper we presented 5 papers that deal with BRDF approximation or estimation. All mention the possibility to use their work in Augmented Reality applications which was of course the topic and an important point of our paper.

TABLE I: Comparison of the methods

Method	BRDF	Input data	Real-time
[1]	Ward	Mesh and image	No
[2]	RADIANCE	LDR images	No
[3]	Modified Phong	Images of the scene Camera, light sources	No
[4]	Phong-Blinn	Images of the scene	No
[5]	Own BRDF model	Mesh	Yes

In each section we summarized the parts of the papers that dealt with BRDF approximation or estimation with respect to their usage in Augmented Reality. We presented results of the different papers and concluded each section with a short paragraph about the performance (if given in the original work) and problems that occurred or might occur when using the corresponding algorithm. At the end we compared all the relevant papers in respect to input data, running time, the illumination method used (Phong, Phong-Blinn, Ward,...), photorealism, whether or not a rough approximation of an image before the actual computation of the end result is possible, their usage in Augmented Reality applications and whether the methods are costly or low budget solutions.

REFERENCES

- [1] S. Boivin and A. Gagalowicz. Image-based rendering of diffuse, specular and glossy surfaces from a single image. In *SIGGraph-01*, pages 107–116, 2001.
- [2] Enhua Wu, Qimin Sun, and Xuehui Liu. Recovery of material under complex illumination conditions. In Yong Tsui Lee, Stephen N. Spencer, Alan Chalmers, and Seah Hock Soon, editors, *GRAPHITE*, pages 39–45. ACM, 2004.
- [3] B. Mercier, D. Meneveaux, and A. Fournier. A framework for automatically recovering object shape, reflectance and light sources from calibrated images. *International Journal of Computer Vision*, 73(1):77–93, June 2007.
- [4] S. Xu and A. M. Wallace. Recovering surface reflectance and multiple light locations and intensities from image data. *Pattern Recognition Letters*, 29(11):1639–1647, August 2008.
- [5] Tobias Ritschel and Thorsten Grosch. On-line estimation of diffuse materials, April 01 2008.
- [6] Gregory J. Ward. Measuring and modeling anisotropic reflection. In Edwin E. Catmull, editor, *Computer Graphics (SIGGRAPH '92 Proceedings)*, volume 26, pages 265–272, July 1992.
- [7] J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.
- [8] Gregory J. Ward. The RADIANCE Lighting Simulation and Rendering System. In *Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings)*, pages 459–472, 1994.
- [9] R. R. Lewis. Making shaders more physically plausible. *Computer Graphics Forum*, 13(2):109–120, January 1994.
- [10] Donald Hearn and M. Baker. *Computer Graphics with OpenGL*. Prentice-Hall PTR, pub-PHPTR:adr, third edition, 2004.
- [11] Shared-Space-ARToolKit. www.hitl.washington.edu/projects/sharedspace.
- [12] Y. Yu, P. E. Debevec, J. Malik, and T. Hawkins. Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *SIGGRAPH-99*, pages 215–224, 1999.

An Evolutionary Optimization Framework for Intelligent Agents

Florin Leon, Bogdan I. Aignătoaiei, and Andrei D. Leca

Abstract— Evolutionary algorithms have become popular in the recent years as a general, simple, and robust technique that can be used when other optimization methods cannot be applied. Presently, there are a number of evolutionary/genetic algorithm libraries publicly available; however, they are not specifically designed for multiagent systems. The framework described in this paper addresses this problem and presents a general architecture for evolutionary optimization aimed to be used by agents in a multiagent system implemented in the JADE framework inspired from the island model of parallel genetic algorithms. The characteristics of evolutionary algorithms that are more appropriate for use in a distributed, cooperative multiagent environment are also identified.

I. INTRODUCTION

AN evolutionary algorithm is a problem solving technique inspired by the process of natural evolution. The main idea is to represent a set of potential solutions as a population of individuals, encoded by chromosomes. Chromosomes consist of genes which reflect different aspects of the solution. Evolution is not a directed process, but a heuristic one, in which the purpose of the individuals is to compete in order to propagate their genetic material to the next generation. Instead, evolutionary algorithms use a fitness function to compute how close a potential solution is to the desired solution [1].

Evolutionary algorithms have become popular in the recent years as a general, simple, and robust technique that can be used when other optimization methods cannot be applied. Among their advantages, which facilitate their application to various fields (such as planning, design, control, classification, clustering, time series modeling), one can mention the following: 1) they do not require the objective function to be continuous and/or differentiable; 2) they do not require extensive problem formulation (in case of traditional methods such as integer programming, geometric programming, branch and bound methods, special mathematical formulation is required for solving a problem); 3) they are not sensitive to the starting point of the search; 4) they do not usually get stuck into the so-called local optima and are more likely to find the global optimum of a function [2], [3].

F. Leon, B. I. Aignătoaiei, and A. D. Leca are with the Automatic Control and Computer Engineering Faculty of the Technical University “Gheorghe Asachi” of Iași, Bd. Mangeron 27, 700050 Iași, Romania (e-mails: fleon@cs.tuiasi.ro, abogdanioan@yahoo.com, andy_leca@yahoo.com).

Holland [4] introduced the concepts of exploration and exploitation as competing forces in adaptation. Exploration can be seen as trying new potential solutions that may not have high fitness, thus widening the search. Exploitation aims to use the power of high quality potential solutions to construct better ones. A good genetic algorithm should find a balance between the two, introducing new genetic material and maintaining the diversity of the population without destroying potential solutions with high fitness [5].

Recent advances in the field of evolutionary computation include [6]: algorithms for multiobjective optimization [7], [8], differential evolution, which uses a specific reproduction mechanism [9], memetic algorithms and cultural evolution, which model the evolution of the culture of a population and the influence of culture on genetic and adaptive evolution of individuals [10], and coevolution and self-generating memetic computing, in which individuals evolve through cooperation or competition or become fit to survive aided by a local search mechanism, possibly rule-based [11], [12].

Presently, there are a number of evolutionary/genetic algorithm libraries available, such as: *Evolving Objects (EO): an Evolutionary Computation Framework* [13], *Genetic Algorithms Framework* [14], *Watchmaker Framework for Evolutionary Computation* [15], *Genetic Algorithms Library* [16], as well as many others. They all aim to ease the utilization of evolutionary optimization methods in user applications; however, they are not specifically designed for multiagent systems.

The framework described in this paper addresses this problem and shows how agents in a multiagent system implemented in JADE [17] can use these techniques for a variety of purposes. We also identify the characteristics of an evolutionary algorithm that can make it more appropriate to be used in a distributed, cooperative multiagent environment.

II. DESIGNING A GENERAL ARCHITECTURE FOR EVOLUTIONARY SOLVING OF OPTIMIZATION PROBLEMS

The purpose of the evolutionary framework is to allow users, including agents from multiagent systems, to define the optimization problems and to choose the desired parameters for solving them in a uniform manner, independent of the actual problem. Figure 1 presents the UML class diagram of the general evolutionary

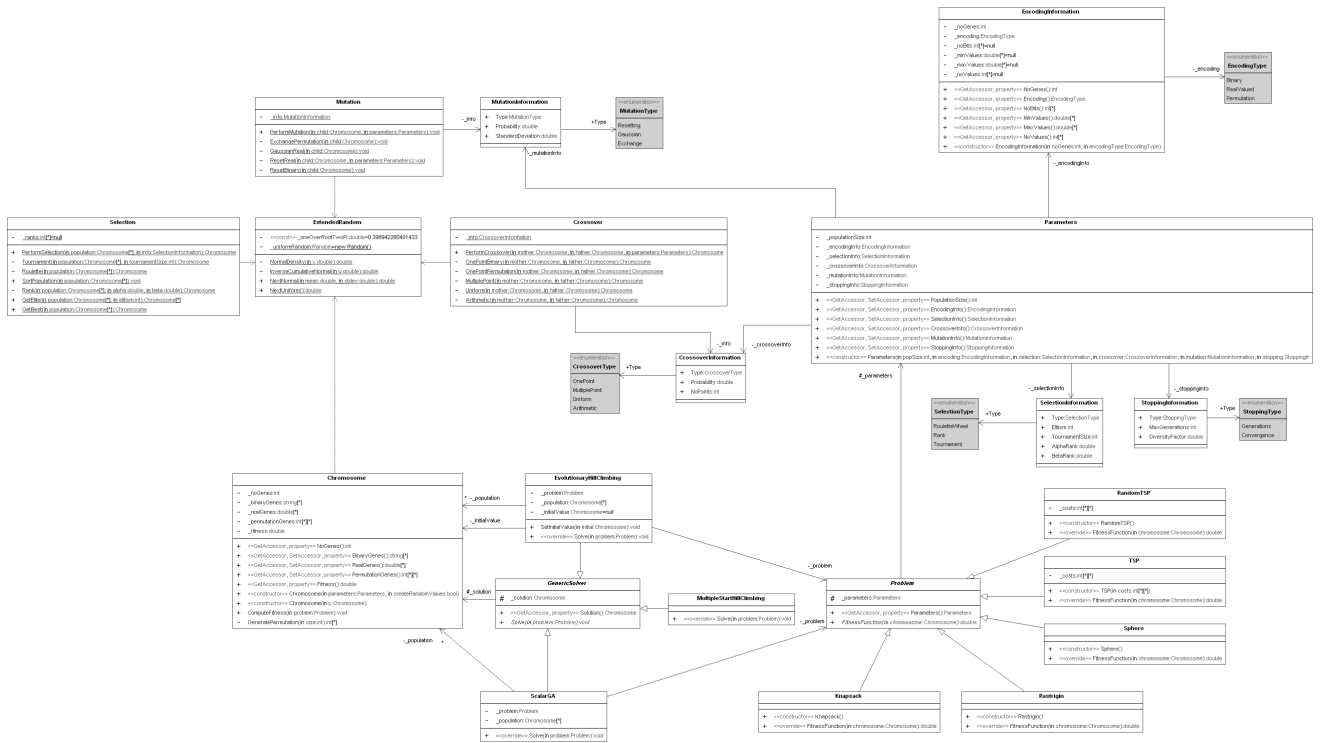


Fig. 1. General architecture for evolutionary algorithms, including class attributes and operations

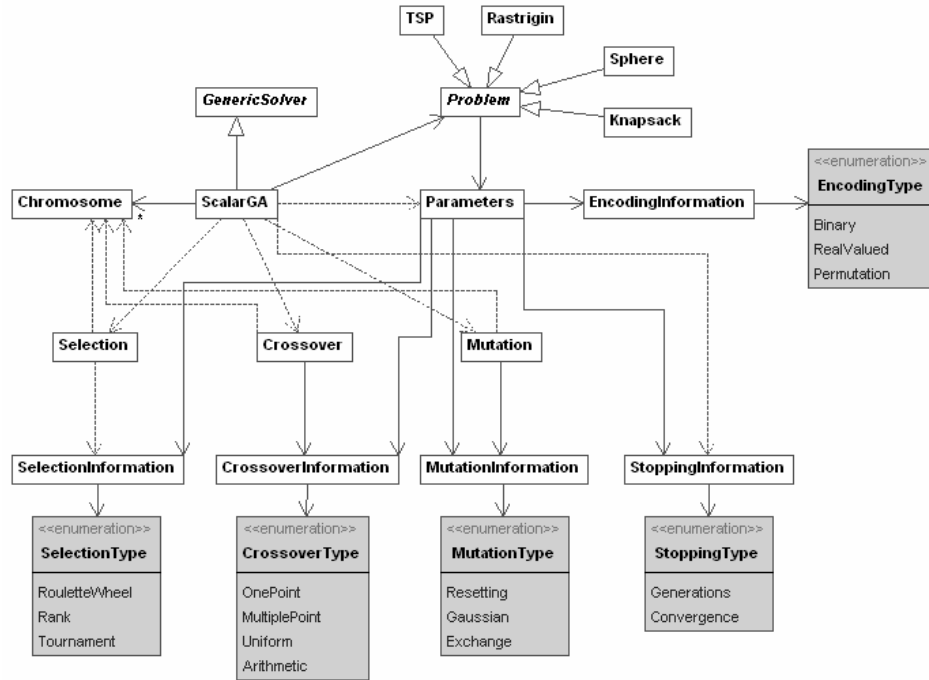


Fig. 2. General architecture for evolutionary algorithms, containing only the names of important types

optimization system.

Figure 2 shows the same architecture, while retaining only the names of the important types, in order to increase clarity.

The *constructor* instantiates the *parameters* field, which controls the encoding type (binary, real-valued or permutation-based), the selection (using roulette wheel, ranks or tournament), crossover (one point, multiple point,

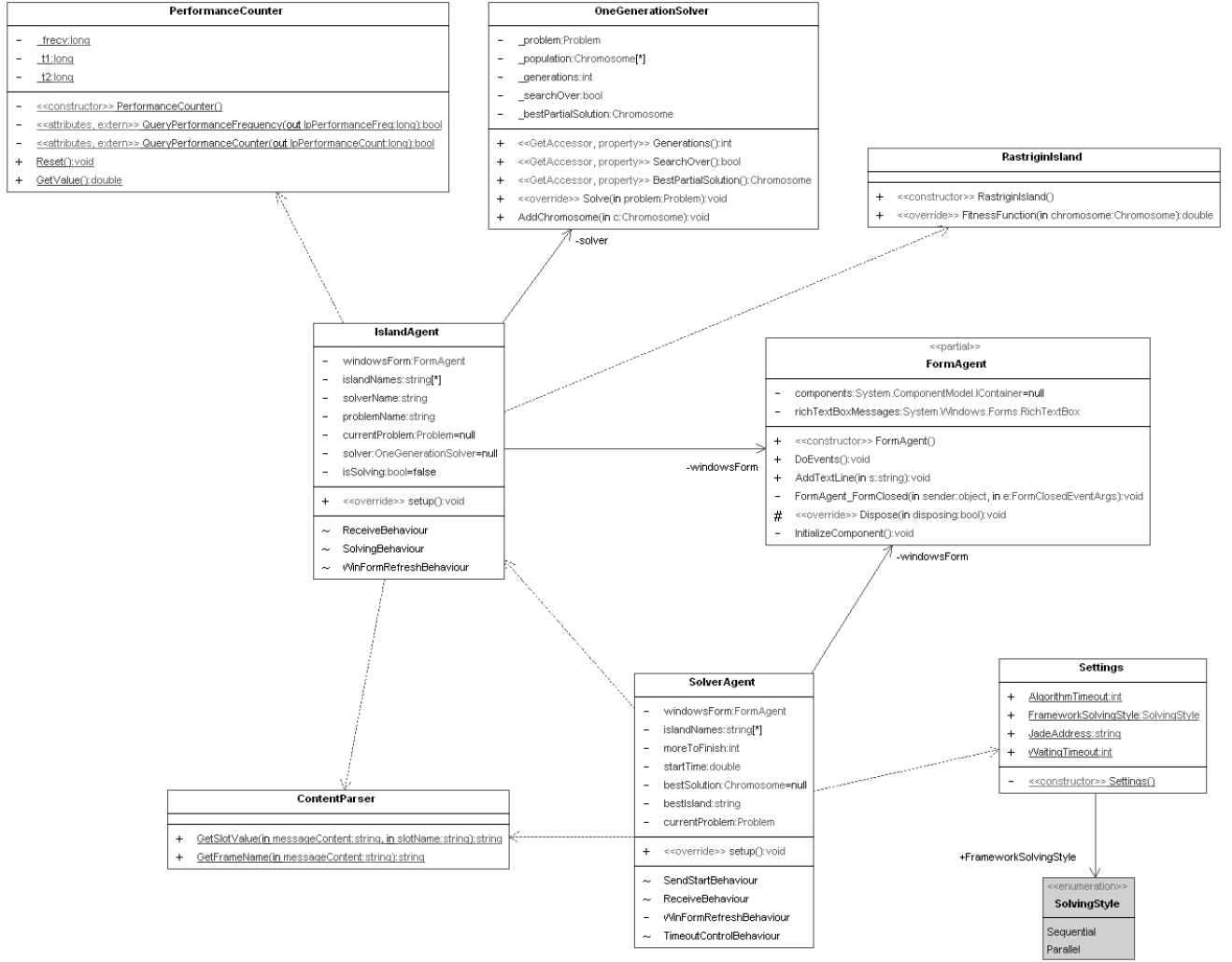


Fig. 3. The architecture of the island agents application

uniform or arithmetic) and mutation (resetting a gene to a value in its definition domain, Gaussian or exchanging two genes for permutation encoding), and the stopping criterion (a maximum number of generations or convergence criteria).

The *FitnessFunction* method effectively defines the problem, computing the fitness function of a chromosome. The solving algorithm, *ScalarGA*, will automatically use these values. The user just needs to instantiate the objects and to call the solving method, for example:

```

Problem problem = new Knapsack();
ScalarGA solver = new ScalarGA();
solver.Solve(problem)

```

After the problem has been solved, the solution is available in *solver.Solution*.

The architecture is flexible so that new optimization algorithms can be easily added by deriving the *GenericSolver* class.

III. DESIGNING A MULTIAGENT PROTOCOL FOR THE INTEGRATION OF OPTIMIZATION CAPABILITIES BASED ON EVOLUTIONARY ALGORITHMS

The island model for a parallel genetic algorithm [18] assumes the presence of several quasi-independent subpopulations that evolve using a serial genetic algorithm for a certain number of generations (isolation time). In order to avoid the genetic drift, there is a migration of individuals, i.e. after the isolation time a number of individuals are distributed between the subpopulations. The number of exchanged individuals (the migration rate), the selection method of the individuals for migration, and the scheme of migration determine how much genetic diversity can occur in the subpopulations and the amount of information exchange between subpopulations.

This model is suitable for an agent-based implementation, because the parallel execution, on separate threads, is a fundamental feature provided by the JADE

framework. Each island runs its own evolutionary algorithm, with a smaller population than it would be the case in the classic approach. After a number of generations, in our case 10, each island agent sends the best solution to the other islands. This migration of the best individuals is performed completely asynchronously; thus each island will directly integrate the received solutions in its current population, with the result of a variable number of individuals in the population. The number is expected to stabilize in the next generation, after the typical stages of selection, crossover, and mutation. After meeting the stopping criterion, the island agents send the solutions to a central *Solver* agent, which then sends its best solution to the *Requester* agent. The architecture of the agent-based island model application is displayed in figure 3.

In terms of sent messages, the solving request is expressed in an ACL message with the content of the following form: (*ea.problem :id a1 :name rastrigin :mode m*), where *m* can have the following values: *centralized* or *distributed n*, where *n* is the number of islands, implemented as JADE secondary containers.

IV. THE INFLUENCE OF THE EVOLUTIONARY ALGORITHM PARAMETERS ON AGENT PERFORMANCE

For the performance tests, the generalized Rastrigin function was considered. This is a rather difficult problem because the function is a non-linear multimodal function, with a large search space and a great number of local minima. It was first proposed by Rastrigin [19] as a 2-dimensional function and generalized by Mühlenbein et al. [20]:

$$f(\mathbf{x}) = nA + \sum_{i=1}^n x_i^2 - A \cos(\omega x_i) \quad (1)$$

where: $A = 10$, $\omega = 2\pi$, $x_i \in [-5.12, 5.12]$. In this case, we chose $n = 10$. The function minimum is $f(\mathbf{0}) = 0$.

Table 1 presents the results of the tests that were centrally performed (the equivalent of a single island). The selection method was the tournament selection.

For all tests, the last column presents the average fitness values for 3 different runs.

TABLE I
THE ALGORITHM PERFORMANCE FOR THE CENTRALIZED APPROACH AND TOURNAMENT SELECTION

Pop. size	Crossover prob.	Mutation prob.	Tour. size	No. gen.	Average fitness
80	0.9	0.02	2	500	0.243852
120	0.9	0.02	2	500	0.16298
160	0.9	0.02	2	500	0.820031
80	0.9	0.02	2	100	3.900015
120	0.9	0.02	2	100	3.035928
160	0.9	0.02	2	100	1.91191
80	0.9	0.02	2	1000	0.058187

Pop. size	Crossover prob.	Mutation prob.	Tour. size	No. gen.	Average fitness
120	0.9	0.02	2	1000	0.002255
160	0.9	0.02	2	1000	0.003299
80	0.9	0.02	3	500	0.319318
120	0.9	0.02	3	500	0.006044
160	0.9	0.02	3	500	0.00592
80	0.9	0.02	3	100	5.189024
120	0.9	0.02	3	100	4.036196
160	0.9	0.02	3	100	1.606778
80	0.9	0.02	3	1000	0.000687
120	0.9	0.02	3	1000	0.000263
160	0.9	0.02	3	1000	0.000709
80	0.9	0.02	4	500	0.021286
120	0.9	0.02	4	500	0.004299
160	0.9	0.02	4	500	0.006499
80	0.9	0.02	4	100	4.576429
120	0.9	0.02	4	100	0.338289
160	0.9	0.02	4	100	3.721454
80	0.9	0.02	4	1000	0.000417
120	0.9	0.02	4	1000	0.000227
160	0.9	0.02	4	1000	0.00048
80	0.7	0.02	2	500	0.253304
120	0.8	0.02	2	500	0.068505
160	1	0.02	2	500	0.050298
80	0.9	0.01	2	500	2.308832
120	0.9	0.05	2	500	0.006301
160	0.9	0.1	2	500	0.107507

Table 2 presents the results of tests that were performed in a distributed manner, with 4 islands. The selection method was the same, tournament selection.

TABLE II
THE ALGORITHM PERFORMANCE FOR THE DISTRIBUTED APPROACH WITH 4 ISLANDS AND TOURNAMENT SELECTION

Pop. size	Crossover prob.	Mutation prob.	Tour. size	No. gen.	Average fitness
20	0.9	0.02	2	500	1.755419
30	0.9	0.02	2	500	0.801004
40	0.9	0.02	2	500	0.361593
20	0.9	0.02	2	100	16.9877
30	0.9	0.02	2	100	8.839644
40	0.9	0.02	2	100	7.14247
20	0.9	0.02	2	1000	0.292459
30	0.9	0.02	2	1000	0.156756
40	0.9	0.02	2	1000	0.019396
20	0.9	0.02	3	500	1.308975
30	0.9	0.02	3	500	0.471907
40	0.9	0.02	3	500	0.105335
20	0.9	0.02	3	100	14.01498
30	0.9	0.02	3	100	13.13762
40	0.9	0.02	3	100	8.367195
20	0.9	0.02	3	1000	0.02541
30	0.9	0.02	3	1000	0.040452
40	0.9	0.02	3	1000	0.006349
20	0.9	0.02	4	500	1.677783
30	0.9	0.02	4	500	1.093129
40	0.9	0.02	4	500	0.387907
20	0.9	0.02	4	100	20.2736
30	0.9	0.02	4	100	12.7956
40	0.9	0.02	4	100	11.83592
20	0.9	0.02	4	1000	0.045207
30	0.9	0.02	4	1000	0.013763
40	0.9	0.02	4	1000	0.030195
20	0.7	0.02	2	500	2.0787

<i>Pop. size</i>	<i>Crossover prob.</i>	<i>Mutation prob.</i>	<i>Tour. size</i>	<i>No. gen.</i>	<i>Average fitness</i>
30	0.8	0.02	2	500	0.685504
40	1	0.02	2	500	0.456059
20	0.9	0.01	2	500	4.221374
30	0.9	0.05	2	500	0.440039
40	0.9	0.1	2	500	0.363517

Table 3 presents the results of tests that were performed in a distributed manner, with 4 islands, with roulette-wheel selection.

TABLE III
THE ALGORITHM PERFORMANCE FOR THE DISTRIBUTED APPROACH WITH THE 4 ISLANDS AND ROULETTE-WHEEL SELECTION

<i>Pop. size</i>	<i>Crossover prob.</i>	<i>Mutation prob.</i>	<i>No. gen.</i>	<i>Average fitness</i>
20	0.9	0.02	500	2.083028
30	0.9	0.02	500	1.794966
40	0.9	0.02	500	2.197628
20	0.9	0.02	100	9.460892
30	0.9	0.02	100	9.414381
40	0.9	0.02	100	8.506805
20	0.9	0.02	1000	0.069107
30	0.9	0.02	1000	0.203195
40	0.9	0.02	1000	1.062255
20	0.7	0.02	500	0.482822
30	0.8	0.02	500	1.325427
40	1	0.02	500	2.241662
20	0.9	0.01	500	2.556254
30	0.9	0.05	500	7.425697
40	0.9	0.1	500	18.74233

Table 4 presents the results of tests that were performed in a distributed manner, with 4 islands, with rank-based selection.

TABLE IV
THE ALGORITHM PERFORMANCE FOR THE DISTRIBUTED APPROACH WITH THE 4 ISLANDS AND RANK-BASED SELECTION

<i>Pop. size</i>	<i>Crossover prob.</i>	<i>Mutation prob.</i>	<i>No. gen.</i>	<i>Average fitness</i>
20	0.9	0.02	500	0.766427
30	0.9	0.02	500	0.013714
40	0.9	0.02	500	0.443239
20	0.9	0.02	100	10.62477
30	0.9	0.02	100	6.591824
40	0.9	0.02	100	5.74972
20	0.9	0.02	1000	0.000448
30	0.9	0.02	1000	0.000772
40	0.9	0.02	1000	0.000904
20	0.7	0.02	500	0.389848
30	0.8	0.02	500	0.02078
40	1	0.02	500	0.007992
20	0.9	0.01	500	1.366324
30	0.9	0.05	500	4.949654
40	0.9	0.1	500	17.20109

We have considered several scenarios of execution, i.e. centralized or distributed with 4 islands, with a variable number of generations (100, 500, 1000), with different population sizes (20, 30, 40 for the islands, and 80, 120,

160 for the centralized case) and with different selection methods (tournament with the size of 2, 3 or 4, the rank-based and the roulette-wheel method). It has been found that the centralized case has better results when the population size and the number of generations are smaller. When the population size increases to 40, the islands begin to have enough diversity to provide better solutions. The main advantage of the distributed case is the lower computing time. The increase of the tournament size improves the results, because the selective pressure increases, except for the case when the population size and the number of generations are small. The search efficiency of the algorithm and the genetic diversity are smaller in this case and the tournament selection with 2 individuals is superior. Comparing the selection methods, the best results are given by the rank-based method, followed by the tournament selection and then by the roulette-wheel method. The disadvantage of the rank-based selection is the step of sorting the population by the value of the fitness function after every generation.

V. CONCLUSION

A number of characteristics of evolutionary algorithms which are particularly suited for their use in the context of multiagent systems were identified as follows: 1) tournament selection; 2) if the number of agents is equal to the number of different machines, the island model can be applied; 3) elitism is used; 4) islands exchange the first chromosome(s) once every few generations; 5) due to parallelism and asynchronous transmission of optimal solutions, the new chromosomes enter directly into the population (which will have a variable size), and the number of individuals in the population will adjust in the next generation; 6) after the maximum number of generations, the island agents send their best chromosome to the central agent; 7) optionally, the central agent can build a new, smaller population based on the best solutions of the islands, and run the optimization for another number of generations, to combine the best features of the islands.

The optimization framework based on evolutionary algorithms can be adapted to the features of many multiagent systems. Thus, if the intelligent agents rely on neural networks for their cognitive model or for their learning capabilities, the optimization can refer to finding the best network inputs in order to maximize utility. Another typical problem is path optimization for mobile agents, as this problem is equivalent to the Traveling Salesman Problem (TSP). Finally, evolutionary algorithms can also be used for the optimization of agent plans.

The solution presented in this paper has several characteristics that distinguish it from other similar software products: it uses JADE multiagent framework in C# language, not Java, it explicitly implements the island

model of evolutionary algorithms using agents, and it easily allows the user to define virtually any optimization problem and to extend the present framework with other optimization algorithms.

ACKNOWLEDGMENT

This work was supported by CNCSIS-UEFISCSU, project number PNII-IDEI 316/2008, *Behavioural Patterns Library for Intelligent Agents Used in Engineering and Management*.

REFERENCES

- [1] S. Curteanu, F. Leon, D. Gălea, "Alternatives for Multiobjective Optimization of a Polymerization Process", *Journal of Applied Polymer Science*, vol. 100, issue 5, pp. 3680-3695, Wiley Periodicals, USA, 2006.
- [2] B. V. Babu, R. Angira, "Modified differential evolution (MDE) for optimization of non-linear chemical processes", *Comp Chem Eng*, vol. 30, pp. 989-1002, 2006.
- [3] S. Curteanu, F. Leon, "Optimization Strategy Based on Genetic Algorithms and Neural Networks Applied to a Polymerization Process", *International Journal of Quantum Chemistry*, vol. 108, pp. 617-630, Wiley Periodicals, USA, 2008.
- [4] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, 1975.
- [5] M. H. Zaharia, F. Leon, D. Gălea, "Parallel Genetic Algorithms for Cluster Load Balancing", Third European Conference on Intelligent Systems & Technologies ECIT 2004, Performantica Press, Iași, 2004.
- [6] R. V. Kulkarni, A. Förster, G. K. Venayagamoorthy, "Computational Intelligence in Wireless Sensor Networks: A Survey", *IEEE Communications Surveys & Tutorials*, vol. 13, issue 1, 2011.
- [7] K. Deb, A. Pratab, S. Agrawal, T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II", *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, 2002.
- [8] S. Chaudhuri, K. Deb, "An interactive evolutionary multi-objective optimization and decision making procedure", *Applied Soft Computing*, vol. 10, no. 2, pp. 496-511, 2010.
- [9] R. Storn, K. Price, "Differential Evolution - a fast and efficient heuristic for global optimization over continuous spaces", *Journal of Global Optimization*, no. 11, pp. 341-359, 2007.
- [10] Y. S. Ong, M. H. Lim, N. Zhu, K. W. Wong, "Classification of Adaptive Memetic Algorithms: A Comparative Study", *IEEE Transactions on Systems Man and Cybernetics, Part B*, vol. 36, no. 1, pp. 141-152, 2006.
- [11] N. Krasnogor, S. Gustafson, "Toward truly memetic memetic algorithms: discussion and proof of concepts", *Advances in Nature-Inspired Computation: the PPSN VII Workshops, Parallel Emergent and Distributed Architectures Lab*, University of Reading, 2002.
- [12] J. E. Smith, "Coevolving Memetic Algorithms: A Review and Progress Report", *IEEE Transactions on Systems Man and Cybernetics, Part B*, vol. 37, no. 1, pp. 6-17, 2007.
- [13] Evolving Objects (EO): an Evolutionary Computation Framework (2011) Available: <http://eodev.sourceforge.net>.
- [14] Genetic Algorithms Framework (2011) Available: <http://sourceforge.net/projects/ga-fwork>.
- [15] Watchmaker Framework for Evolutionary Computation (2011) Available: <http://watchmaker.uncommons.org>.
- [16] Genetic Algorithms Library (2011) Available: http://www.aforgenet.com/framework/features/genetic_algorithms.html.
- [17] F. L. Bellifemine, G. Caire, D. Greenwood, *Developing multi-agent systems with JADE*, Wiley Series in Agent Technology, Wiley, 2007.
- [18] J. Cohoon, S. Hegde, W. Martin, D. Richards, "Punctuated equilibria: a parallel genetic algorithm", in Grefenstette, J., ed., *Proceedings of the Second International Conference on Genetic Algorithms*, pp. 148-154, Hillsdale, NJ, USA, 1987.
- [19] A. Törn, A. Zilinskas, "Global Optimization", *Lecture Notes in Computer Science*, no. 350, Springer-Verlag, Berlin, 1989.
- [20] H. Mühlenbein, D. Schomisch, J. Born, "The Parallel Genetic Algorithm as Function Optimizer", *Parallel Computing*, no. 17, pp. 619-632, 1991.

An overview on Interrelated Negotiations

Lucian Luncean,
Romanian-German University of Sibiu,
Faculty of Economic Sciences,
lucian.luncean@roger-univ.ro

Abstract - This overview presents the current status in the field of interrelated negotiations applied to supply chain management domain as a test case. The paper begins with a short introduction and some background information in negotiation and supply chain management. Next is presented a review of current status and progress done in domain. There are presented some potential ideas for work to be performed in interrelated negotiations. The paper ends with the conclusions and the next steps to be taken in approaching the problems, which will eventually lead to further researches.

I. INTRODUCTION

“In business, you don't get what you deserve, you get what you negotiate” is the title of a famous book [5] by Chester L. Karrass highlighting the role of negotiations in life (be it private or business). Humans have used them from early times even if they were not named “negotiations”. The bargaining processes appeared and developed simultaneously with good exchanges. Even today, in the modern world, bargaining is a central custom (a way of life) for commerce in several cultures.

Different negotiation methods have been developed in time, each with its own characteristics, rules or applicability domains. Various researches have been conducted to use them in automated or semi-automated negotiations, after the introduction of digital computers.

While negotiations are encountered in most aspects of life there are some domains where the automation can greatly increase the benefits of using it. One such domain is the management of supply chains (or, as it is more often known, supply chain management – SCM). The economic development has increased the specialization and dependencies between economic actors. Most of the current products are complex and are made from parts obtained from different suppliers. The resulting goods arrive to the final customers through several intermediaries. In each phase of the supply chain from purchase of the components (or raw materials) until delivery of the finished goods there is a competition with other actors doing similar work. The raw materials are finite as it is the demand for the finished goods. The economic agent will have to negotiate quantities, prices, quality, delivery times, terms and conditions [1] with the suppliers and the customers; the process is time consuming, requires skills and constant attention. The goal is to use an automated system to perform the negotiations based on the current needs which will optimize the costs. The academic community has even developed a benchmark system, as a game, in which various software agents can compete in order to validate their approaches and implementations [2].

The scope of the research beginning with this overview is to investigate the existing negotiation algorithms and protocols and to develop them using new approaches taken from artificial intelligence unexplored (or insufficiently explored) until now. The supply chain management will be used as a test domain for validating the new approaches.

II. NEGOTIATION

To start with a quote, “negotiation is something that everyone does, almost daily”. Even if it is used by most of us, almost daily, few really understand what it means [3].

Negotiation is an interactive process between two or more parties seeking to find common ground on an issue or issues of mutual interest or dispute where the involved parties seek to make or find a mutually acceptable agreement that will be honored by all the parties concerned. Negotiation occurs for several reasons: to agree on how to share or divide a limited resource; to create something new that neither party could attain on his or her own; to resolve a problem or dispute between the parties [3].

In order to design an automated negotiation system three basic components are used: negotiation protocol, negotiation subject and negotiation strategies (a more detailed classification scheme can be found in [6]).

The *protocols* are the rules that negotiation participants must obey when negotiating. They describe the circumstances under which the interactions between the agents take place: what deals can be made and what sequences of offers are allowed. The *negotiation subject* is what is negotiated between the parties (also known as *negotiation object*). An agent's *negotiation strategy* is the specification of the sequence of actions (usually offers or responses) the agent plans to make during the negotiation.

In [9] any negotiation setting, four basic components are used: a negotiation set (space of possible proposals that agents can make.), a protocol (the legal proposals that agents can make), a collection of strategies (one for each agent, which determine what proposals the agents will make) and a rule (that determines when a deal has been struck, and what this agreement deal is).

The negotiation protocol, subject and strategies for the agents involved make a *negotiation mechanism* [6].

The design of an appropriate negotiation scenario depends on a number of parameters.

The *subject of negotiation* may be a state of the environment in which the parties to reach an action that would like to perform or an item would like to have. If the subject is composed of multiple attributes it is called *multi-*

issue, otherwise it is called *single-issue* [7].

Interactions between agents can be classified in terms of the number of participants in the negotiation [6][9].

One-to-one negotiation - one agent is negotiating with exactly one other agent. A particularly simple case of one-to-one negotiation is the one where agents involved have symmetric preferences with respect to the possible deals.

Many-to-one negotiation - several agents negotiate with a single agent. This is the standard setting of actions where one agent plays the role of the seller, while the other ones play the role of buyers.

Many-to-many negotiation - many agents negotiate with many other agents. This scenario requires the most complex automated negotiations research.

Negotiation can be categorized in different ways. Research in negotiations has been done heavily, resulting in the description of several methods of negotiation. Below are just a few ways that we can look at negotiation: distributive, integrative, zero-sum, win-win, positional, "interest-base", competitive, cooperative and claiming value - creating value [10].

A *distributive negotiation* type is a process that normally entails a single issue to be negotiated. The single issue often involves price and frequently relates to the bargaining process. Also referred to as "win-lose" or "fixed-pie" negotiation because one party generally gains at the expense of another party, "win-win" negotiation is conversely often referred to as integrative negotiation.

A distributive negotiation will focus on the division of a set amount of resources, largely determined by the aspiration price (the maximum that party A would like to get, and which is also the least amount that party B would like to pay), and the reservation price (the least amount that party A would accept, and the maximum that party B would prepared to pay). Well over 90% of all civil lawsuits in the America are settled out of court, and most are largely resolved through the application of a distributive negotiation.

Integrative negotiation (also called "*interest-based bargaining*," "win-win bargaining") is often referred to as "win-win" (everybody wins something, usually) and typically entails two or more issues to be negotiated. It often involves an agreement process that better integrates the aims and goals of all the involved negotiating parties through creative and collaborative problem solving. Relationship is usually more important, with more complex issues being negotiated than with distributive negotiation. In other words integrative bargaining is a negotiation strategy in which parties collaborate to find a "win-win" solution to their dispute. This strategy focuses on developing mutually beneficial agreements based on the interests of the disputants. Interests include the needs, desires, concerns, and fears important to each side. They are the underlying reasons why people become involved in a conflict.

Win-lose negotiation refers to a distributive negotiation whereby one party's gain is another party's loss. Both parties are competing to get the most value from the

negotiation. This negotiation type is frequently referred as "fixed-pie" scenario because it is used when there is a limited amount of resources to be distributed. The term is used in game theory and economics, as well, to describe similar situations.

A *win-win negotiation* settlement is an integrative negotiated agreement. In theory this means the negotiating parties have reached an agreement after fully taking into account each others' interests, such that the agreement cannot be improved upon further by any other agreement. By definition, there are no resources or 'gold' left on the table and all creative options have been thoroughly exploited. "Win-win" has its roots in Economics Game Theory [12].

Zero-sum, positive-sum, and negative-sum are all game theory terms that refer to the outcomes of a dispute or negotiation. They refer to the actual amount of wealth (money, land, vacation time) -- measurable rewards -- that each party receives. Though similar, these terms differ from the terms "win-win, win-lose, and lose-lose" which refer to wins or losses relative to expectations. In a *zero-sum situation*, it is impossible for one party to advance its position without the other party suffering a corresponding loss. If one side gets \$1,000 more, that means the other side gets \$1,000 less. The wins and losses add up to zero. These situations typically arise in distributive bargaining cases where a "fixed pie" must be divided between the parties. For example, if two departments are splitting up a fixed amount of money, the more one gets the less available to the other. Or, if there is only one job, one person will get it and the other person will not. (One job won and one job lost equals zero). *Positive-sum* outcomes are those in which the sum of winnings and losses is greater than zero. This becomes possible when the size of the pie is somehow enlarged so that there is more wealth to distribute between the parties than there was originally, or some other way is devised so everyone gets what they want or need. The most difficult problems are *negative-sum situations*, where the pie is shrinking. In the end, the gains and losses will all add up to less than zero. This means that the only way for a party to maintain its position is to take something from another party, and even if everyone takes his or her share of the "losses," everyone still loses in comparison to what they currently have or really need. This type of situation often sparks serious competition.

Positional bargaining is a negotiation strategy that involves holding on to a fixed idea, or position, of what you want and arguing for it and it alone, regardless of any underlying interests. The classic example of positional bargaining is the haggling that takes place between proprietor and customer over the price of an item. The customer has a maximum amount she will pay and the proprietor will only sell something over a certain minimum amount. Each side starts with an extreme position, which in this case is a monetary value, and proceeds from there to negotiate and make concessions. Eventually a compromise may be reached. For example, a man offers a vendor at the

flea market \$10 for a rug he has for sale. The vendor asks for \$30, so the customer offers \$15. The merchant then says he will accept \$25, but the customer says the highest he will go is \$20. The vendor agrees that \$20 is acceptable and the sale is made at \$20. So the customer pays \$10 more than he originally wanted and the vendor receives \$10 less.

Creating and claiming value are two of the most fundamental aspects of negotiation strategy that exist in tension with one another. In any negotiation, the parties must decide whether to be competitive, cooperative, or some of both. (David Lax and James Sebenius call this the "negotiator's dilemma" [13]. It is similar to the "prisoners' dilemma" in game theory [14], because the best outcome for one person is not necessarily the best for both, but if both pursue their best option, they will often both get the worst outcome.) Value is created (or the "pie is enlarged") in negotiations through the cooperative process of integrative or interest-based bargaining. This means that the parties in a dispute have managed to find ways to increase the amount of beneficial goods (things they want or that will make their situation better than before) that will be divided between them. This may also be called "joint value" or "joint gains," meaning that new developments are considered improvements by both sides.

Difference between bargaining and negotiation is that bargaining often describes the competitive, win-lose situation and negotiation refers to win-win situation. Next we briefly describe three major themes, namely: the definition and the basic characteristics of negotiation situation, interdependence and understanding the conflict management.

The characteristics of a negotiation situation are as follows: in the negotiation process involving two or more parties, in this situation there is a conflict of needs and desires, the parties think they can get a better deal and parties expect a "give and take" process [3].

In negotiation, parties need each other to achieve their preferred outcomes or objectives. This mutual dependency is called interdependence, interconnected or interrelated. Interdependence also means reciprocally connected. Things that are interdependent are connected to one another in more than one way. Interdependent goals are an important aspect of negotiation, so in situation win-lose, I win, you lose and in case win-win are opportunities for both parties to gain.

Interdependent negotiation [15] is *competitive* (distributive) - interaction is considered a zero sum game (win-lose) and the goal is victory, maximizing the benefits of unilateral (winner takes all). Also, interdependent negotiation is *cooperative* (integrative) - participants are aware of the potential benefits of an agreement and act cooperative, considering the situation of negotiation a positive sum game (win-win) and aim is to achieve a mutually beneficial distribution of benefits.

Interdependent parties are characterized by interlocking goals [15]. Having interdependent goals does not mean that everyone wants and needs exactly the same thing. Most interdependent relationships are characterized by mix of

convergent and conflict goals. Often the two sides' goals are linked together, or interdependent. The parties' interaction will be shaped by whether this interdependence is positive or negative, because there are several interrelated interests and objectives. *Positive interdependence* of goals is when a party chance of achieving a particular goal is higher when the other side achieves it as well. *Negative interdependence* is when a party chance to achieve the goal is lower when the other party carries out its goal.

Negotiation outcomes and shape processes are affected by the types of interdependent and the nature of the situation. Zero-sum and distributive methods have result one winner while non-zero-sum or integrative methods have result a mutual gains situation.

There are alternative shapes of interdependence such as evaluating interdependence depends heavily on the alternatives to working together, the desirability to work together is better for outcomes best available alternative: BATNA (acronym for Best Alternative to a Negotiated Agreement [16]). It is the alternative action that will be taken should your proposed agreement with another party result in an unsatisfactory agreement or when an agreement fails to materialize. If the potential results of your current negotiation only offer a value that is less than your BATNA, there is no point in proceeding with the negotiation, and one should use their best available alternative option instead. Prior to the start of negotiations, each party should have ascertained their own individual BATNA.

III. SUPPLY CHAIN

A supply chain is a system of organizations, people, technology, activities, information and resources involved in moving a product or service from supplier to customer [17]. Supply chain activities transform natural resources, raw materials and components into a finished product that is delivered to the end customer. In sophisticated supply chain systems, used products may re-enter the supply chain at any point where residual value is recyclable. Supply chains link value chains [18].

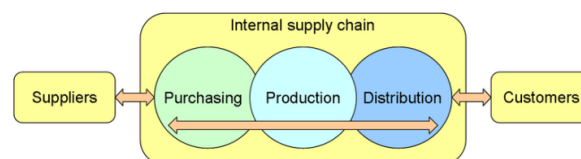


Figure 1: An illustration of a company's supply chain

The supply chain management (SCM) is the management of a network of interconnected businesses involved in the ultimate provision of product and service packages required by end customers [4]. Supply chain management spans all movement and storage of raw materials, work-in-process inventory, and finished goods from point of origin to point of consumption [17].

A supply chain, as opposed to supply chain management, is a set of organizations directly linked by one or more of the upstream and downstream flows of products, services,

finances, and information from a source to a customer. Managing a supply chain is “supply chain management” [19].

Companies are required to comply with customer orders even if it may be hard to do so. Companies must respond quickly and effectively to customer requests in a timely manner. Unexpected urgent orders, in most cases, delivery delays and less efficient. Unexpected urgent orders, in most cases, delivery delays and less efficient. Negotiation techniques are used to overcome conflicts and coalitions, and to come to an agreement among agents [20].

IV. RELEVANT WORKS

Various approaches have been used for implementing negotiations in electronic systems. The most used and successful is the one using software agents. A software agent is a piece of software that acts for a user or other program in a relationship of agency, having the authority to decide which actions are appropriate in order to reach a predefined goal [21]. The agents have to learn, collaborate and be autonomous. The agents run in multi-agent systems (MAS) in which different agents (running on behalf of different users with different interests) have to interact and take actions to reach goals (predefined or self imposed). [22] identifies four general issues, which need to be addressed for software agents in a multi-agent system for supply chain management: communication, representation, problem solving and human interaction. The behaviors of a software agent, which correspond to the above four aspects of negotiation, are called *negotiating actions*. A software agent is a *negotiating agent* if it can at least take communicating and problem solving actions in a specific domain.

All of the negotiating agents have some understanding of system ontology and use a certain Agent Communication Language (ACL) to make conversation [22].

As mentioned in section II the negotiations can be of several types. The interrelated negotiations (subject of this paper) have been addressed by several works published in recent years.

Yain-Whar Si, David Edmond and others [23] have described a model which allows for constructing trading engines capable to participate concurrently in multiple interrelated negotiations using heterogeneous protocols. The behavior of the trading engines is specified using a generic synchronization construct. The construct can use a number of well known strategies from the financial trading domain to build incrementally complex trading schemes. It is improved with a priority-based scheduling algorithm which selects a set of nodes for negotiations, based on their estimated profit, remaining time and concurrency degree needed.

In order to specify trading strategies they are split in elementary trading activities (ETA). Each ETA is defined as a set of operations required for reaching a trading agreement (a deal). The operations are usually structured in 3 phases: discovery of trading partner, negotiation and trade settlement. Several ETA can be grouped into a Composite

Trading Activity (CTA). The ETAs composing a CTA can be independent or interrelated. The interrelations between ETAs can be of 2 types: complementary or alternative [23]. A complementary relationship consists in ETAs which need either to all succeed or no one to succeed. Alternative relationship involves that at most one of the ETA has to be successful. A CTA can contain both alternative and complementary activities. In order to allow for concurrent negotiations during various ETA in complex CTAs the authors describe a model for the representation of CTA and ETAs, for performing synchronization and issuing iterative negotiation instructions essential in any complex trading environment. The results were promising, the system being one of the first of its kind.

The research was further developed in [24] in order to reduce the role of human intervention in synchronization. The provided algorithm generates negotiation plans with the exact time, duration and trading activities which are going to be executed [25].

A different model is developed in [26] to address multi-linked negotiations. A multi-linked negotiation problem occurs when an agent needs to negotiate with multiple other agents about different subjects (tasks, conflicts, or resource requirements), and the negotiation over one subject has influence on negotiations over other subjects. The paper details several techniques to address this type of negotiations. The first one uses a partial-order schedule and a measure of the schedule, called flexibility, to enable the agent to explicitly reason about the interactions among multiple negotiations. The second one defines a model for the multi-linked negotiation problem. The model is used to develop a heuristic search algorithm for finding a near-optimal ordering for negotiation issues. By using the algorithm the agent is able to evaluate and compare different negotiation approaches and choose the best one.

The experimental results show the efficiency of using the heuristic search algorithm without performing a full-search.

The technique is further developed in [33] with application in multi-agent systems (MAS). In a multi-task, resource sharing environment, an agent needs to deal with multiple related negotiation issues including: task contracted to other agents, task requested by other agents, external resource requirements for local activities and interrelationship among activities distributed among multiple agents. These issues are related to each other. The result of one issue influences the possible solutions for the other issues. The relationships among these negotiation issues are classified in two types: *directly-linked* relationships and *indirectly-linked* relationships.

[29] presents the design of an algorithm used by agents involved in several simultaneous English auctions for buying more goods. The algorithm ensures the coordination of appropriate bids in different auctions to buy a fair number of products. It is augmented (coordinated) with a second algorithm used to determine which bid is the maximum amount of an agent in an auction is about to realize. By interleaving the application of the two algorithms the agent

can effectively participate in multiple English auctions. It will use the coordination algorithm to place lowest possible bids across auctions. It will use the bid withdrawal algorithm to determine when it is worth bidding higher in an auction which is about to terminate, as opposed to transferring to other auctions where the active bids are currently lower.

A protocol to support many-to-many bilateral multiple-issue negotiation in a competitive environment is developed and presented in [31]. The protocol can be used in the context of service-oriented negotiation, where one or more self-interested parties will provide services to one or more other parties. It extends existing negotiation protocols and enables both service requestors and service providers to manage several negotiation processes in parallel. Moreover, this protocol mitigates the situation where most one-to-many negotiations are biased in favor of one participating agent, and reduces the de-commitment situation for both participants. The paper assumes all agents are self-interested and have their own preferences about the services.

But how can an agent deal with multiple related negotiations? A possible answer is presented in [32] by answering two other questions. The first one is in what order should the negotiations be performed and the second one is how the agent assigns values for the attributes (also referred as “features”) in negotiation used to minimize the conflicts and maximize the utility. The paper describes a decision-making process that enables the agent to manage the multi-linked negotiations and choose the appropriate solution based on knowledge about each negotiation and the interrelationships among them. In order to effectively engage in multiple concurrent negotiations the agent must carefully coordinate them. In general, coordination of multiple concurrent negotiations is a sensitive issue and [30] presents a heuristic model for coordinating multiple bilateral negotiations. The mathematical model is evaluated and shown to be effective in various scenarios of negotiation. It is useful especially when there are large numbers of participants and when the agent is given sufficient times to reach a deal. The model for the buyer agent consists of two main components: a *coordinator* and a number of *negotiation threads*. The negotiation threads deal directly with the various sellers (one per seller) and are responsible for deciding what counter-offers to send to them and what proposals to accept. The coordinator is the most important component of the buyer. It is responsible for coordinating all the negotiation threads and choosing an appropriate negotiation strategy for each thread.

A concurrent negotiation problem occurs when an agent needs to negotiate with multiple agents to reach agreement.

Negotiation chains are analyzed in [27]. A negotiation chain is formed when multiple related negotiations are spread over multiple agents. In order to appropriately order and structure the negotiations occurring in the chain so as to optimize the expected utility the authors conceive and develop an extension to a single-agent concurrent negotiation framework. The extension can be used in the case of semi-cooperative multi-agent systems, where agents

do not cheat or intend to hurt other agents even as they focus on their own goals. The agents can use a pre-negotiation phase in which to exchange meta-level information. With this information the agents can build a better model of the negotiation process. This more accurate model helps the agent in choosing a good negotiation solution. The agent can also use this information to allocate appropriate time for each negotiation, hence to find a good ordering of all related negotiations. The experimental data shows that these mechanisms improve the agents’ and the system’s overall performance significantly.

All these approaches share a few common characteristics. They use trees to represent the negotiation chains or dependencies. Selection of the path through the tree is done using a complete search or by applying some kind of heuristic to speed up the search result.

Modern AI techniques can be used for designing the negotiation agents. Zang and Wong present in [11] a negotiation agent that applies Case-Based Reasoning (CBR) techniques to capture and re-use previously successful negotiation experiences in the multi-agent negotiation context. This negotiation agent provides adaptive negotiation strategies that can be generated dynamically and are context-sensitive. The paper presents the negotiation agent in the context of used car trading done with the help of a web-based application for used car trading. Relevant issues are discussed, including formulation of negotiation episodes, negotiation experience representation, negotiation experience matching/selection and previous negotiation experience adaptation/reuse.

In order to test the agents and algorithms used in negotiation the academic community has developed a platform and a competition, trading agent competition (TAC) [28] in the form of a game. There are 5 sections each dealing with a different type of trading and negotiations (energy trading, lemon stand game, ad auctions, market design game and supply chain management). Entries in the Trading Agent Competition are agents designed to trade in electronic markets. The agents operate autonomously in the market - sending bids, requesting quotes, accepting offers, and generally negotiating deals according to market rules. The trading behavior is fully automated in that the humans do not intervene while the negotiation is in progress.

Trading agents face a most challenging task. To play the market effectively, an agent must make real-time decisions in an uncertain and fast-changing environment, taking account of other agents doing the same. Capable agents rapidly assimilate market information from many sources, forecast future events, optimize complex offers and resource allocations, anticipate strategic interactions, and learn from experience. Successful trading agents adopt and extend state-of-the-art techniques from artificial intelligence, operations research, statistics, and other relevant fields.

V. CONCLUSIONS AND FUTURE WORKS

The research in the automated negotiations field is open; there are still many possibilities for improvements and

evolutions. The overview will be followed by a full state of the art document which will take a snapshot of the current status in the domains and will highlight open issues and possible solutions. Based on these findings a couple of research papers will contour the final direction for future studies. It will focus on applying proved AI techniques (like CBR and NN) in solving the current problems lurking in the negotiations for SCM systems.

REFERENCES

- [1] N. Jennings, S. Parsons, C. Sierra, and P. Faratin, "Automated negotiation," in *Proc. 5th Int. Conf. on Practical Application of Intelligent Agents and Multi-Agent Systems*, Manchester, UK, 2000, pp. 23–30.
- [2] Trading Agent Competition (TAC) Supply Chain Management (SCM) game, Available: <http://www.sics.se/tac/page.php?id=13>
- [3] Roy Lewicki, David Saunders, Bruce Barry, "Negotiation 5th Edition", 2005.
- [4] C. M. Harland, "Supply Chain Management, Purchasing and Supply Management, Logistics, Vertical Integration, Materials Management and Supply Chain Dynamics," in Slack, N Encyclopedic Dictionary of Operations Management, (ed.) Blackwell, UK, 1996.
- [5] Chester L. Karrass, "In Business As in Life, You Don't Get What You Deserve, You Get What You Negotiate", Stanford Street Pr, June 1996.
- [6] A. R. Lomuscio, M. Wooldridge and N. R. Jennings. "A classification scheme for negotiation in Electronic Commerce" in *Proceeding of Agent-Mediated Electronic Commerce: A European AgentLink Perspective*, Springer-Verlag, Berlin, 2000, pp. 19-33.
- [7] Mihnea Scafeş, "Contributions to multi-issue negotiation mechanisms between self-interested agents," February 2nd, 2010.
- [8] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, "Automated negotiation: Prospects, methods and challenges," *International Journal of Group Decision and Negotiation* 10/2, 2001, pp. 199-215.
- [9] Wooldridge, M. "An Introduction to Multi-Agent Systems," John Wiley & Sons, 2002.
- [10] The Negotiation Institute: Training & Seminars, [Online], Available: <http://www.negotiation.com>
- [11] Dong Mei Zhang, Wai Yat Wong, "A Web-Based Negotiation Agent Using CBR," in *Proc. Revised Papers from the PRICAI 2000 Workshop Reader, Four Workshops held at PRICAI 2000 on Advances in Artificial Intelligence*, 2001, London, UK, pp. 183-195.
- [12] Theodore L. Turocy, Bernhard von Stengel, "Game Theory," *CDAM Research Report* LSE-CDAM-2001-09, 2001.
- [13] Lax, David and James Sebenius. "The Manager as Negotiator: The Negotiator's Dilemma: Creating and Claiming Value," in Goldberg, Stephen, Frank Sander and Nancy Rogers, eds. *Dispute Resolution*. 2nd ed. Boston, MA: Little, Brown and Co., 1992, pp. 49-62.
- [14] S. Plous, "Prisoner's Dilemma or Perceptual Dilemma?" in *Journal of Peace Research*, Vol. 30, No. 2, 1993, pp. 163–179.
- [15] R. J. Lewicki, D. M. Saunders, B. Barry, J. W. Minton, "Essentials of Negotiation (3rd edition), New York, NY: McGraw-Hill/Irwin, 2004.
- [16] Best alternative to a negotiated agreement, [Online] Available: http://en.wikipedia.org/wiki/Best_alternative_to_a_negotiated_agreement
- [17] Anna Nagurney "Supply Chain Network Economics: Dynamics of Prices, Flows, and Profits," Edward Elgar Publishing, 2006.
- [18] Chen, I. J., Paulraj, "A. Towards a theory of supply chain management: the constructs and measurements," in *Journal of Operations Management*, Vol. 22, No. 2, 2004, pp. 119-150.
- [19] J. T. Mentzer et. al. "Defining Supply Chain Management," in *Journal of Business Logistics*, Vol. 22, No. 2, 2001, pp. 1–25.
- [20] Saberi S., Makatsoris C., "Multi-agent system for negotiation in supply chain management," in *Proceedings of The 6th International Conference on Manufacturing Research (ICMR08)*, Brunel University, UK, 2008, pp. 311-317.
- [21] Nwana, H.S., "Software Agents: An Overview," *Knowledge Engineering Review*, Vol.11, No.3, 205-244, Cambridge University Press, 1996. Available: <http://www.sce.carleton.ca/netmanage/docs/AgentsOverview/ao.html>
- [22] Ye Chen et al., "Negotiating agents for supply chain management," in *Proc. Proceedings of the AAAI Workshop on Artificial Intelligence for Electronic Commerce*, 1999, Orlando, Florida, pp. 113-114.
- [23] Si Yain-Whar, Edmond David, ter Hofstede, Arthur H.M., & Dumas, Marlon, "Orchestrating Interrelated Trading Activities," in *International Journal of Business Process Integration and Management*, Vol. 1, No. 1, 2005, pp. 12-25.
- [24] Si Yain-Whar, Edmond David and Dumas Marlon and ter Hofstede, Arthur H. M., "Specification and Execution of Composite Trading Activities," *Electronic Commerce Research*, 7(3-4), 2007, pp. 221-263.
- [25] Y. Matsumoto, S. Fujita. "An auction agent for bidding on combinations of items," in *Proceedings of the 5th International Conference on Autonomous Agents*, (AGENTS'01), Montreal, Canada, 2001, pp. 552-559.
- [26] Xiaojin Zhang, Victor Lesser, Sherief Abdallah, "Efficient Management of Multi-Linked Negotiation Based on a Formalized Model," in *Autonomous Agents and Multi-Agent Systems*, Vol. 10, Num. 2, 2005, pp. 165-205.
- [27] Zhang Xiaojin, Victor Lesser, "Solving Negotiation Chains in Semi Cooperative Multi-Agent Systems," in *Proc of 6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007)*, Honolulu, Hawaii, USA, 2007, pp. 939-946.
- [28] Trading Agent Competition (TAC) association for trading agent research, Available: <http://tradingagents.org/>
- [29] Chris Preist, "Algorithm Design for Agents Which Participate in Multiple Simultaneous Auctions," in *Proceeding Agent-Mediated Electronic Commerce III, Current Issues in Agent-Based Electronic Commerce Systems (includes revised papers from AMEC 2000 Workshop)*, London, UK, 2001, pp. 139-154.
- [30] Thuc Duong Nguyen, Nicholas R. Jennings, "Coordinating multiple concurrent negotiations," in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-agent Systems*, Volume 3, New York, USA, 2004, pp. 1064-1071.
- [31] Jiangbo Dang, Michael N. Huhns, "An Extended Protocol for Multiple-Issue Concurrent Negotiation," *Proceedings of the 20th national conference on Artificial intelligence*, Volume 1, Pittsburgh, PA, 2005, pp. 65-70.
- [32] Xiaojin Zhang, Victor Lesser, Sherief Abdallah, "Efficient Ordering and Parameterization of Multi-Linked Negotiation," in *Proceedings of the second international joint conference on Autonomous agents and multi-agent systems (Extended abstract)*, Melbourne, VIC, Australia, 2003, pp. 1170-1171.
- [33] Xiaojin Zhang, Victor Lesser, "Multi-Linked Negotiation in Multi-Agent System," in *Proceedings of the first international joint conference on Autonomous agents and multi-agent systems*, Bologna, Italy, 2002, pp. 1207-1214.

Eye Tracking Based Communication System for Patient with Major Neuro-locomotor Disabilities

Robert G. Lupu, Radu G. Bozomitu, Florina Ungureanu and Vlad Cehan

Abstract— This paper presents a new technology used for communicating with people with major neuro-locomotor disability by determining gaze direction on a monitor screen. Gaze direction is determined by pupil position through image analysis. The proposed communication technology ensures communication with the patient as follows: (1) on a monitor screen divided in four sections are displayed key words and corresponding ideograms; (2) the patient sees the key words and focuses his/her gaze on a certain section – a pointer allows the control of gaze direction; (3) an infra-red mini/micro video camera placed on the patient's head sends the image of the eye to a computer; (4) the focusing of the patient's gaze on a certain key word i.e. a certain section of the screen is identified by image analysis; (5) voluntary blinking, also objectified by image analysis, shows the selection of a certain word. The patient's wish or need, represented by the key word, is then sent to a server and from there to a caretaker by means of a wire or a wireless transmission system. The absence of a selection within a certain time period causes the displaying of a new set of ideograms.

The new technology is useful in the case of the patients who cannot communicate verbally, through signs or in writing and is based on their ability to control eye movement, which is often the case with people with major locomotor disabilities.

I. INTRODUCTION

A report from 2007 of the World Health Organization (WHO) shows that neurological disorders, ranging from epilepsy to Alzheimer disease, and from stroke to headache affect up to one billion people worldwide. Neurological disorders also include brain injuries, neuroinfections, multiple sclerosis and Parkinson disease [1]. Communicating with this kind of people is a challenge even today. Those with aural or visual disabilities can use sign language, hand writing or Braille alphabet as a way of expressing themselves. People with severe neurological

disorders and especially those with major neuro-locomotor disabilities who can produce only inarticulate sounds or make certain movements or perform uncontrolled muscular contractions, can use a board with ideograms or an eye gaze communication board (e-tran) in order to communicate. However, this requires constant presence of a caretaker. Taking into account that there are periods of time when the patient does not want or need to communicate, the caretaker has to wait near the bedside or to check from time to time the needs of the patient. This raises the medical insurance costs.

Using computer-based systems and technological advances in communication systems such as graphical user interface (GUI)-driven computer systems[2], we can facilitate communication with these patients.

Human interaction with a computer based system endowed with graphical user interface may be classified in two categories:

- pointing: moving the pointer over an object, text or icon on the screen;
- selection: left or right click, action interpreted as a selection of that object from the corresponding area.

These operations are very easy to perform for healthy people but they raise big problems for those with disabilities. There are solutions like tongue movement detection device, touch screens, but these can be used only by those capable of fine movements. However, even patients with neuro-locomotor disabilities may have fine eye movements which can be detected and used to point



Fig. 1. Video cam position for eye tracking

Manuscript received May 30, 2011. This work was supported by the Romanian Government under PN-II grant (PN-II, ASISTSYS 12-122/2008).

R. G. Lupu is with the "Gh. Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering, 700050 Romania (fax: +40232231343; e-mail: robert@cs.tuiasi.ro).

R. G. Bozomitu is with the "Gh. Asachi" Technical University of Iasi, Faculty of Electronics, Telecommunications and Information Technology, 700506 Romania (fax: +40232217720; e-mail: bozomitu@etti.tuiasi.ro).

F. Ungureanu is with the "Gh. Asachi" Technical University of Iasi, Faculty of Automatic Control and Computer Engineering, 700050 Romania (fax: +40232231343; e-mail: fungurea@cs.tuiasi.ro).

V. Cehan is with the "Gh. Asachi" Technical University of Iasi, Faculty of Electronics, Telecommunications and Information Technology, 700506 Romania (fax: +40232217720; e-mail: vlcehan@etti.tuiasi.ro).

objects on the screen. This can be done by measuring and interpreting biopotentials at the level of eye balls through non-invasive techniques using electrodes placed above and underneath the eyes and to their left and right to detect up and down and left – right movement. This method is called electro-oculogram [3].

Another method, different from electro-oculogram one, is eye tracking method which involves the use of a video cam and image processing technique to determine the position of

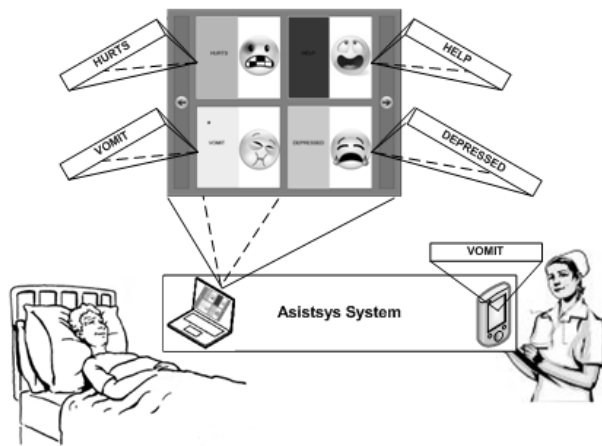


Fig. 2. Asistsys communication system

the eye. The video cam is placed near the eye in order to take only the image of the eye, as is presented in Fig. 1. Corresponding to eye position, a cursor is moved inside a certain area of the screen. The selection is made when eye blink is detected. To eliminate fake selections by involuntary blinks, the patient has two possibilities: close the eyes and keep the eyes closed until a beep is heard, or winking two or three times in a short period of time.

The proposed system uses eye tracking technique, facilitating selection of a key word and sends it to caretaker. In this way the patient no longer needs the continuous presence of the caretaker.

II. SYSTEM PRINCIPLE

The communication system that offers to the patient the possibility to express his needs whenever he wants even he is alone, is depicted in Fig. 2.

In front of the patient's eyes, a computer based unit displays keywords on a screen together with a suggestive icon and background colour. Whenever the cursor floats over a certain group (keyword – image) the area is highlighted and the keyword is heard into the headphones. The selection is made detecting a voluntary blink. The selected keyword is sent wirelessly to caretaker's PDA/smartphone. The caretaker notices the patient request and takes action.

The proposed system, named ASISTSYS, has the block diagram presented in Fig. 3:

- AsistsysPatient - a computer based device assisting the patient in keyword selection process and ensures the communication with the server;
- AsistsysServer - manages all the information

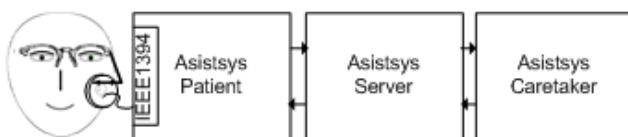


Fig. 3. System block diagram

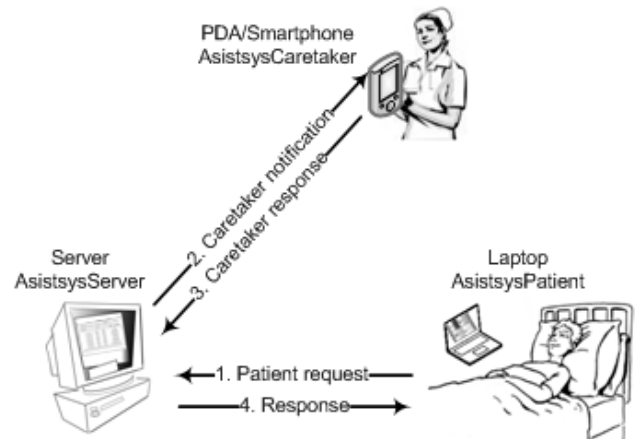


Fig. 4. Asistsys system – communication flow

received/forwarded;

- AsistsysCaretaker - identified as a PDA or smartphone announces the caretaker about patient request.

The communication flow is shown in Fig. 4. Briefly, the system works like this: the patient makes a selection and the keyword is sent via Ethernet to the server. There, based on the information in the database, is selected the caretaker who has to answer and the keyword is sent wireless to his/her PDA. The caretaker's answer is sent back to the server and from there to the patient's device screen.

The major advantage of the proposed system is that it replaces the traditional manner of communicating, which relies on the constant presence of a caretaker at the patient's bedside, raising thus the cost of patient care taking [3].

The patient gaze direction is determined by the eye tracking technique. To perform this, a video cam is placed very close to the eye right underneath avoiding to obstruct the patient's frontal visual field. The camera was modified to take images only in infrared light. To track eye movements an open source eye tracking Starburst algorithm was used [4]. The original open source code was written in C++ programming language for Linux based operating systems using OpenCV library. The source code was modified to be compiled with ANSI C compilers, more precisely with National Instruments LabWindows/CVI compiler because of already existing project developed in

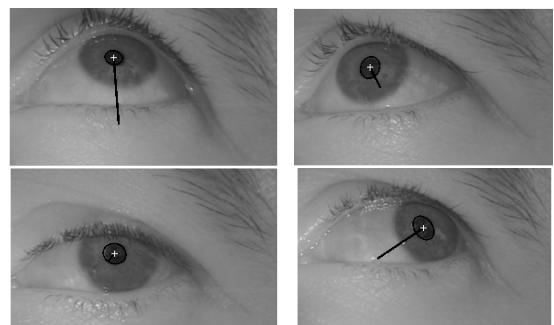


Fig. 5. Typical eye positions – left column (up, down), right column (left, right)

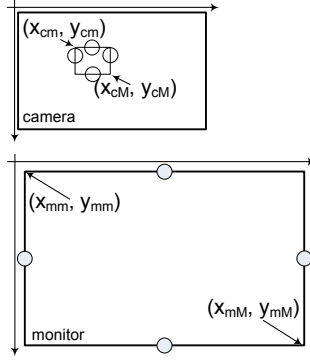


Fig. 6. Window-to-viewport transformation/mapping

this environment, for Windows operating systems. A conversion function from NI Vision Image data type to OpenCV IplImage data type was also implemented.

The typical eye positions for basic movements - up, down, left, right, shown in Fig. 5, defines a rectangle $((x_{cm}, y_{cm}), (x_{cM}, y_{cM}))$ as it is represented in Fig. 6.

For the monitor, it is assumed that $x_{mm}=0$, $y_{mm}=0$, $x_{mM}=x_{res}$ and $y_{mM}=y_{res}$. The relations (1) express the scaling factors s_x and s_y and for the same orientation of the axes the translation terms have the expressions from eq. (2). Relation (3) represents the matrix form for the window-to-viewport mapping.

$$s_x = \frac{\Delta x_m}{\Delta x_c} = \frac{x_{res}}{x_{cM} - x_{cm}} \quad (1)$$

$$s_y = \frac{\Delta y_m}{\Delta y_c} = \frac{y_{res}}{y_{cM} - y_{cm}}$$

$$\begin{aligned} t_x &= -s_x \cdot x_{cm} \\ t_y &= -s_y \cdot y_{cm} \end{aligned} \quad (2)$$

$$\begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x_c \\ y_c \\ 1 \end{bmatrix} \quad (3)$$

The patient needs to send the selected key word/ ideogram to Asistsys server and from here to the caretaker. The caretaker identified automatically by searching in the database receives the keyword on his mobile devices. The response is sent back to the patient via server. The TCP/IP protocol is used for communication between system devices, via local computer network in three ways: Ethernet (IEEE 802.3) and Wi-Fi (IEEE 802.11 b, g, n), Ethernet and GPRS, mixed. A more detailed communication diagram between Asistsys system components, when the AsistsysPatient is the initiator is shown in Fig. 7.

The keyword selected by the patient is sent to AsistsysServer. The server selects the caretaker from the

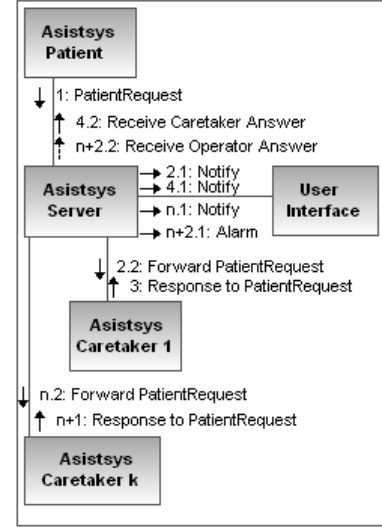


Fig. 7. Communication diagram within Asistsys system

database according to priorities and work load, and a corresponding keyword is forwarded to her/his mobile device. The positive or negative response, together with or without an optional sentence is sent back to the server. In case of positive response, the server forwards the caretaker's message to the patient's device. In case of negative or no response another caretaker is selected and the process is repeated. If no one sends a positive response, then the server operator is announced and he can take action corresponding to patient's needs.

All the messages sent trough the server, are recorded in log files. This way can be generated statistical reports like caretakers load, the use of the system and so on.

The Asistsys system is a client/server 3 tier model [5] system architecture where AsistsysPatient and AsistsysCaretaker are the clients (Fig. 8). The server accesses a database with patients, caretakers and logs and manages all the communication in the system.

The caretaker application is written in Java for a better portability. It is always connected to the server through radio (Wi-Fi/GPRS) receiving keywords from it. The keywords are added to a list on the screen from where the caretaker can select the patient and answer.

On the other side, the AsistsysPatient application is constantly connected to the server through Ethernet/Wi-Fi. As it can be seen in Fig. 8, AsistsysPatient is structured in three major layers. The input layer controls the camera and image acquisition. The images are preprocessed (horizontally flipped, 640x480 resampling, grayscale conversion) before being added into memory buffers. The middle layer takes images from safe thread memory buffers, detects the position of the pupil and simulates the user (patient) input by moving the cursor on the screen. Every time the cursor enters in a certain area, that area is highlighted and the associated keyword can be heard at the

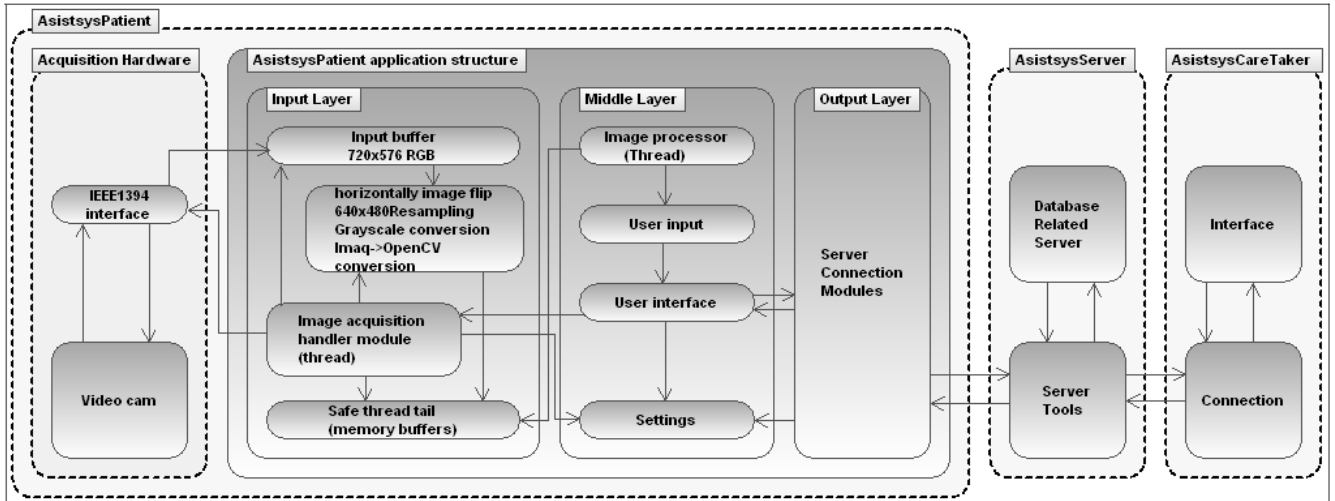


Fig. 8. System flowchart

loudspeakers. In that moment, the selection can be made by closing the eyes. Once the keyword has been selected, the third layer sends it to the server. Messages received from the server are displayed on the screen by the same layer.

The graphic user interface of AsistsysPatient (PC screen), is divided into four sections, in which keywords are displayed on a corresponding background color, as it is illustrated in Fig. 9. Close to every keyword, an appropriate ideogram is also displayed. The keywords window changes automatically every n seconds or can be changed by the patient, selecting left or right pointing arrows. A keyword should symbolize not only a desire, a necessity or a demand but also a subset of other keywords.

The experiments carried out with some volunteers' help (students and patients) reveal that they need minimal training to learn how to point and select a keyword in order to use the system. Also "navigating" trough keywords in order to select the appropriate one is very easy to perform because it is intuitive and similar to any other contemporary

communication device.

III. SYSTEM TESTING

Tests were carried out in two stages. The first one was necessary to ensure a good functionality of the system. The second one certified the need of this kind of communication system for patient with neurolocomotor disabilities.

In the first stage every component was tested in part and in system. The tests took place in laboratory as follow:

1) *AsistsysServer*: two software products were developed in order to simulate a caretaker device (Fake Caretaker Device - FCD) and a patient device (Fake Patient Device - FPD). The FPD application was installed on twenty computers and FCD application on five computers. All computers were connected in a network. The FPD computers were configured to send randomly key words, to disconnect from and reconnect to server randomly, to display messages received from server. The FCD computers were configured to answer positive or negative (randomly) to any request sent by the server regarding FPD, to disconnect from and reconnect to server, to display received messages.

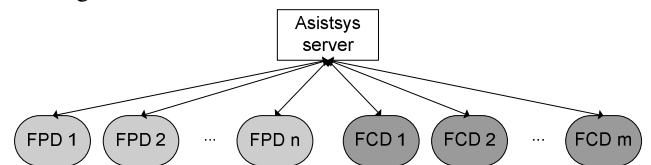


Fig. 9. Asistsys testing scenario

The communication protocol, the load balancing of FCD and the login process of FPD and FCD were tested in this test scenario.

2) *AsistsysCaretaker*: a small custom server was built in order to test the communication protocol and functionality of AsistsysCaretaker application. The same server was used in development process. The server was programmed to

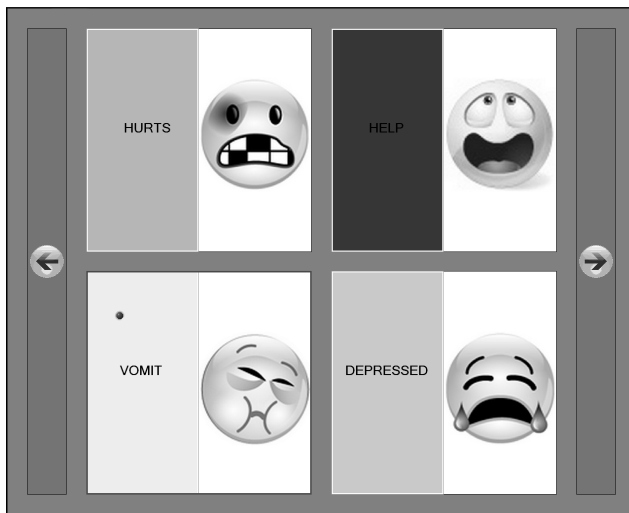


Fig. 9. AsistsysPatient print screen

automatically/manually send keywords. The graphical user interface functionality was also tested.

3) *AsistsysPatient*: firstly, the possibility to point and select an image from screen was tested. The number of images was varied from one to six. This test revealed that the mapping function allows pointing maximum four images. The arrow images are harder pointed and selected. Therefore an auto play was added. Images were changed if no action were taken in n seconds (from 2 to 6 seconds).

The communication protocol was tested using a custom small server configured to answer automatically to any sent key word. The close connection function was added in order to test the reconnection function of the *AsistsysPatient* application.

Finally, all three components were tested together.

In the second stage, the tests were carried out at “Bagdasar Arseni” hospital from Bucharest, Romania and five patients were involved. After a short presentation of the system, the patients started using the *AsistsysPatient*. Firstly, one image was displayed and selection was made with a switch. After accommodation with the system, the webcam was used and blink detection module was enabled, replacing the switch in selection process. Then, the tracking module was enabled and calibrated and patients were trained to move the pointer on the screen. After that, the number of images from the screen were increased gradually corresponding with the patient’s ability to point and select the image.

Also for the nurses and caretakers, training was necessary to use to *AsistsysCaretaker* device.

A questionnaire with a five-point Likert scale (from 5 = completely satisfied to 1 = completely unsatisfied) was used to rate the performance of the system. The questionnaire had ten questions, nine for patients and one for medical staff. The system was rated by the patients with 210/225 points and by the caretakers with 18/25 points. The staff included two females and three males with ages between 28–50 years and with 3–25 years experience in neurodisabled care.

IV. CONCLUSION

Asistsys, the proposed communication system for patients with neuro-locomotor disabilities is based on eye-tracking and can be used by patients with no controlled movement except eye movement. This system assists and offers to the patients a way of expressing their needs or desires in the absence of the caretaker. Once the patient accustoms himself with the system, a more complex list of keywords including, “I want my wife/husband to visit me” or “I want to see a movie” should be used.

Some improvements should be considered in the future. For a smooth cursor movement a filter for both axes is needed. That allows a virtual keyboard to be added, with

circularly arranged keys, to enable “typing” without blinking.

A lower cost can be achieved for *AsistsysPatient* device by porting the *AsistsysPatient* application on ARM devices with additional DSP core for image processing, HDMI or SVIDEO output and Linux operating system.

REFERENCES

- [1] Plus One Foundation, <http://plusonefoundation.org/statistics.html> (last visit April 2011).
- [2] Craig A. Chin and Armando Barreto, “Integrated electromyogram and eye-gaze tracking cursor control system for computer users with motor disabilities”, *Journal of Rehabilitation Research & Development*, Pages 161–174, Volume 45, Number 1, 2008.
- [3] R.G. Lupu, R. G. Bozomitu, V. Cehan, D.A. Cehan “A New Computer-Based Technology for Communicating with People with Major Neuro-Locomotor Disability Using Ocular Electromyogram”, *ISSE 2011*
- [4] D. Li, D. Winfield, D.J. Parkhurst, “Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches”, *Human Computer Interaction Program Iowa State University, Ames, Iowa, 50010*
- [5] Unified Modeling Language: Superstructure, version 2.0, p. 488 (2005). Available: <http://www.omg.org/spec/UML/2.0>

Time-delay Analysis of a Robotic Stereo Active Vision System

Gigel Macesanu, Sorin M. Grigorescu, *Member, IEEE*,
and Vasile Comnac, *Member, IEEE*

Abstract— The time-delay introduced by an image processing system is a crucial factor which plays an important role in the robustness and stability of an active vision system. In this paper, the authors present a feedback control analysis of a stereo active vision platform used to control the pan, tilt and zoom of the active stereo camera. The main goal of the analysis is to determine the maximum time-delay, or delay margin, which can be introduced by the image processing chain without affecting the stability of the overall machine vision system. The proposed procedure considers the investigation of the frequency domain using two indicators of relative stability: gain and phase margin. The obtained delay-margin is used to design the feedback control using the Hermite-Bihler theorem.

I. INTRODUCTION

REAL world scenes are usually characterized by objects moving in cluttered environments, which change their position and orientation (pose) stochastically. One of the main objectives of a stereo active vision system, with which nowadays robots are equipped, is to track such objects and reconstruct their poses in a virtual 3D space. In order to track these objects, the stereo camera's viewpoint and zoom must adapt along with the poses of the objects.

In many robotics applications, the image acquisition system is used with a constant orientation and zoom, this fact leading to physical limitations in the scene understanding process. In order to enhance the visual perception capabilities of an autonomous robot, the camera's extrinsic parameters (i.e. pose and zoom) must be adapted according to the imaged environment.

In this paper, a time-delay and feedback control analysis of a robotic stereo active vision system is presented. The idea of *visual feedback control*, or *active vision*, has been heavily investigated in the computer vision community [12], [15]. One of the first comprehensive papers on the usage of feedback information at the image processing level can be found in [13], where visual feedback is used to control a robot manipulation task. This process is also encountered under the name of *visual servoing* [3]. Visual servoing is usually divided into two categories [3], [13]: *position* and *image based*. Recent investigations on active vision involve the camera's zoom adaptation, which considers the problem of controlling the focal length in order to keep a constant-sized image of an object moving along the camera's optical axes [5], [17]. Also, a multi-view active object search is presented in [16], where multiple objects are detected using

the feedback information provided by a humanoid head.

The main goal of the research presented in this paper is to analytically compute the maximum value of the time delay which can be introduced by the image processing chain within an active vision system, without affecting the stability of active object tracking. Despite its large usage, to the best of our knowledge, there is no known publication that deals with the maximum values of the time delay component introduced by the machine vision software in the robot tracking system. For comparison, in [9], a design and optimization process for a visual control system is proposed for solving problems caused by large delays introduced by image processing algorithms. The closed-loop controller for time-delay systems can be chosen as is explained in [14]. The obtaining controller is tuned so that the closed-loop control system remains stable. In [1], a complete analytical description of all stabilizing parameters for a *Proportional-Integer* (PI) controller used over a communication network link is presented.

The rest of the paper is organized as follows. In Section II, the object recognition and 3D reconstruction methods used by our active vision system are introduced, along with the variables that have to be adjusted by the control system. The mathematical modeling of the considered active vision architecture is given in Section III, followed in Section IV by its stability analysis. Finally, conclusions are stated in Section V.

II. 3D OBJECT DETECTION AND TRACKING

A. Object Detection

In 3D vision, one of the most important issues to be solved is the distance computation between the imaged object and the stereo camera. This distance is determined through a calibrated stereo camera. The block diagram of the proposed object detection and 3D reconstruction chain is presented in Fig. 1. Firstly, the left and right images are acquired from two *Pan-Tilt-Zoom* (PTZ) cameras which form the hardware of the stereo vision system. Both images are then segmented and the object of interest recognized. Once the corresponding points of the object have been determined in the left and right images, its 3D location can be calculated using the stereo triangulation method [8]. The obtained 3D pose is further used as feedback information to control the orientation and zoom of the two PTZ cameras.

In the 2D image plane, the object of interest is segmented using a robust region-based segmentation method [7], while its recognition is performed using invariant Hu moments that are invariant to object rotation, translation and scaling. Based on the determined 2D image position of the object,

This work was supported by the Sectoral Operational Program Human Resources Development (SOP HRD).

The authors are with the Department of Automation, Transilvania University of Brasov, Mihai Viteazu 5, 500174, Brasov, Romania ({gigel.macesanu, s.grigorescu, comnac}@unitbv.ro).

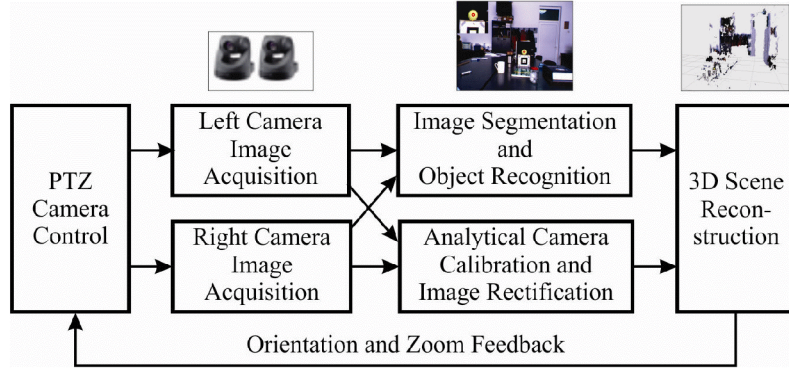


Fig. 1. The block diagram of the image processing chain.

the 2D correspondence points are calculated as the object's center of gravity in both images. Through the 2D positions of the correspondence points and the epipolar geometry constraints the 3D pose of the imaged object can be calculated [8].

The 3D pose of an object can only be obtained from rectified stereo images. Namely, the rectification process transforms each image plane in such a way that the epipolar lines become collinear and parallel to one of the image's axes [6].

B. Control Objectives

The goal of the active vision system is to maintain the image object in the middle of the 2D image plane, as well as within the image boundaries. In surveillance applications, in which wide areas have to be monitored, the camera's *Field of View* (FOV) must be automatically adjusted in order to cover a wider scene as possible. In such an active vision control system, only the adjustment of the pan and tilt of the camera is insufficient, since it cannot cope with objects that are too close or too far from the vision sensor. The problem can be solved by adding an extra *Degree of Freedom* (DoF) to the camera, that is, of the zoom control system, which aims at controlling the focal length through with the environment is sensed.

The errors which have to be compensated by the active vision system are illustrated in Fig. 2, where the position of a real world point, $pt_{int}(x, y, z)$ in the 3D Cartesian space, along with its 2D location in the image plane $pt_{proj}(x, y)$ is

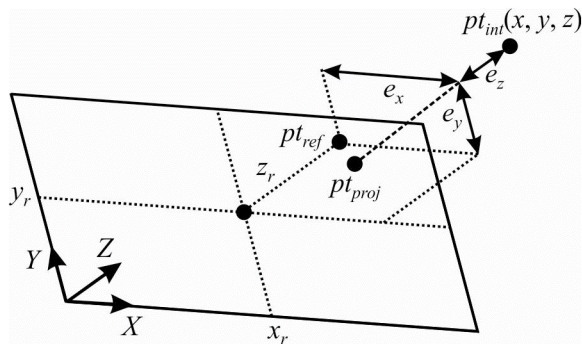


Fig. 2. The position error for a 3D point.

presented. $p_{ref}(x_r, y_r, z_r)$ represents the desired, or *reference*, coordinates of the object of interest in pixels. We consider x_r and y_r as the central 2D coordinates in the left image of the stereo camera system. The third coordinate, that is z_r , is the distance, or depth, from the middle point on the baseline between the two cameras and the imaged object. All the considered measurements are in pixel values. As will be explained in the next section, their conversion to real world meters is performed by the PTZ controllers.

Having in mind the above notations, a position error vector $e = [e_x, e_y, e_z]$ can be defined, where the errors are the differences between the position of the object of interest and $p_{ref}(x_r, y_r, z_r)$ along the three considered axes. The goal of the visual controller is to minimize the e_x and e_y errors by adjusting the pan and tilt values of the cameras and e_z by controlling the focal length of the cameras.

The proposed approach aims at imaging the full shape of the object in the middle of the FOV of the stereo camera system, while maintaining its boundaries within the left image. The object boundaries are maintained by the zoom controller, as will be further explained.

III. ACTIVE VISION SYSTEM MODELING

The goal of the proposed active vision system is to control the orientation and zoom of the stereo camera. Since a stereo camera is considered, the control system has to adapt the two PTZ cameras which make up the stereo vision robotic platform, thus ending up with a 6-DoF system (e.g. 2x pan, 2x tilt and 2x zoom). Since, in both cameras, the pan, tilt and zoom are modified using the same time of servo-drive, we consider the modeling and analysis of only a camera's DoF, that is, of the left camera's pan. The control of the other 5 DoF is analogous.

The control system, with the block diagram of a single DoF shown in Fig. 3, consists of two Sony-Evi-D70P[®] PTZ video cameras. In the diagram, the system is represented in the continuous time domain, being composed of a *Proportional* (P) controller $C(s) = k_c$, with the choice for k_c detailed in next sections, a system element $M(s) = k_m$,

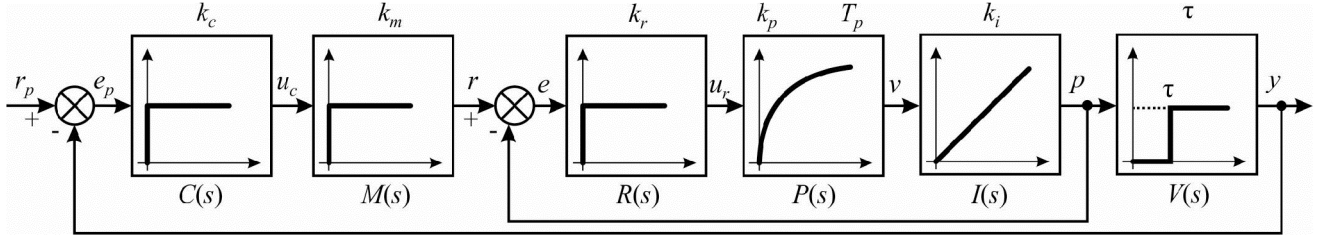


Fig. 3. Block diagram of the pan control system within the proposed active vision architecture.

used to convert the controlled signal u_c from the pixel camera metric to real 3D world coordinates. r_p is the reference position signal, for the pan case, being the central image plane coordinate.

The input signal e_p represents the object's 3D position error with respect to the camera coordinate's system, while r is the reference position, in real world coordinates. $R(s) = k_r$, is the controller transfer function of the inner position control loop. The process, or servo-drive, is modeled as a first order lag element $P(s) = k_p / (1 + T_p s)$.

Physically, the process is a *direct current* (DC) motor, used to adapt the camera's pan. The output v represents the camera velocity. Since the inner loop should control the pan position, an extra integrator element, $I(s)$, is added to integrate the position v over time. The signal e represents the error between the current pan orientation of the camera and the position along the x image axis of the object of interest. The goal of the closed-loop system is to maintain the x axis Cartesian error at zero. The last block shown in the diagram from Fig. 3 is represented by the image processing chain $V(s)$. Mathematically, the image processing system can only be expressed as a time-delay transfer function, written as $V(s) = e^{-s\tau}$ [3]. The delay τ represents the time needed to process a pair of images in order to extract the 3D pose of the object of interest.

Having in mind the above explanations, the transfer function of the inner-loop can be express as:

$$G_{il}(s) = \frac{R(s) \cdot P(s) \cdot I(s)}{1 + R(s) \cdot P(s) \cdot I(s)} = \frac{k_r k_p k_i}{s(s \cdot T_p + 1) + k_r k_p k_i}, \quad (1)$$

where the parameters values have been determined using the Matlab® identification toolbox. The identification process was needed because of the lack of an analytical process model. The model for the internal camera was chosen as in [4]. The transfer function parameters are estimated based on these two known parameters: the input reference signal r and the output camera position p . Using the reference position, considered as a step signal, we have evaluated the output camera pan positions. Using Eq. (1) as process model, the

obtained plant's transfer function has the gain product $k_r \cdot k_p \cdot k_i = 1.74$ and the time constant $T_p = 0.84$ sec.

From Eq. (1), the open-loop transfer function of the entire system can be written as:

$$G_{ol}(s) = C(s) \cdot M(s) \cdot G_{il}(s) \cdot V(s), \quad (2)$$

or:

$$G_{ol}(s) = k_c \cdot k_m \cdot G_{il}(s) \cdot e^{-s\tau}, \quad (3)$$

where $k_m = 1.37$. The numerical domain values for k_c are obtained using Theorem 1 (see Appendix).

Although the camera model, $G_{il}(s)$ and the $M(s)$ elements are linear, the time delay introduced by the image processing chain changes the system into a nonlinear one. The process modeled by $G_{ol}(s)$ is time-delay dependent, since it is always influenced by the processing time needed by the vision component.

IV. STABILITY ANALYSIS

A. Problem Formulation

To evaluate the stability of the proposed control system we have used the *Nyquist criterion* with the *gain margin* K_g and *phase margin* γ for the frequency response method [4]. The basic idea is to determine the behavior of the open-loop transfer function in the frequency domain, that is $G_{ol}(j\omega)$. Having in mind the Nyquist criterion, the locus of $G_{ol}(j\omega)$ represent a measure of the system's relative stability. The objective of this analysis it to determine the gain and phase margins in order to observe the system's stability reserve.

A system containing a time-delay τ has a stability which can be classified into two possible situations [10]:

- 1) *delay-independent stability*, if the locus of $G_{ol}(j\omega)$ is on the right side of the $(-1, +j0)$ point, for all positive and finite values of the delay, with $\tau \in [0, \infty)$;
- 2) *delay-dependent stability*, if the locus of $G_{ol}(j\omega)$ is on the right side of the point $(-1, +j0)$ for a finite number of delays $\tau \in [0, \tau_{\max})$, while for $\tau \in [\tau_{\max}, \infty)$ the locus of $G_{ol}(j\omega)$ is on the left side of the point

$$(-1, +j0).$$

The performance of a system which contains time delays, that is stable in closed-loop form $\tau = 0$, can be evaluated using the so-called *delay-margin*. The delay-margin can be defined as be the largest time-delay $\tau_{\max}$ for which the closed-loop system remains internally stable [2]. Hence, for any $\tau < \tau_{\max}$ a closed-loop system is stable. In the next section, we presented the computation of the maximum time-delay $\tau_{\max}$ which can be introduced by the machine vision component without affecting the stability of the pan control system.

B. Time-delay Stability Analysis

Through the stability analysis, that is delay dependent stability, of the system we aim to determine the value of $\tau_{\max}$ for which the system remains stable.

The analysis is performed in an open-loop loop manner on the transfer function $G_{ol}(s)$, without considering the time-delay and the system controller, $C(s)$. The system stability with $C(s)$ controller is presented in next section. The goal of the approach is to determine the open-loop stability and to observe the closed-loop system's evolution when a time-delay is introduced. For this purpose, the phase margin and its associated gain crossover frequency ω_c has to be determined [4]. When a time-delay exists within the system, it affects the phase margin and the phase crossover frequency.

From Eq. (1-3) and the model's parameters the delay free system has been represented as a second order element, with its frequency domain transfer function defined as:

$$G'_{ol}(j\omega) = \frac{2.38}{j\omega(0.84 \cdot j\omega + 1) + 1.74}. \quad (4)$$

The element from Eq. (4) is stable, since the gain margin and phase crossover frequency, ω_p , have infinite values [11]:

$$K_g \rightarrow \infty, \omega_p \rightarrow \infty, \quad (5)$$

and, according to the Nyquist stability criterion, the locus of (4) resides on the right side of the critical point $(-1 + j0)$.

The gain crossover frequency is determined using the following equation:

$$|G'_{ol}(j\omega_c)| = 1. \quad (6)$$

After solving the above equation, a value of $\omega_c = 1.93 \text{ rad/sec}$ is obtained. The phase margin for the time-delay free open-loop transfer function $G'_{ol}(j\omega_c)$ can be written as:

$$\begin{aligned} \gamma &= 180^\circ + \arg G'_{ol}(j\omega_c) = \\ &= 180^\circ - \arctg(0.48 \cdot \omega_c) - \arctg(0.57 \cdot \omega_c) = 53.84^\circ. \end{aligned} \quad (7)$$

The phase margin is the amount of additional phase lag, at the gain crossover frequency, which can bring the system to the stability limit [11]. Hence, the characteristic shows the open-loop system's stability reserves, before the insertion of the time-delay component. Since the time-delay component does not actually modify the gain crossover frequency, the time-delay phase margin can be computed as:

$$\gamma = 180^\circ + \arg G'_{ol}(j\omega_c) - \omega_c \cdot \tau, \quad (8)$$

where, τ represents the time-delay in seconds.

When the system is at stability limit, the time-delay phase margin is zero:

$$\gamma_{TD} = 180^\circ + \arg G'_{ol}(j\omega_c) - \omega_c \cdot \tau = 0. \quad (9)$$

Using the previous expression, we can evaluate the delay margin, that is, the maximum value of the time-delay after which the overall closed-loop system becomes unstable:

$$\tau_{\max} = \frac{[180^\circ + \arg G'_{ol}(j\omega_c)]_{rad}}{\omega_c} = \frac{\gamma}{\omega_c} = 0.48 \text{ sec}. \quad (10)$$

The obtained time-delay margin from Eq. (10) indicates that the image processing chain from Fig. 1 should process a pair of stereo images (e.g. image acquisition, segmentation, object recognition and 3D reconstruction) in less than 0.48 sec. Larger time-delay values will render the active vision system unstable.

C. Designing the system k_c controller

As before, the stability of the proposed active vision system has been analyzed in the frequency domain. The goal is to determine the delay margin, or the maximum time-delay, which renders the system unstable.

The basic idea here is to consider the transfer function of the open-loop system, containing the time-delay, and to find the $C(s)$ controller parameters which can stabilize the closed-loop process. In this case, the process is described by the following transfer function:

$$\begin{aligned} G_{ol}(s) &= \frac{2.38}{0.84 \cdot s^2 + s + 1.74} \cdot e^{-s\tau} = \\ &= \frac{2.83}{s^2 + 1.19 \cdot s + 2.07} \cdot e^{-s\tau}. \end{aligned} \quad (11)$$

where τ represent the time-delay in seconds. Having in mind [14], we propose a P controller, $C(s) = k_c$, which could stabilize the system for $\tau = \tau_{\max}$. The main goal of this method is to analytically determine the region in the k_c

parameter space, using Theorem 1 [14] (see Appendix), for which the closed-loop system remains stable.

Since the considered open-loop system from Eq. (5) is stable [11], using Theorem 1 we can determine the region for the k_c parameter which delivers a stable close-loop system. Firstly, the computation of parameter α is needed:

$$\alpha = \tau \sqrt{a_0 - (a_1^2/2)} = 0.88. \quad (12)$$

Further, the od and ev parameters need to be determined, as in (A.6) and (A.7) from Theorem 1. In the considered case they have the values $od = 1$ and $ev = 2$. The stable region of the control can be obtained by computing the roots z_l , for $l = 1, 2, 3, 4$, of equation:

$$\cot(z) = \frac{z^2 - 0.47}{0.57 \cdot z}, \quad (13)$$

where $z_1 = 0.9301$, $z_2 = 3.319$, $z_3 = 6.3734$ and $z_4 = 12.5663$. The z_l roots are further used to compute parameter A (see Appendix):

$$A = \frac{1.19 \cdot z_l}{2.83 \cdot 0.48 \cdot \sin(z_l)}, \quad (14)$$

for each z_l previously determined. Using relation (A.1):

$$\max_{l=ev, ev+2} \{A\} < k_c < \min_{l=od, od+2} \{A\}. \quad (15)$$

The stable values of k_c can be obtained:

$$-16.4754 < k_c < 1.0164. \quad (16)$$

In order to check the stability of the obtained interval of k_c parameter, we use a generalized form of the Hermite-Biehler Theorem, as presented in [1] and [14]. A random value of the controller's gain has been chosen in the determined interval, such as $k_c = 0.2$. The characteristic quasi-polynomial $\delta^*(s)$ of the system is given by [14]:

$$\delta^*(s) = 0.566 + (s^2 + 1.19 \cdot s + 2.07) \cdot e^{0.48 \cdot s}. \quad (17)$$

Substituting $s = j\omega$ we end up with:

$$\begin{aligned} \delta^*(j\omega) = & \\ = [0.56 + (2.07 - \omega^2) \cos(0.48\omega) - 1.19\omega \sin(0.48\omega)] + & \quad (18) \\ + j[(2.07 - \omega^2) \sin(0.48\omega) + 1.19\omega \cos(0.48\omega)]. \end{aligned}$$

In Fig. 4 the behavior of the real and the imaginary parts of $\delta^*(j\omega)$ is presented. As can be seen, the conditions of

Hermite-Biehler Theorem [14] are satisfied, since the roots of the real and imaginary parts interlace.

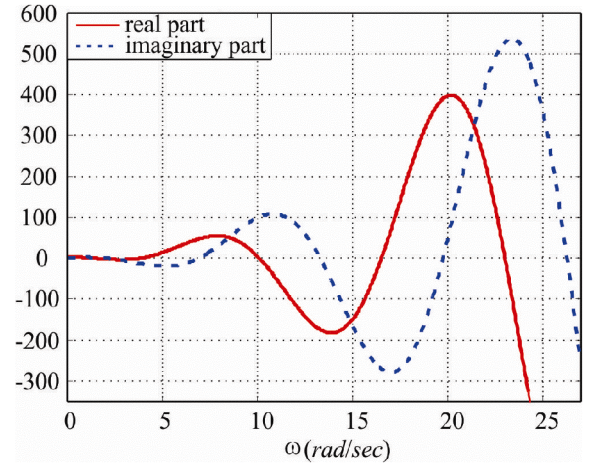


Fig. 4. Plot of the real and imaginary parts of $\delta^*(j\omega)$.

V. CONCLUSION

In this paper, the analysis of an active vision system was presented. The main goal was to determine the maximum value of the time delay component introduced by the image processing system (so called performance characteristic: delay margin) and to analytically calculate the parameters of the closed-loop controller for the considered system. The performance of the considered active vision model was studied using the frequency characteristics, phase margin and gain margin. Also, we use the generalized theorem of Hermite-Biehler to study the proposed closed-loop system. As future work, the authors consider the overall modeling of the visual understanding process, taking into account also the dynamics of the image processing algorithms..

APPENDIX

The Theorem 1 uses a constant gain feedback controller for stabilizing time-delay systems. Consider systems with a time delay that can be mathematically described by the transfer function:

$$G(s) = \frac{k}{s^2 + a_1 s + a_0} \cdot e^{-Ls}. \quad (A.1)$$

Theorem 1: The set of all stabilizing gains k_c for a given open-loop stable plant with a transfer function $G(s)$ as in (A.1) is given by:

1) If $a_1^2 \geq 2a_0$, then

$$-\frac{a_0}{k} < k_c < \frac{a_1 \cdot z_1}{k \cdot L \cdot \sin(z_1)}, \quad (A.2)$$

where, z_1 is the solution of equation

$$\cot(z) = \frac{z^2 - \tau^2 \cdot a_0}{\tau \cdot a_1 \cdot z}, \quad (A.3)$$

in the interval $(0, \pi)$.

2) If $a_1^2 < 2a_0$, then

$$\max_{l=ev, ev+2} \{A\} < k_c < \min_{l=od, od+2} \{A\}, \quad (A.4)$$

where, $A = \frac{a_1 \cdot z_l}{k \cdot \tau \cdot \sin(z_l)}$ and z_l is the solution of the equation:

$$\cot(z) = \frac{z^2 - L^2 \cdot a_0}{L \cdot a_1 \cdot z}, \quad (A.5)$$

in the interval $((l-1)\pi, l\pi)$; $\alpha = \tau \sqrt{a_0 - (a_1^2/2)}$; od represent an odd number, defined as:

$$od = \arg \min_{l=1,3,5,\dots} \{\alpha - z_l\}, \text{ subject to } \alpha - z_l \geq 0, \quad (A.6)$$

and ev represent an even number, defined as:

$$ev = \arg \min_{l=0,2,4,\dots} \{\alpha - z_l\}, \text{ subject to } \alpha - z_l \geq 0. \quad (A.7)$$

ACKNOWLEDGMENT

This paper is supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the contracts number POSDRU/88/1.5/S/59321 and POSDRU/89/1.5/S/59323.

REFERENCES

- [1] N. A. Almutairi, M. Y. Crow, Stabilization of Networked PI Control System Using Fuzzy Logic Modulation, In *Proc. of the American Control Conference*, Colorado, USA, 2003, pp. 975-980.
- [2] V. D. Blondel, A. Megretski, *Unsolved Problems in Mathematical Systems and Control Theory*, Princeton University Press, New Jersey, USA, 2004.
- [3] P.I. Corke, *Visual Control of Robots: A High-performance Visual Servoing*, Research Studies Press Ltd., John Wiley & Sons Inc, Great Britain, 1996.
- [4] R. C. Dorf, R. H. Bishop, *Modern Control Systems*, 12th ed., Prentice Hall, New Jersey, USA, 2010.
- [5] J. A. Fayman, O. Sudarsky, E. Rivlin, Michael Rudzsky, "Zoom Tracking and its Applications", In *Machine Vision and Applications*, Springer, vol. 13, no. 1, 2001, pp. 25-37.
- [6] A. Fusiello, E. Trucco, A. Verri, "A Compact Algorithm for Rectification of Stereo Pairs", In *Machine Vision and Applications*, vol. 12, no.1, 2000, pp. 16-22.
- [7] S.M., Grigorescu, "Robust Machine Vision for Service Robotics", Ph.D. Thesis, Faculty of Physics and Electrical Engineering, University of Bremen, Bremen, Germany, 2010.
- [8] R., Hartley, A., Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge Univ. Press, Cambridge, UK, 2000.

- [9] M. B. Khamesee, M. Rapolti, A. Amirfazli, "Controller Design and Optimization for Large-delays Image Processing in Visual Closed-loop Systems", In *Mechatronics*, Elsevier, vol. 18, 2008, pp. 251-261.
- [10] T. Mori, "Criteria for Asymptotical Stability of Linear Time-delay Systems", In *IEEE Trans Automatic Control*, vol. 30, 1985, pp. 158-160.
- [11] K. Ogata, *Modern Control Engineering*, 4th ed., Prentice Hall, New Jersey, USA, 2002.
- [12] H. Rotstein, E. Rivlin, "Control of a Camera for Active Vision: Foveated Vision, Smooth Tracking and Saccade", In *Proc. of the 1996 IEEE Int. Conf. on Control Applications*, Dearborn, USA, 1996, pp. 691-696.
- [13] R. Sharma, "Role of Active Vision in Optimizing Visual Feedback for Robot Control", In *Collection: The confluence of vision and control*, Springer, 1998.
- [14] G. J. Silva, A. Datta, S.P. Bhattachaiyya, *PID Controllers for Time-Delay Systems*, Birkhauser, Boston, USA, 2005.
- [15] V. Sundareswaran, P. Bouthemy, F. Chaumette, "Active camera self-orientation using dynamic image parameters", In *Proc. of the Third European Conference on Computer Vision*, Stockholm, Sweden, 1994, pp. 110-116.
- [16] K. Welke, T. Asfour, Rudiger Dillmann, "Active Multi-View Object Search on a Humanoid Head", In *IEEE Inter. Conf. on Robotics and Automation (ICRA)*, Kobe, Japan, 2009, pp. 417-423.
- [17] Y. Yao, B. Abidi, M. Abidi, "3D Target Scale Estimation and Target Feature Separation for Size Preserving Tracking in PTZ Video", In *Int. Journal of Computer Vision*, Springer, vol. 82, 2009, pp. 244-263.

Hexapod Robot. Mathematical Support for Modeling and Control

Mănoiu – Olaru Sorin, Nițulescu Mircea, *Member, IEEE*, Stoian Viorel

Abstract—In this paper the authors present a complete kinematical models of a hexapod robot's leg. There is also described a dynamical model of the selected leg. First, the complete kinematical and dynamical models of the leg are included. Then, a hexapod robot structure using this leg type is generally presented. Finally, simulation results are attached to evaluate the performances of the proposed model. More relevant experimental results can be seen in the article: *Hexapod Robot. Virtual models for preliminary studies*.

I. INTRODUCTION

MOST of the earth's surface is inaccessible to regular vehicles [1]. Today's robots are mostly designed for traveling over relatively smooth, level or inclined surfaces.

The terrain in question is either outdoor environments, that is generally considered difficult for mobile robots, or indoor environments like staircases, doorsteps or tight corners can cause difficulties.

An important drawback of legged machines is the complexity of the control required to achieve walking even on completely flat and horizontal surface in which much simpler wheeled machines work perfectly well [2].

Research on legged vehicles has been going on for over a hundred years. The reason for this persistence is that only half of the earth's surface is accessible to wheeled and tracked while legged animals can be found almost anywhere (from the top of the mountain to the sea).

A lot of research has been put into legged machines and several successful applications are available but most of them have never been outside a laboratory [3],[9].

The main difference between walking machine and wheeled machines is ground contact which is not continuous; instead they use isolated footholds by coordinating reposition of their legs.

Many scientists use animal's similarities in their mechanical design for leg configuration and leg design leading often to simplifications of the actual biological

system. Walking machines are classified depending on the number of the legs used. The most common are: octopods, hexapod, quadrupeds and bipeds.

The number of the legs of a robotic structure affects stability (more legs better stability, Fig 1) and the workspace. There are two main types of legs configurations that are biological inspired. The main difference is in the way the leg swing relative to body.

The first type is similar to cats, humans and birds where the leg swings around a horizontal axis.

The second type is more similar to insects where the legs swing around a vertical axis providing better stability.

Most popular hexapods can be grouped into two categories: rectangular (with two groups of three legs distributed symmetrically on the two sides) and hexagonal (round or hexagonal body with evenly distributed legs)[1].

The motion of legged robots can be divided into statically and dynamically stable. Static stability means that the robot is stable at all times during its gait cycle. Dynamic stability means that the robot is only stable when it is moving. For legged robots, static stability demands that the robot has at least three legs on the ground at all times and the robot's centre of mass is inside the support polygon, i.e. the convex polygon formed by the feet supporting the robot (Fig.1).

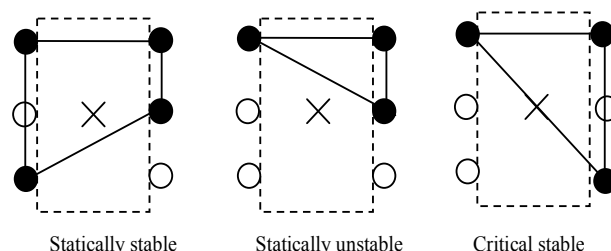


Fig. 1. Stability cases for a hexapod robot: statically stable, unstable cases and critically stability

On the left side four legs provide support and the centre of mass is located inside the support polygon so the robot is statically stable. On the middle the bottom left leg has been lifted, putting the centre of mass outside the support polygon which made the robot unstable. On the right side three legs provide support and the centre of mass is located on one side of the support polygon. This case is called critical stability.

Some of the most important advantages of legged locomotion are [4]:

- Accommodation to uneven terrain.
- Use of isolated footholds.
- Providing active suspension.

--Environmental effects of legged vehicles are less than wheeled or tracked vehicles.

Manuscript received June 17, 2011. This work was supported in part by the strategic grant POSDRU/88/1.5/S/50783, Project ID 50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resource Development 2007 – 2013

Mănoiu Olaru Sorin is with the University of Craiova, Faculty of Automation, Computers and Electronics, Bulevardul Decebal, Nr. 107, 200440, Craiova, ROMANIA Tel./Fax: +40.251.436999 (e-mail: manoiusorin2006@yahoo.com).

Mircea Nițulescu is with the University of Craiova, Faculty of Automation, Computers and Electronics, Bulevardul Decebal, Nr. 107, 200440, Craiova, ROMANIA Tel./Fax: +40.251.436999 (e-mail: nitulescu@robotics.ucv.ro).

Stoian Viorel is with the University of Craiova, Faculty of Automation, Computers and Electronics, Bulevardul Decebal, Nr. 107, 200440, Craiova, ROMANIA Tel./Fax: +40.251.436999 (e-mail: stoian@robotics.ucv.ro).

Among disadvantages of legged locomotion we can enumerate:

- Artificial walking mechanisms are so far heavy due to large number of actuators.
- Control of walking is very complex and so far walking vehicles are rather slow.
- Bad payload-weight-to-mechanism-weight ratio compared to wheeled or tracked vehicles.
- Appearance of an impact force with each step made.

II. HEXAPOD ROBOT LEG

The successful design of a legged robot depends mostly on the design of the chosen leg. Since all aspects of walking are ultimately governed by the physical limitations of the leg, it is important to select a leg that will allow a maximum range of motion and that will not impose unnecessary constraints on the walking.

A. Direct Kinematics

A three-revolute kinematical chain has been chosen for each leg mechanism in order to mimic the leg structure (Fig. 2). A direct geometrical model for each leg mechanism is formulated between the moving frame $O_i(x_i, y_i, z_i)$ of the leg base, where $i=1\dots3$, and the fixed frame $O_W(X_W, Y_W, Z_W)$.

The coordinate frames for the robot legs are assigned as in fig. 2. The assignment of link frames follows the Denavit-Hartenberg direct geometrical modelling algorithm. The robot leg is made of links and joints as noted in fig. 2. The different links of the robot legs are called *coxa*, *femur* and *tibia*.

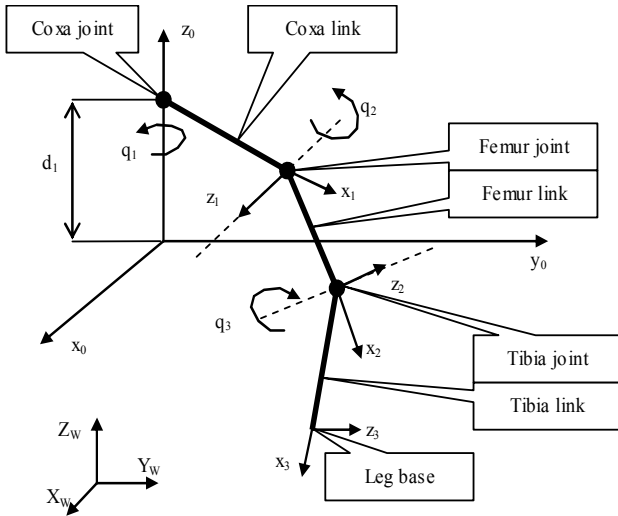


Fig. 2. Model and coordinate frame for leg kinematics.

The robot leg frame starts with link 0 which is the point on the robot where the leg is attached; link 1 is the coxa, link 2 is the femur and link 3 is the tibia. Legs are distributed symmetrically about an axis in the direction of motion (Y in this case). The general form for the transformation matrix from link i to link $i-1$ using Denavit Hartengberg parameters [5] is given in (1):

$$T_i^{i-1} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

The transformation matrix is a series of transformations:

1. Translate d_i along z_{i-1} axis,
2. Rotate θ_i about z_{i-1} axis,
3. Translate a_i along x_{i-1} axis,
4. Rotate α_i about x_{i-1} axis.

The overall transformation is obtained as a product between three transformation matrixes:

$$T_{base}^{coxa} = T_{femur}^{coxa} T_{tibia}^{femur} T_{base}^{tibia} \quad (2)$$

where (considering Fig. 3):

$$T_{femur}^{coxa} = \begin{bmatrix} \cos \theta_1 & 0 & \sin \theta_1 & L_1 \cos \theta_1 \\ \sin \theta_1 & 0 & -\cos \theta_1 & L_1 \sin \theta_1 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$T_{tibia}^{femur} = \begin{bmatrix} \cos \theta_2 & \sin \theta_2 & 0 & L_2 \cos \theta_2 \\ \sin \theta_2 & -\cos \theta_2 & 0 & L_2 \sin \theta_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$T_{base}^{tibia} = \begin{bmatrix} \cos \theta_3 & -\sin \theta_2 & 0 & L_3 \cos \theta_2 \\ \sin \theta_3 & \cos \theta_3 & 0 & L_3 \sin \theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Using (2) and considering (3)-(5) the coordinates of the leg tip are:

$$\begin{aligned} x &= \cos \theta_1 (L_1 + L_2 \cos \theta_2 + L_3 \cos(\theta_2 - \theta_3)) \\ y &= \sin \theta_1 (L_1 + L_2 \cos \theta_2 + L_3 \cos(\theta_2 - \theta_3)) \\ z &= d_1 + L_2 \sin \theta_2 + L_3 \sin(\theta_2 - \theta_3) \end{aligned} \quad (6)$$

where:

d_1 is the distance from the ground to coxa joint.

L_i are the lengths of the leg links.

The centre of mass of each link is positioned relative to the link frame by a position vector $p_i = [x_i, y_i, z_i, 1]^T$. To find the position of the centre of mass of each link relative to leg frame, the coordinates p_i are multiplied with the D-H transformation giving the centre of mass positions:

$$p_{CoM_i} = T_0^i p_i, i = 1 \dots 3 \quad (7)$$

The position of centre of mass of the leg is calculated using the following equations:

$$x_g = \frac{\sum_{i=1}^3 x_i m_{li}}{\sum_{i=1}^3 m_{li}}, y_g = \frac{\sum_{i=1}^3 y_i m_{li}}{\sum_{i=1}^3 m_{li}}, z_g = \frac{\sum_{i=1}^3 z_i m_{li}}{\sum_{i=1}^3 m_{li}} \quad (8)$$

where m_{li} is the mass of link i

B. Inverse Kinematics

The geometrical model described above establishes a connection between the joint variable and the position and orientation of the end frame. The inverse kinematics problem consists of determining the joint angles from a given position and orientation of the end frame. The solution of this problem is important in order to transform the motion assigned to the end frame into the joint angle motions corresponding to the desired end frame motion.

The goal is to find the three joint variables θ_1 , θ_2 , and θ_3 corresponding to the desired end frame position. The end frames orientation is not an issue, since we are only interested in its position.

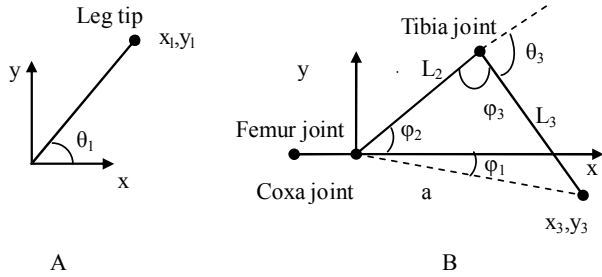


Fig. 3. Illustrations for solving inverse kinematics.

Using (6) and considering the following constraints: all joints allow rotation only about one axis, femur and tibia always rotate on parallel axes, and the physical limitation of each joint we can determine the joint angle.

The coxa joint angle can be found using $\text{atan2}(y, x)$ function as can be seen from fig.3.A.

$$\theta_1 = a \tan 2(y_1, x_1) \quad (9)$$

In order to determine the other two angles a geometrical approach was considered. To further simplify the approach the leg tip coordinates were transformed to coxa frame using the transformation matrix below:

$$T_{coxa}^{femur} = \begin{pmatrix} (R_{femur}^{coxa})^T & -(R_{femur}^{coxa})^T * d_{femur}^{coxa} \\ 0 & 1 \end{pmatrix} \quad (10)$$

The angle φ_2 which is the angle relating to the femur servo position, can be derived directly from the triangle:

$$\theta_2 = \varphi_2 \quad (11)$$

The angle φ_1 is the angle between the x-axis and line a

and can be calculated with atan2 function:

$$\varphi_1 = a \tan 2(y_3, x_3) \quad (12)$$

where x_3 and y_3 are the leg tip coordinates in coxa frame.

If we consider φ_1 to be the entire femur span and apply the law of cosines results:

$$\varphi_1 = \arccos\left(\frac{L_2^2 + a^2 - L_3^2}{2L_2a}\right) \quad (13)$$

where:

$$a = \sqrt{x_3^2 + y_3^2} \quad (14)$$

Next the femur angle can be found from:

$$\theta_2 = \arccos\left(\frac{L_2^2 + a^2 - L_3^2}{2L_2a}\right) + a \tan 2(y_3, x_3) \quad (15)$$

Again, applying the law of cosines we find the φ_3 angle:

$$\varphi_3 = \arccos\left(\frac{L_2^2 + L_3^2 - a^2}{2L_2L_3}\right) \quad (16)$$

Considering fig. 3B we see that θ_3 can be found as follows:

$$\theta_3 = \pi - \varphi_3 \quad (17)$$

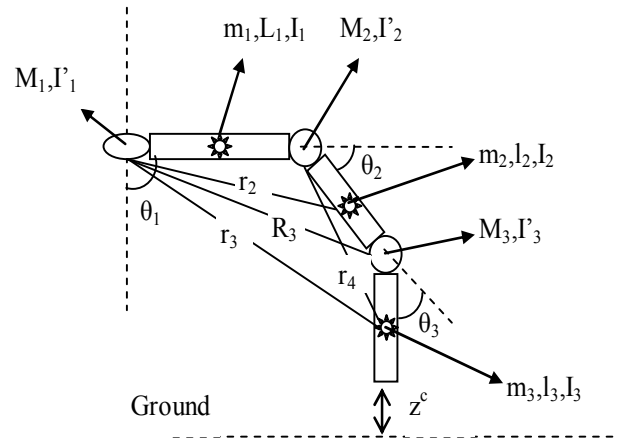


Fig. 4. Illustration for developing dynamic model of the leg.

C. Dynamic model

The purpose of robot dynamics is to determine the generalized forces required to not only overcome the self-inertial loads (due to the weight of the links inside the robot) but also to produce the desired input motions to the robot's mechanism [7].

In order to obtain a more precise model we divided the mass of each link in two (M_i - servomotor mass, m_i -link mass, $M_i > m_i$) (Fig. 4).

Considering the generalized coordinates vector $q = [q_1, q_2, q_3]^T$ the generalized vector forces can be computed using the below equation:

$$\tau_i = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} \quad (18)$$

where: $L(q, \dot{q}) = E_c(q, \dot{q}) - E_p(q)$.

Considering that all three servomotors have the same mass $M_1=M_2=M_3=M$ and the last two links have the same mass $m_2=m_3=m_2$ but different lengths the expressions for generalized forces are:

$$\tau_1 = \ddot{\theta}_1 (I'_1 + I''_1 + M(I_1^2 + R_3^2) + m_2(r_2^2 + r_3^2)) \quad (19)$$

$$\tau_2 = \ddot{\theta}_2 (I'_2 + I''_2 + Ml_2^2 + m_3r_4^2) - \{g[l_3 \cos(\theta_2 + \theta_3) * (3M + m_1 + \frac{3m_2}{2}) + l_2 \cos(\theta_2)(2M + m_1 + \frac{m_2}{2})]\} \quad (20)$$

$$\tau_3 = \ddot{\theta}_3 (I'_3 + I''_3) - gl_3 \cos(\theta_2 + \theta_3)(3M + m_1 + \frac{3m_2}{2}) \quad (21)$$

where:

I'_i -are the moments of inertia associated with the servomotors;

I''_i -are the moments of inertia associated with the links;

r_i -radius of instantaneous circle of rotation of the center of mass associated with the link i of the leg, $i=2...4$;

R_3 -radius of instantaneous circle of rotation of the servomotor 3.

III. WORKSPACE ANALYSIS OF THE LEG

There are several things that are important when defining the leg workspace. A basic one is that the workspace should not be larger than the reach of the leg. Workspace must be defined in order to maximize motion but in the same time to minimize singularities or any other pitfalls [8].

Workspace is also useful for walking algorithm to prevent collision of adjacent legs by imposing restrictions for direct or inverse kinematics.

The actuators used on the legs are capable of rotating 180 degrees. In order to minimize leg collisions the coxa angle was limited between $-\pi/4$ and $\pi/4$. This restriction was introduced due to mechanical construction of the leg. To get the most out of motion the operation space of variable θ_2 was set from $-\pi/2$ to $\pi/4$ and θ_3 from 0 to $3\pi/4$. In our case there is a singularity when the x and y coordinates of the leg tip are zero. In this configuration the value for θ_1 is

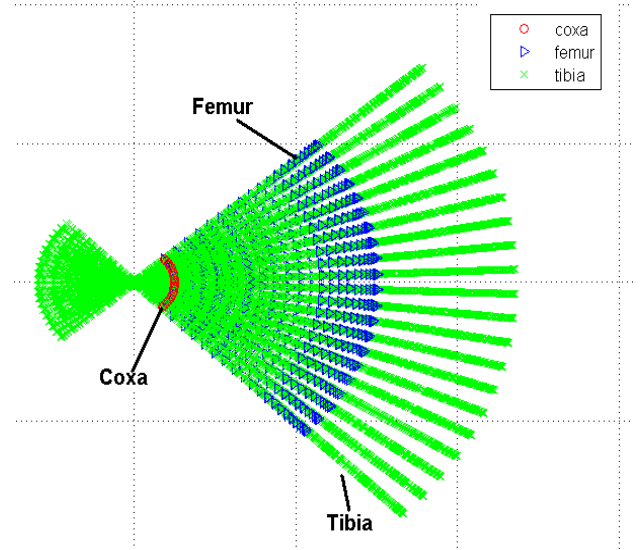


Fig. 5. Leg workspace. XY view.

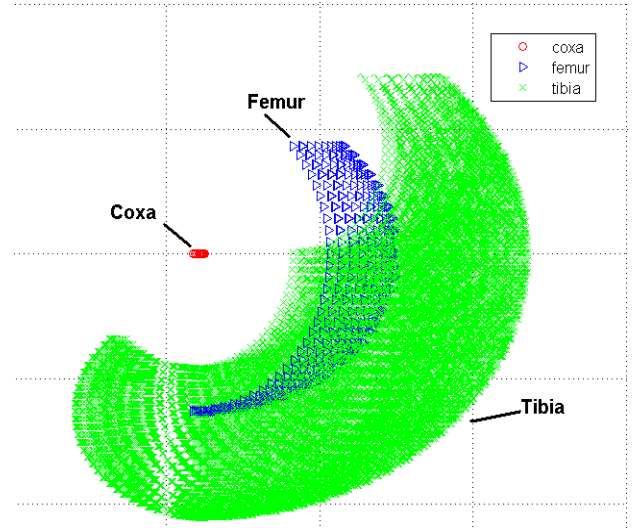


Fig. 6. Leg workspace. XZ view.

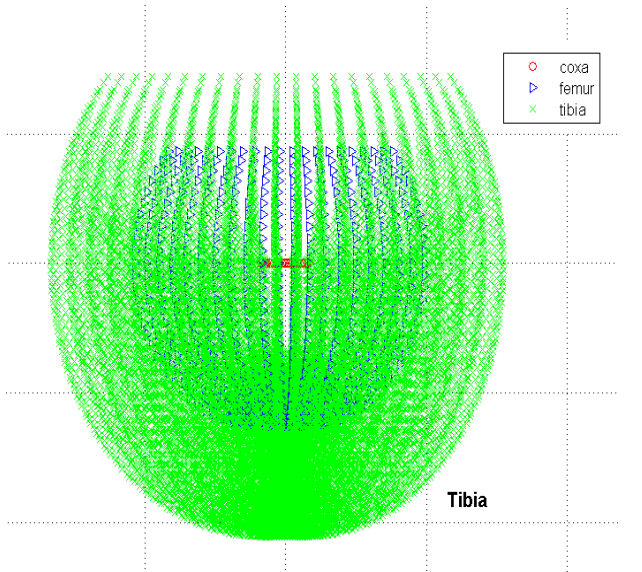


Fig. 7. Leg workspace. YZ view.

arbitrary. To avoid this scenario we can define the x coordinate greater than zero.

Physical limitations due to mechanical construction for each joint angle are:

$$q_{coxa} = [-\pi/4, \pi/4],$$

$$q_{femur} = [-\pi/4, \pi/2],$$

$$q_{tibia} = [0, 3\pi/4].$$

The defined workspace with the above limitation is presented in fig. 5, 6 and 7.

The leg workspace with the important point is presented in fig. 8 and synthesized in table 1.

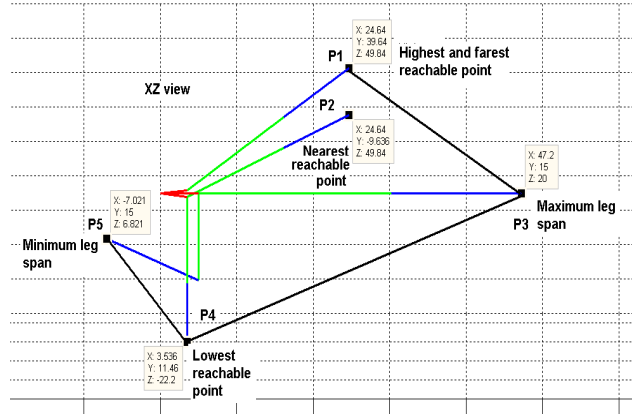


Fig. 8. Leg workspace. Extreme interest points.

TABLE I
LEG WORKSPACE

Coordinate	Min	Max
x	-7	47,2
y	-9,6	39,6
z	-22,2	49,8

Valid for $l_1=5$, $l_2=25.2$, $l_3=17$ and for coxa joint coordinates (0, 15, 20)

IV. HEXAPOD ROBOT MODEL

The legged locomotion on natural terrain presents a set of complex problems (foot placement, obstacle avoidance, load distribution, general stability) [6] that must be taken into account both in mechanical construction of vehicles and in development of control strategies.

The robot structure considered has 6 identical legs and each leg has 3 degree of freedom (RRR) (Fig. 9). All the relevant points have been put on the model as can be seen from figure 9: coordinates of the centre of mass of each leg G_i , $i=1...6$; leg numbering for easy understanding (1 to 6), coordinates of the centre of mass of the robot G , projection of the centre of mass into the support polygon G' , robot's centre of symmetry with the attached frame $O_R(X_R, Y_R, Z_R)$, the global frame $O_w(X_w, Y_w, Z_w)$.

The global frame is the frame that all other frames will be defined relative to. The global frame is rigidly attached to the lower left corner of the world so that the z-axis is vertical and the xy-plane is aligned with the floor surface.

The origin of the robot coordinates is attached in the centre of symmetry with the z-axis pointing up, the x-axis

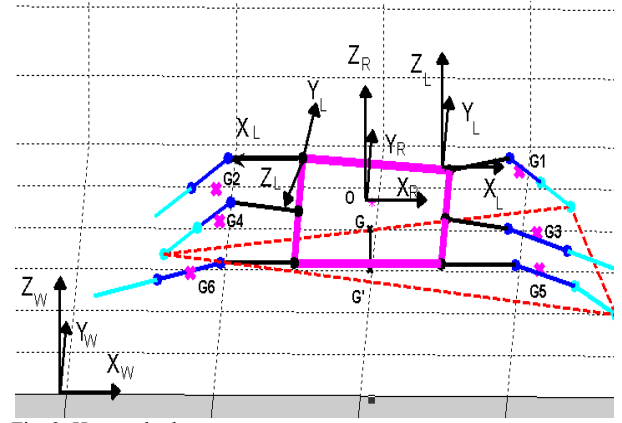


Fig. 9. Hexapod robot structure

pointing left and the y-axis pointing forward.

The overall transformation from global frame to each leg tip is obtained as follows:

$$T_{tip_i}^W = T_R^W T_{coxa_i}^R T_{tip_i}^{coxa_i} \quad i=1...6 \quad (22)$$

where:

T_R^W is the transformation matrix between the global frame and the robot frame and defines the rotation about y,x,z (roll, pitch, yaw matrix).

$T_{coxa_i}^R$ is the transformation matrix from the robot centre to each leg.

The position of robot's centre of mass is calculated using the following equations:

$$X_G = \frac{\sum_{i=1}^6 x_{gi} m_{Li}}{\sum_{i=1}^6 m_{Li}}, Y_G = \frac{\sum_{i=1}^6 y_{gi} m_{Li}}{\sum_{i=1}^6 m_{Li}}, Z_G = \frac{\sum_{i=1}^6 z_{gi} m_{Li}}{\sum_{i=1}^6 m_{Li}} \quad (23)$$

where:

$$m_{Li} = \sum_{i=1}^3 m_{li}, \quad m_{li} - \text{mass of each leg}$$

V. CONCLUSION

In this paper were presented fundamental theoretical aspects regarding direct kinematical model, inverse kinematical model and direct dynamical model for a mechanical structure of a leg in order to build a hexapod robot. All models were based on the information about the leg chosen.

For the same leg structure a workspace analysis has been made in order to highlight the restrictions when developing the walking algorithms of the robot.

The paper includes also an overview of the coordinate frames attached to all the robot components in order to calculate the position of the center of mass. This point is

important for stability analysis for different walking algorithms and space structure.

ACKNOWLEDGMENT

This work was partially supported by strategic grant POSDRU/88/1.5/S/50783, Project ID 50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resource Development 2007 – 2013.

REFERENCES

- [1] Xilun Ding, Zhiying Wang, A. Rovetta, J. M. Zhu, "Locomotion analysis of hexapod robot", "Climbing and Walking Robots", March 2010, ISBN: 978-953-307-030-8 pp.291-310 India.
- [2] G. Carbone, M. Ceccarelli, "Legged robotic Systems", "Cutting Edge Robotics", ARS Scientific Book, pp. 553-576, Wien.
- [3] G. Figliolini, P. Lea, "Mechanics and simulation of six-legged walking robots", "Climbing & Walking Robots, Towards New Applications", ISBN: 978-3-902613-16-5, pp.546, October 2007 Austria.
- [4] M. Nitulescu, "Robotic systems with navigation capabilities", Universitaria Craiova, ISBN: 973-8043-143-3, Romania 2002.
- [5] R. J. Schilling, "Fundamentals of robotics: analysis and control", ISBN: 0-13-344433-3, Prentice Hall, New Jersey, USA, 1990.
- [6] W. Krzysztof, B. Dominik, K. Andrzej, "Control and Environment sensing system for a six-legged robot", "Journal of Automation, Mobile Robotics & Intelligent Systems", Vol 2 N°3 2008.
- [7] F. Fahimi Autonomous robots: modeling, path planning and control, Springer 2008.
- [8] K. H. Maki, "Bioinspiration and robotics: walking and climbing robots", ISBN: 978-3-902613-15-8, I-Tech Education and Publishing, Croatia 2007.
- [9] G. M. Nelson, R. D. Quinn, R. J. Bachmann, W. C. Flannigan, "Design and simulation of a cockroach-like hexapod robot", "Proceedings of the 1997 IEEE International Conference on Robotics and Automation", Albuquerque, April 1997, New Mexico.
- [10] S. Sarjoun, G. Kantor, G. Maiwand, A. Alfred, "Inertial navigation and line following for a dynamical hexapod robot", "Proceedings of the 2003 IEEE/RSJ Int. Conf on Intelligent Robots and Systems", Las Vegas, Nevada 2003.

Hexapod Robot. Virtual Models for Preliminary Studies

Mănoiu – Olaru Sorin, Mircea Nițulescu, *Member, IEEE*

Abstract— In this paper the authors presents a software platform for controlling a virtual hexapod robot made using Matlab. The interface allows full control of the hexapod robot and single leg control. The main purpose of the interface is to simulate hexapod robot stability in gravitational field for a certain configuration of legs. The model of the robot and the leg were developed using geometrical modeling based on Denavit-Hartenberg algorithm. The analysis of robot stability is made for different stages of locomotion on horizontal surface and for different leg configurations.

The leg control part is an experimental one and it is a combination between Matlab and Arduino Duemilanove development board. In order to control the robotic leg the direct kinematics model and inverse kinematics model were used. The application consists of moving the leg tip along a predefined trajectory using a walking algorithm. Moving the leg tip from one point to another is made in two steps: stance phase and swing phase.

The paper includes some experimental results related to the static gravitational stability depending on the support polygon formed by the legs on the ground and for single leg control.

I. INTRODUCTION

IN order to traverse a given terrain the robot needs to be equipped with some sort of locomotion system whether it is wheels, legs or articulated bodies (snakes).

Different types of fields require different types of locomotion, wheels and tracks are preferred on even terrain but on uneven terrain the advantages of wheels became minimal or obsolete. Instead the advantages of legs make walking more useful in an uneven terrain because of the positioning the leg with high precision.

The significant drawback to legged locomotion is the difficulty in control. The difficulty is not moving the individual legs, but in coordination of the movement of the legs and the body. The legged robot control system must generate a sequence of leg and body motions, a gait, which will propel it along desired trajectory.

Gait generation is the formulation and selection of a sequence of coordinated leg and body motions that propel the robot along the desired path.

Manuscript received June 17, 2011. This work was supported in part by the strategic grant POSDRU/88/1.5/S/50783, Project ID 50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resource Development 2007 – 2013

Mănoiu Olaru Sorin is with the University of Craiova, Faculty of Automation, Computers and Electronics, Bulevardul Decebal, Nr. 107, 200440, Craiova, ROMANIA Tel./Fax: +40.251.436999 (e-mail: manoiusorin2006@yahoo.com).

Mircea Nițulescu is with the University of Craiova, Faculty of Automation, Computers and Electronics, Bulevardul Decebal, Nr. 107, 200440, Craiova, ROMANIA Tel./Fax: +40.251.436999 (e-mail: nitulescu@robotics.ucv.ro).

The most studied problem for multi-legged robots concerns how to determine the best sequence for lifting off and placing the feet. Hexapod gaits have been widely studied as a function of robot characteristics [1].

The large diversity of the existing walking animals offers innumerable examples of the possibilities of this type of locomotion.

The gait analysis and selection requires an appreciable modeling effort for the improvement of mobility with legs in structure/unstructured environments.

Nowadays studies are focused primarily on using artificial neural networks, fuzzy logic or central pattern generators for both leg coordination and leg control.

Arduino Duemilanove is a development board equipped with an AVR ATMEGA328 microcontroller [2]. Today all the microcontrollers are made with CMOS technology because of the large density of integration at a lower cost

AVR uses Harvard architecture, with separate memories and buses for program and data.

Advantages of using Arduino Duemilanove are:

- inexpensive. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than \$50.

- easy to program. Programming environment is easy-to-use for both beginners and advanced programmers

- open source. Lots of code libraries are available for free for a wide range of external components (sensors, actuator, LCD).

II. ROBOTIC LEG MODEL

The performance of a legged robot depends mostly on the design choose for the leg. So it is important to select a mechanical design of a leg that will maximize motion and impose fewer constraints on walking.

A three-revolute kinematical chain has been chosen for each leg mechanism in order to mimic the leg structure (Fig. 1).

The assignment of link frames follows the Denavit-Hartenberg direct geometrical modelling algorithm. The robot leg is made of links and joints as noted in [3].

The coordinates of leg tip considering [3] are:

$$\begin{aligned}x &= \cos \theta_1 (L_1 + L_2 \cos \theta_2 + L_3 \cos(\theta_2 - \theta_3)) \\y &= \sin \theta_1 (L_1 + L_2 \cos \theta_2 + L_3 \cos(\theta_2 - \theta_3)) \\z &= d_1 + L_2 \sin \theta_2 + L_3 \sin(\theta_2 - \theta_3)\end{aligned}\quad (1)$$

The position of the centre of mass of the leg can be found using Denavit – Hartenberg transformation [4]:

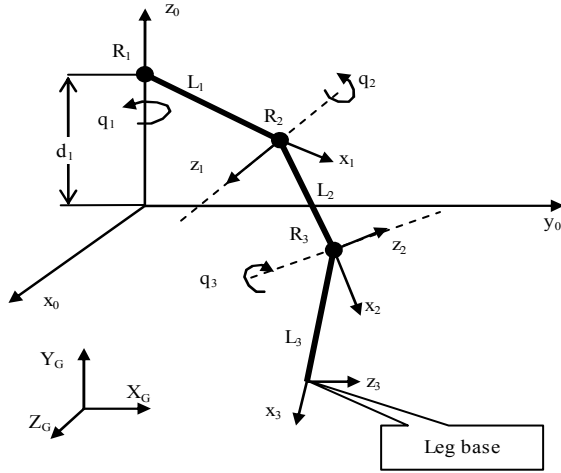


Fig. 1. Model and coordinates frame for leg kinematics.

$$p_{CoM_i} = T_0^i p_i, i = 1 \dots 3 \quad (2)$$

The position of centre of mass of the leg is calculated using the following equations:

$$x_g = \frac{\sum_{i=1}^3 x_i m_{li}}{\sum_{i=1}^3 m_{li}}, y_g = \frac{\sum_{i=1}^3 y_i m_{li}}{\sum_{i=1}^3 m_{li}}, z_g = \frac{\sum_{i=1}^3 z_i m_{li}}{\sum_{i=1}^3 m_{li}} \quad (3)$$

where: m_{li} is the mass of link i .

In order to control the robotic leg along a given trajectory we will use the inverse kinematics. The joints angles depending on the leg tip coordinates can be found using the following equations:

$$\theta_1 = \begin{cases} a \tan 2(y, x) + 180, & x < 0 \\ a \tan 2(y, x), & \text{otherwise} \end{cases} \quad (4)$$

$$\theta_2 = a \cos \left(\frac{l_2^2 + \left(\sqrt{(x'^2 + y'^2)} \right)^2 - l_3^2}{2l_2 \sqrt{(x'^2 + y'^2)}} \right) + a \tan 2(y', x') \quad (5)$$

$$\theta_3 = \pi - a \cos \left(\frac{\left(\sqrt{(x'^2 + y'^2)} \right)^2 - l_3^2 - l_2^2}{-2l_2 l_3} \right) \quad (6)$$

where:

l_1, l_2, l_3 - are the links lengths

x, y - the coordinates of the leg tip in the leg frame

x', y' - the coordinates of the leg tip in coxa frame.

III. HEXAPOD ROBOT MODEL

The legged locomotion on natural terrain presents a set of complex problems (foot placement, obstacle avoidance, load distribution, general stability) [5] that must be taken into

account both in mechanical construction of vehicles and in development of control strategies.

One way to handle these issues is using models that mathematically describe the different situations. Therefore modelling becomes a useful tool in understanding systems complexity and for testing and simulating different control approaches [6], [7].

The robot structure considered (Fig. 2) has 6 identical legs. All the relevant points have been put on the model as can be seen from fig. 2 [3].

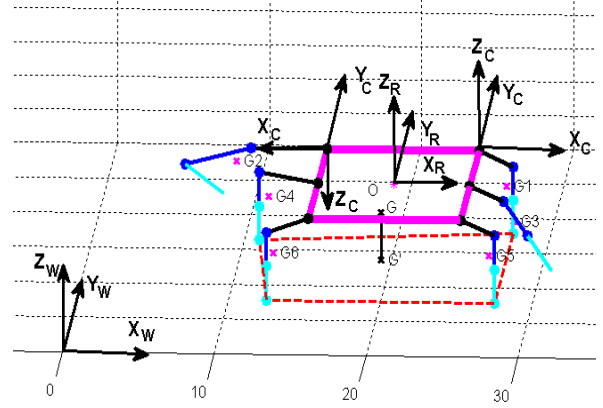


Fig. 2. Hexapod robot mode and coordinates frame.

The overall transformation from global frame to each leg tip is obtained as follows:

$$T_{tip_i}^W = T_R^W T_{coxa_i}^R T_{tip}^{coxa_i} \quad i=1 \dots 6 \quad (7)$$

where: T_R^W is the transformation matrix between the global frame and the robot frame and defines the rotation about y, x, z (roll, pitch, yaw matrix).

$T_{coxa_i}^R$ is the transformation matrix from the robot centre to each leg.

The position of robot's centre of mass is calculated using the following equations:

$$X_G = \frac{\sum_{i=1}^6 x_{gi} m_{Li}}{\sum_{i=1}^6 m_{Li}}, Y_G = \frac{\sum_{i=1}^6 y_{gi} m_{Li}}{\sum_{i=1}^6 m_{Li}}, Z_G = \frac{\sum_{i=1}^6 z_{gi} m_{Li}}{\sum_{i=1}^6 m_{Li}} \quad (8)$$

where:

$$m_{Li} = \sum_{i=1}^3 m_{li}, m_{li} - \text{mass of each leg}$$

IV. SIMULATION PLATFORM

The main purpose of this simulation platform is to show how the support polygon modifies when legs lose contact with ground and if the projection of the centre of mass is within the support polygon. The simulation program was made using MatLab GUIDE [8].

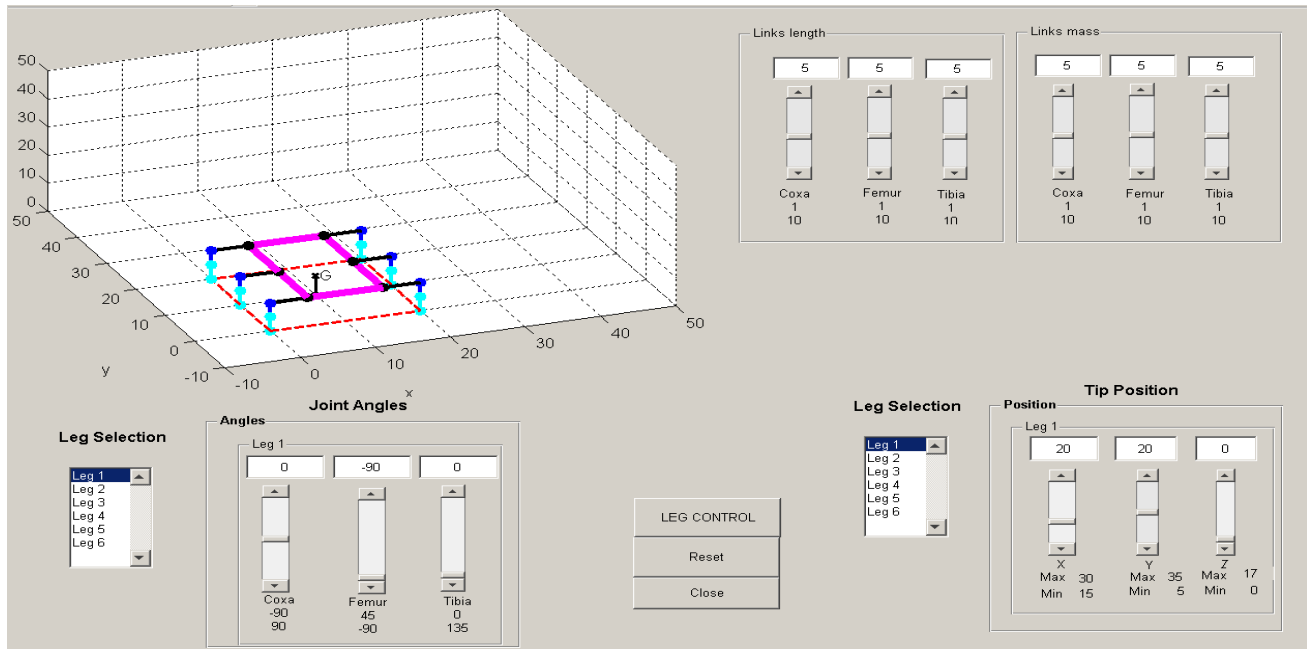


Fig. 3. Hexapod robot control interface.

The developed software platform can be used to analyze what happens with the hexapod robot in gravitational field and it also allows communication with the leg in real world.

The analysis of the hexapod robot in gravitational field can be group in two modes:

--Free fall mode. In this mode the mechanical configuration of the legs is defined by their joints values, no additional move is allowed.

--Transitory analysis. This mode is used to analysis what happens with the robot between two static regimes.

The communication between the physical leg and Matlab environment is made using Arduino Duemilanove development board.

Giving a set of values for legs joints yields a stationary mechanical configuration which generates a certain support polygon in relation with which we analyze the gravitational stability of the hexapod robot.

For a certain configuration of legs, the robot is statically stable if the projection of the centre of mass is inside the support polygon; it is at stability limit if the projection of the centre of mass is on one side of the support polygon and it is statically unstable if the projection of the centre of mass is outside the support polygon. In this last case the robot is shifting gradually its support polygon by lifting/touching the ground with its legs, due to gravity, until the condition for static stability is accomplished.

The shape of the support polygon needed for minimum static stability is the triangle.

A. Program Interface

When the interface is launched the robot is first drawn in a stable configuration as shown in fig. 3. The interface is divided basically into 2 areas: in the upper left the robot is displayed (plotted) according to the values set by the user and in the upper right and bottom we can find the controls for the robot. The controls for the robot are structured

mainly in 2 parts: joints control, position control. Joints control and position control consists of a list where the leg number it put on and a panel where the user can set the values for each joint or position.

The controls for each leg are encapsulated into a panel identified by leg number. Every slider has an editable textbox where the value is displayed and controls a certain link of the robot. If we use a slider the associated editable textbox value is updated and vice versa. Also the robot leg position is updated with the data from the slider or from the textbox. The controls for link length or mass affect all the legs because they are considered identical.

The button label *Leg Control* from fig. 3 launches a second interface like in fig. 4. The second interface allows control of a single leg.

This interface can be used both in offline mode or online mode.

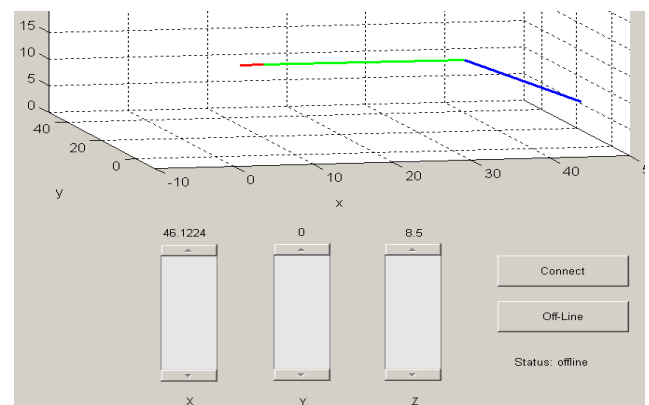


Fig. 4. Leg control using Arduino Duemilanove.

The graphical representation of the robot also allows seeing the shape of the support polygon which is updated according to the legs on the ground. The support polygon is drawn only if the distance from the tip of the leg to the ground is smaller than a threshold. This threshold is

modifiable (but not present in the interface) and was introduced as a way to compensate certain position errors that may occur due to real servomotors.

The simulation program shows all the stages the robot goes through for a better understanding. For simplicity and better understanding of the robot stages the model is drawn in a simpler way.

B. Simulation Algorithm

The authors have elaborated an algorithm in order to achieve the goal of analyzing the static stability of a hexapod robot in gravitational field. The algorithm is structured in 5 steps as following:

- setting the joints values
- determine the mechanical configuration
- determine which legs are on the ground
- evaluation of the static stability condition
- while (condition of static stability = false)
 - determine the rotation line using the minimum distance from G' to support polygon's sides
 - rotate the robot about the line found
 - determine which legs are on the ground
 - evaluation of the static stability condition

The determination of static stability condition is resumed at finding if the projection of robot's centre of mass is inside the support polygon. For this the authors used the following algorithm:

- determine the convex polygon
- determine the area of the convex polygon (A)
- form the n triangle using 2 consecutive sides of the support polygon and the projection of centre of mass, G', (e.g. ABG', BCG'... etc.)
- determine the areas of the n triangles formed (Ai)
- if ($\sum A_i = A$) then condition=true
- else condition=false

C. Leg Control

The system proposed by the authors (Fig. 5) is similar with the system xPC-Target component of Matlab. The software that make possible the communication between Arduino board and Matlab has been released with the last version of Matlab. Mathworks [5] has also developed support for Arduino in Simulink. A part of the communication software is uploaded on the Arduino board and plays the server role.

There are two programs that can be uploaded on the board:

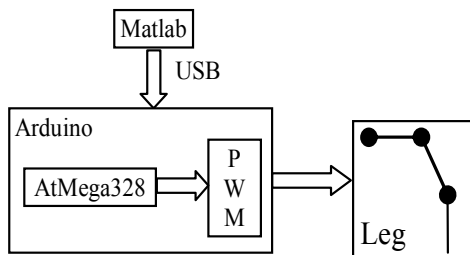


Fig. 5. Leg control system.

--*adiosrv.pde* with which all the input and output of the board can be manipulated.

--*motorsrv.pde* designed exclusively for motor control via a motor shield.

The major drawback of using the original *motorsrv.pde* was that this file was developed as support for a specific motor shield that can only control 2 servomotors. So we modify the file in a way that now allows using all 6 PWM channels available.

The other part is a Matlab class (*arduino.m*) and plays the role of the client. Once the class is instantiated it makes possible sending commands over USB port.

The direction of motion of the servo is automatically determined. If we set a rotation with 90° and then a 30° rotation the servo will rotate 60° anticlockwise.

D. Walking Algorithm

During walking or running the leg move cyclically and in order to facilitate analysis or control, the motion of the leg is often partitioned in two parts:

- support phase or stance when the robot uses the leg to support and propel.
- transfer phase or swing when the leg is moved from one foothold to the next.

The stance part of the walking algorithm is supposed to move the leg in a straight line. The swing part of the algorithm must lift the leg off the ground, move it back to the starting position and lower it down to the ground again. The walking algorithm is based on the kinematical model of the leg.

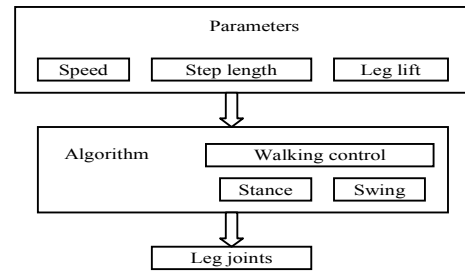


Fig. 6. Walking Algorithm Diagram.

Parameters introduce in the algorithm (Fig. 6) are:

- speed; define the speed of the leg tip,
- step length; define the length of the step,
- leg lift; this defines how high the leg is lifted when it's in the swing-cycle.

For a smooth straight motion, the swing time must be equal to the stance time for each leg.

In the stance part of a step, the leg only moves in a straight line. At time t from the start of the step, the coordinates (x, y, z) for the trajectory will be:

$$\begin{aligned} x &= \text{leglength} \\ y &= \frac{\text{steplength}}{2} - t * \text{speed} \\ z &= -\text{legheight} \end{aligned} \quad (9)$$

In the swing part of the step, the leg first has to be lifted of the ground, and then moved back to where the next step is supposed to start and then lowered to the ground. To find out where in the swing-cycle the leg is at time t from the start of the cycle we first have to calculate the total length the leg has to travel:

$$dist = \frac{(2 * leglift - steplength) * t}{swing_cycle} \quad (10)$$

One step consists of one stance cycle and one swing cycle but to maintain a smooth motion of the leg, it's important that the swing cycle continues where the stance cycle ended, and that the swing cycle ends where the new stance cycle starts. This does not only hold for the position, but also for the speed and the direction of the speed.

V. EXPERIMENTAL RESULTS

A. Free fall analysis

The free fall analysis represents what happens with the robot left to fall on the ground from a height greater than the extension of the legs. Keeping in mind that the joints are locked by the values prescribed by the user, no extra movements are allowed (no active stability). The only force that acts upon the robot is the gravitational force. For a given set of joint values the robot passes through many transitory stages until it becomes statically stable (Fig. 7.a, b, and c). Legs that have contact with the ground determine the shape of the support polygon (triangle, quadrilateral, pentagon or hexagon). In order to know if the robot achieves static stability the projection of G (G') must be inside the support polygon. To solve this problem the above algorithm is applied.

As it can be seen from figure 7.a, when the robot falls the first legs that reach the ground are those nearest the ground. Also G' is not inside the support polygon and the robot continuing its falling and rotates about the line determined by the leg 4 and leg 5. The rotation line is determined by calculating the minimum distance from G' to polygon's sides. In this case the first leg closest to ground that will provide support is leg 2.

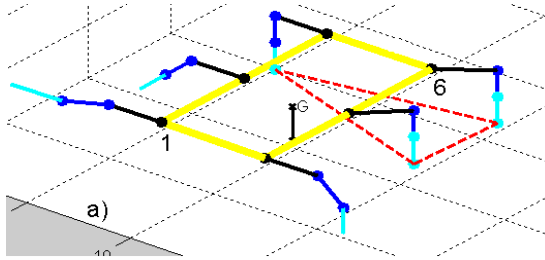


Fig. 7.a: Phase one of falling.

In figure 7.b it can be seen that even in this configuration G' is not inside the support polygon and the robot continues it's falling and rotates about the line determined by leg 2 and leg 5 until the first leg touches the ground, which in this case, is leg 3.

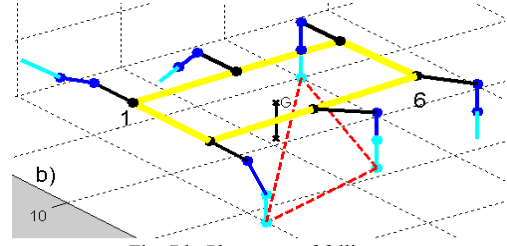


Fig. 7.b: Phase two of falling.

In figure 7.c a new configuration is formed and if the algorithm described above it is applied, point G' is inside the support polygon and the robot becomes statically stable and the falling stops.

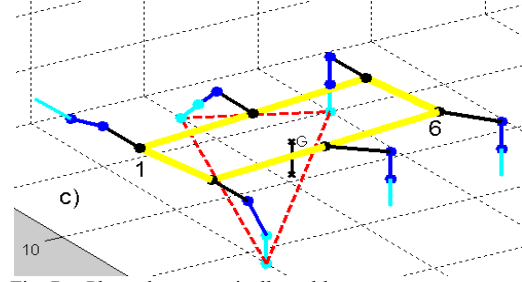


Fig. 7.c: Phase three – statically stable.

B. Transitory Analysis

In this mode of analysis the robot passes between two static regimes. A static regime is identified by the condition of stability. The user can alter a static regime using the controls for joint values or position of the leg tip. This analysis can also be interpreted as a continuation of *free fall analysis* case if the condition of stability has been met.

In figure 8.a the robot has static stability, the support polygon described by the legs on the ground is a quadrilateral (formed by legs 1, 2, 5 and 6) and the projection of the centre of mass is inside the support polygon.

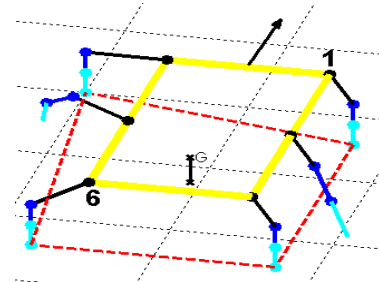


Fig. 8.a: Phase one – statically stable.

Next, lifting leg 2 the support polygon changes its shape becoming a triangle. Using the algorithm described above, the projection of G is not inside the support polygon and the robot becomes statically unstable and starts falling (Fig. 8.b).

Following the algorithm the next leg closest to the ground is leg number 4. In figure 8.c the projection of G is inside the newly support polygon, the robot stops falling and becomes statically stable.

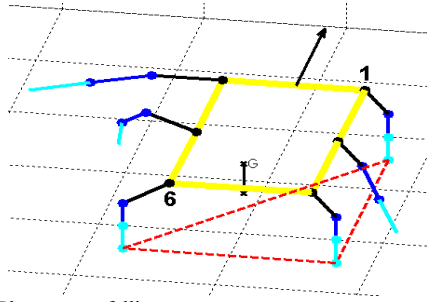


Fig. 8. b: Phase two – falling

C. Leg Control

The algorithm for controlling the leg joints, including direct kinematics and inverse kinematics, was implemented in Matlab in order to analyze leg performance.

The results of the tests can be view in the chart below (Fig. 9).

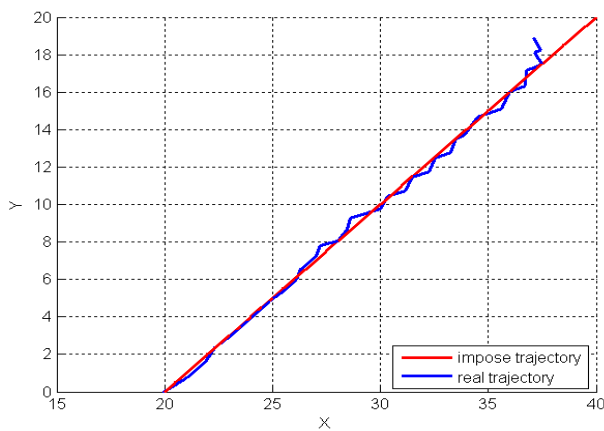


Fig. 9. Leg control results.

The main errors occurred due to the fact that we can only send integer numbers for joint values.

Other errors occurred due to the fact that the joint are assembled using bolts and nuts. Normal usage of the robot causes the nuts and bolts to loosen causing a lot of clearance in the joints.

VI. CONCLUSION

In this paper a simulation platform for legged mobile robots was presented, that allows stability analysis and full control of the robot.

Free fall analysis is useful for investigating what happens with the robot on uneven terrain or for accidents that may happen due to lose of contact, slippery surface, and servomotor failure, power supply failure.

Transitory analysis represents a more important case for locomotion, gait generation. When starting to develop a gait cycle we can have a big picture of what happens when the robot starts to move, how the support polygon changes and what actions (what leg should be actuated) must be applied to the robot in order to meet condition of stability.

The interface was designed to be simple and intuitive and to offer the user a simple and efficient way to control every aspects of the robot (angles, masses, lengths, leg control).

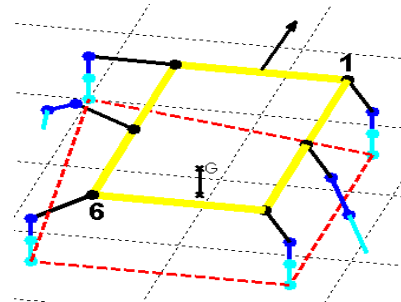


Fig. 8. c: Phase three – statically stable

The two analyses made in this article represent the bases for next activities on static stability.

Hardware leg control will be imbue to better presicion.

In the future the program will be upgraded permitting additional controls and functions for stability analysis (including dynamic stability) on uneven ground and implementing collision detection algorithms. Also the results of these studies represent the bases for different strategies of locomotion on different terrains. The experimental results will become a standard for a real hexapod robot.

ACKNOWLEDGMENT

This work was partially supported by strategic grant POSDRU/88/1.5/S/50783, Project ID 50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resource Development 2007 – 2013.

REFERENCES

- [1] Xilun Ding, Zhiying Wang, A. Rovetta, J. M. Zhu, "Locomotion analysis of hexapod robot", *"Climbing and Walking Robots"*, March 2010, ISBN: 978-953-307-030-8 pp.291-310 India.
- [2] http://www.atmel.com/dyn/resources/prod_documents/8271S.pdf
- [3] S. Manoiu-Olaru, M. Nitulescu, "Stability analysis software platform dedicated for a hexapod robot", *"Proceedings Conference on Control Systems and Computer Science"* ISSN: 2066-4451, pp.385-390 Bucharest Romania 2011.
- [4] R. J. Schilling, *"Fundamentals of robotics: analysis and control"*, ISBN: 0-13-344433-3, Prentice Hall, New Jersey, USA, 1990.
- [5] W. Krzysztof, B. Dominik, K. Andrzej, "Control and Environment sensing system for a six-legged robot", *"Journal of Automation, Mobile Robotics & Intelligent Systems"*, Vol 2 N°3 2008.
- [6] J. Barreto, A. Trigo, P. Menezes, J. Dias, *"Kinematic and dynamic modeling of a six legged robot"*.
- [7] F. Fahimi, *"Autonomous robots: modeling, path planning, and control"*, Springer. 2008
- [8] R. Brian, R. Hunt, L. Lipsman, J. M. Rosenberg, *"A guide to MATLAB for beginners and experienced users"*, ISBN-13:978-0-521-00859-4, Cambridge University Press 2001.

Closed loop control for a 3D tentacle robot using artificial vision system

F. Manta, S. Dumitru, D. Cojocaru

Abstract– Nowadays we can observe a grown in interest regarding the robots with high mobility degree, wanted for their capability to avoid obstacles and reach difficult positions in their working space. One particular class of these robots consists in tentacle robots, biologically inspired by cephalopods. Mathematical models and several control structures were presented during the last years, but a major problem remains: how to sense the current robot shape and position, since this class of robots doesn't have discrete joints? A solution to this problem is given by visual servoing systems, and in this paper we will present a novel approach to solve it.

I. INTRODUCTION

HIGH mobility degree robots promise better performance in unstructured and difficult environments. Tentacle robots, like the one presented in this paper, being biologically inspired, feature a continuous shape with no joints and no rigid links. The wide range of abilities recommends this class of robots for operations on difficult environments, such as industrial rescue operations on seabed.

First attempts to realize a tentacle robot was made by department research team in 1984, when Ivănescu published first mathematical models for such robot. First model used wired actuation and has just one segment. Over time, the project evolved, being used different actuation systems (servo electric motors, shape memory alloy motors, electro- and magnetorheological fluid hydraulics) and different configurations.

The project of current tentacle robot was started in 2008, when a new experimental platform was designed. The robot arm has three segments, serial connected along a central elastic backbone, each one being actuated by 3 stepper motors. The exterior shape of the robot arm is at the second version (fig. 1), a truncated cone shape, first one being cylindrical. Until now the research team explored the finite element model simulation, the kinematic and dynamic models, and an open loop control, based on the stepper motors properties. In this paper we will present a visual servoing for closed loop control.

Manuscript received June 3, 2010.

F. Manta is with the Faculty of Automation, Computers and Electronics, University of Craiova, 200440 Romaina (e-mail: manta@robotics.ucv.ro).

S. Dumitru is with the Faculty of Automation, Computers and Electronics, University of Craiova, 200440 Romaina (e-mail: dumitru@robotics.ucv.ro).

D. Cojocaru is with the Faculty of Automation, Computers and Electronics, University of Craiova, 200440 Romaina (e-mail: cojocaru@robotics.ucv.ro)

The artificial vision system is formed by 2 color video cameras, a video frame grabber, and a computer to analyze the image and to extract relevant data from it. The final output will be the curvature and orientation for each segment, and will be sent to the control program as inverse reaction on the closed loop. Using the artificial vision system arises 3 problems: how to place the camera, how to extract the relevant information and how to work on unstructured environment.



Fig. 1. Current version of the Tronconic Tentacle Robot developed at University of Craiova

II. CAMERA POSITIONING AND CALIBRATION

As previously presented, the visual servoing system uses 2 identical video color cameras. While we want to control the robot in the 3D space, thus a 3D information about the robot curvature and orientation is needed, and from here the necessity of the 2 identical cameras. The final desiderate is that system should work in any unstructured environment, so we used color video cameras in order to achieve the highest contrast between natural background and markers that we will place on robot. Since the designed control system it is a pseudo-real time (under some restrictions and hypothesis can be considered a real-time control system), the acquisition of the image should be as fast as possible, and the best solution to acquire simultaneous and synchronous image from two cameras was to use a frame grabber with two independent but synchronized acquisition / conversion circuits. A cheaper and more up-to-date solution seemed to be the use of digital output video cameras (offering USB or FireWire data transmission protocol), but using this kind of cameras raises problems regarding the time-arrival of the

video data, and also the synchronization of the images provided by the cameras .

As camera positioning, we concluded that the most reliable and convenient solution is to place the cameras so their optical axes forms a plane, perpendicular on the “origin” position of the robot (the robot arm in a perfect straight position), and the optical axes to be perfectly perpendicular. The height of the previously defined plane is at the middle of the robot arm length (fig. 2), for optical perspective reasons. A fully explanation why we adopted this configuration is given in the Section III.

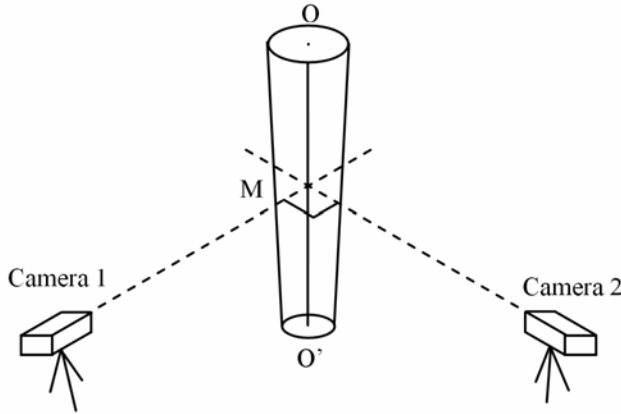


Fig. 2. Camera positioning. Optical axis of the Camera 1 is perpendicular and concurrent relative to the optical axis of the second Camera; the plane determined by the two axes is perpendicular on the OO' backbone in the M point, the median of OO' .

A. Camera positioning

One of the camera positioning problems is to find a way to place the cameras in the proper positions, as defined above, especially to assure that the optical axes are perpendicular and concurrent. The solution proposed by our research team is to use a laser source and a laser beam splitter. A laser beam splitter (or beamsplitter, power splitter) is an optical device which can split an incident laser beam into two or more beams, which may or may not have the same optical power. The laser beam splitter used by us for camera calibration has the form of a cube, made of two triangular glass prisms which are glued together with a transparent resin. The beam separation occurs at this interface within the cube, and if the incident laser beam is perpendicular on the input interface, he is split in two secondary beams, perpendicular one on the other (Fig. 3).

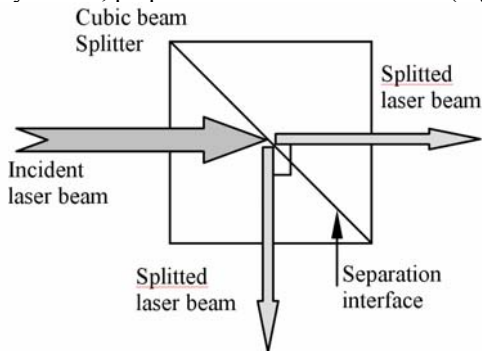


Fig. 3. Cubic laser beam splitter: the incident laser beam and the splitted secondary beams, perpendicular one on the other.

In the following paragraph we will detail de positioning procedure. First, the robot arm will be manually set in the “origin” position. Second, the cubic beam splitter will be placed at the height M , as defined in Fig. 2, as close as possible to the robot’s backbone and perfectly aligned with it, so the plane formed of splitted beams to be perpendicular on the backbone. Third, a laser beam will be oriented perpendicular to the input interface and the splitted laser beams will be generated. On the fourth step, we will place the cameras as follows. The cameras will be oriented thus the cubic prism will be visible in the acquired image. If the camera is not at the proper vertical or horizontal position, the splitted laser beam emitted toward the camera will appear as a thin line, starting from the cubic prism and going toward edge image. Depending on which edge the laser line ends, camera will be translated toward that edge until the laser line will be perfectly horizontal in image, if the edge is one of the upper / lower edges of the image, or perfectly vertical if the edge is one of the left / right edges of the image. Further, the camera will be translated again, on a perpendicular direction, until the laser line becomes a laser spot, meaning that the camera is at the corresponding height / position relative to the wanted final placement. The final step is to change the camera orientation, thus the laser spot to be “seen” in the center of the image. In this moment, the camera optic axis goes thru point M . Same procedure will be repeated for the second camera, following the other splitted beam. Since the goal of this article is to present the entire visual servoing system, we will not insist on the geometrical theory we used.

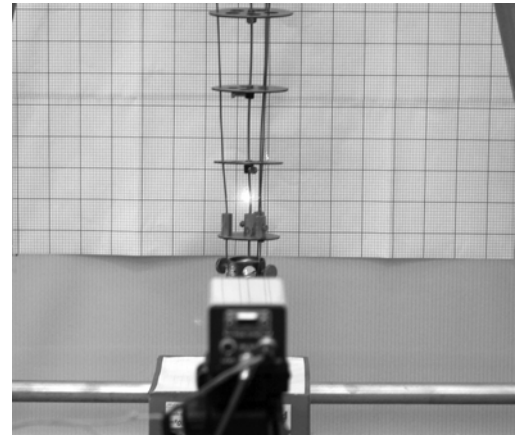


Fig 4. Camera positioning using laser beams

B. Camera calibration

As stated in paper introduction, one of our final goals is to use the robotic system (including the artificial vision control loop) in an unstructured environment. Thus, for a better performance, is preferable that the system should require the fewer setup operations as possible. Therefore, when we chose the method to extract the relevant information from the image, one of the considerations was to use a representation system which to be invariant in relation with robot image upscaling / downscaling. The chosen method consists in determination of curvature described as an arc circle, defined by a radius, center angle

and sector orientation. Using proper markers, as described in Section III, will make the process invariant in relation with robot's dimensions in image, the relevant information being extracted only from robot's shape. This approach will eliminate the necessity of a classic camera calibration, in which to establish a transformation relation between real world dimensions and image dimensions, and the algorithm can extract the relevant informations using one pixel as elementary size unit, since the pixels on the camera sensor have a square shape design. The only parameters left to be calibrated are the image luminosity, focal distance and color space, which can be done using available software functions and which is not relevant for the topic of current paper.

III. DETERMINING CRITICAL PARAMETERS FROM THE IMAGE

A. Image space definition

As previously enounced, the desired result from artificial vision system is the current curvature and orientation of the robot in the three-dimensional space. This information can be defined and extracted in many coordinate systems, but in our application we take in consideration that an image is a projection of the 3D space on a plane. This made Cartesian coordinate system the most convenient solution, since the camera calibration is not required, one pixel will be considered the elementary size unit. As enounced, an image is a plane projection of a three-dimensional scene, so isn't enough to express 3D information. Yet a combination of information contained in two images taken from 2 different known points of observation can describe a 3D space position – one particular case is human visual system: the two eyes who enable us to perceive three-dimensional images. Reproductions and applications of such stereoscopic visual systems are much known nowadays; but for our application this kind of visual system is not the best choice, since a large volume of computations is needed in order to process the perspective geometry involved, and our desiderate is to achieve a very fast algorithm in order to have a pseudo-real time control system for our robot. The perspective geometry can be avoided by placing the cameras in such way that the information extracted from the image to be fit directly the Cartesian coordinate system. These make us to conclude that the best arrangement is to have the image planes perpendicular to each other, hence the camera configuration presented in Section II.

B. Critical points in the image - markers

Now having the image space defined, we chose as critical points the endpoints and the middles of the segments. Since the segments are serial connected, the endpoints between segment I and II, respectively segment II and III are identical, so we will consider it once. In order to identify these points in images, they must be marked on the robot structure, and also must be visible against the background. As first solution, we used a white background and we covered the robot in a white coating. Each end of segment was marked with a black line, and each middle point with a

dot (fig. 5). The critical points are determined as the center of mass for each line or dot, respectively.



Fig 5. Black lines mark the extremities of segments. Critical points are extracted as the center of mass for each black line

This marking method was very good for identifying the endpoints for each segment, but is not able to cover a situation that is very likely to occur during robot operation. Since the actuation system is based on stepper motors, in certain operating modes the actuation system limits can be reached and there can appear the loss of steps phenomenon, which will lead to a robot malfunction. This malfunction can be observed as unwanted and permanent twist of the robot structure.

In order to approach this situation in our visual servoing control, we changed the markers to another design that will permit the "twist" detection (fig 6).



Fig 6. New set of markers which able de artificial vision system to identify the critical points, but also if the twist phenomenon occurs.

The markers are designed to contain a white region and they are placed on the robot structure so this region can be found the middle of each visible black line. In robot normal operation, this white region will be always found in the middle of the black line. If some offset (to the right or to the left) is detected in the image by the artificial vision algorithm, it means that a robot malfunction occurred, the stepper motors lost a certain number of steps and the robot is “twisted”.

C. Artificial vision algorithm

The role of the artificial vision system is to identify the markers on the image, to compute relevant data and to offer as result the orientation and the curvature of the robot segments.

In a first stage, the acquired pairs of images are filtered and critical points determined and identified. For a faster algorithm, some heuristics are considered. Thus, knowing where the robot base is placed – recognized from the image as a simple black line, as seen on the upper part of fig 6 – and the maximum deformation that robot arm can exhibit, the next marker will be searched only in a small area of the image, as shown in fig 7.



Fig 7. Markers detection. With darker color is indicated the search region. With lighter color are labeled the found markers.

Also, for each black line the position of the white spot is checked, to verify if a malfunction occurred. The result will be the segments extremities positions expressed in image coordinates, as vertical and horizontal locations expressed in pixels. In this stage the information is bi-dimensional, a critical point place having attached 2 sets of 2D positions extracted from the two images.

Since the cameras was positioned so their optical axes to be concurrent and perpendicular one on each other and also on the robot backbone (see Section II paragraph A), in a second stage, we can superpose the two images to the Cartesian coordinate system planes, one to XOZ plane, and the other to the YOZ plane (fig. 7.). In this way the horizontal and vertical positions identified in the two images

will be unified in a three-dimensional space, and the image coordinates will become a 3D space coordinates.

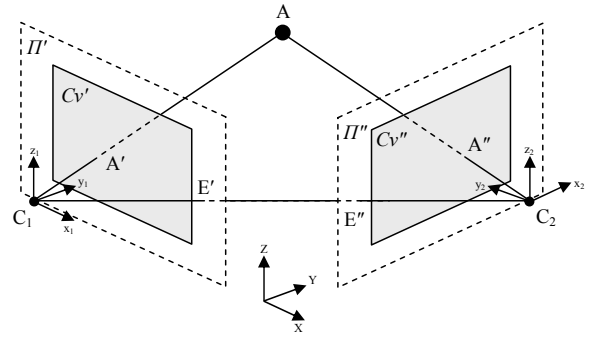


Fig 7. The unification of the two image planes in a three-dimensional image space.

After this unification of two 2-D images in a 3-D image space, each point from common field of view of the two video cameras should be identified by its X, Y, Z coordinates. Thus, the position of this point, on the Z axis, should be the same in both images. But this is true only for an ideal situation, when the cameras are placed at infinite distance from the robot, and all the rays are parallel. In practice, this affirmation is valid only for a very particular class of space points. For all the remaining points, the Z position “seen” from the two images will be different. Moreover, the markers are placed on the exterior robot surface, and there will be position differences between markers used for same element (ie. for marking the end of first segment), one viewed from one camera and the other one viewed by the other camera.

In this context, we propose a solution to solve the problem, a solution which will also bring significant improvements to the algorithm processing speed. Instead of computing each time the critical points positions, we propose the use of a Look-Up Table (LUT), in which for each point in space (in X, Y, Z coordinates) we will attach the $P_{i_1 j_1 i_2 j_2}$ position from the two images, where i_1 is the Z position extracted from image “seen” by first camera, j_1 is the X position, i_2 is the Z position extracted from image “seen” by the other camera, and j_2 is the Y position as viewed from the same camera. In this way, by knowing the $P_{i_1 j_1 i_2 j_2}$ position after marker detection process, the P_{XYZ} is immediately known without any additional computations.

Once known the $P_{i_1 j_1 i_2 j_2}$ positions for all markers, hence the P_{XYZ} positions, we can determine the curvature for robot’s segments. For one segment curvature determination, we will use the three P_{XYZ} points corresponding to the segment extremities and the middle of segment. With only these three points we can determine the radius and the angle of the arc circle which approximate the segment curvature. The geometry involved is explained briefly as follows.

Having three points $P_{1XYZ}, P_{2XYZ}, P_{3XYZ}$, on an arc circle of center C , radius R and angle θ , P_{1XYZ} and P_{3XYZ} placed at extremities and P_{2XYZ} on the middle, we are looking to

determine the arc circle properties R and θ . For this, a fourth point M_{XYZ} will be determined, the middle of the chord uniting the arc circle extremities P_{1XYZ} and P_{3XYZ} (fig. 8).

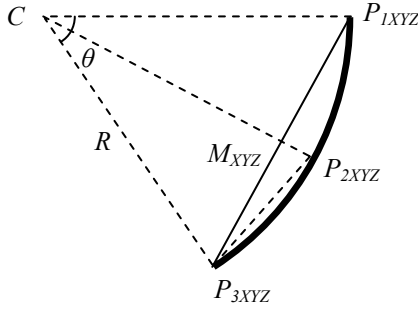


Fig 8. Determination of arc circle parameters

The M_{XYZ} position is determined as an average of P_{1XYZ} and P_{3XYZ} positions. Thus we can form the $M_{XYZ} P_{2XYZ} P_{3XYZ}$ triangle, from which we will determine the θ angle.

In the $CP_{2XYZ} P_{3XYZ}$ triangle, the $P_{2XYZ} CP_{3XYZ}$ angle is equal with $\frac{\theta}{2}$, since CM_{XYZ} is the median in isosceles $CP_{1XYZ} P_{3XYZ}$ triangle. The $CP_{2XYZ} P_{3XYZ}$ angle, computed in $CP_{2XYZ} P_{3XYZ}$ isosceles triangle, is the equal with:

$$\angle CP_{2XYZ} P_{3XYZ} = \frac{180^\circ - \frac{\theta}{2}}{2} = 90^\circ - \frac{\theta}{4}$$

(1)

In $M_{XYZ} P_{2XYZ} P_{3XYZ}$ triangle, the $MP_{2XYZ} P_{3XYZ}$ angle is equal with:

$$\begin{aligned} \angle M_{XYZ} P_{2XYZ} P_{3XYZ} &= 90^\circ - \angle CP_{2XYZ} P_{3XYZ} \\ &= 90^\circ - \left(90^\circ - \frac{\theta}{4}\right) = \frac{\theta}{4} \end{aligned} \quad (2)$$

In the same triangle $M_{XYZ} P_{2XYZ} P_{3XYZ}$,

$$\sin(\angle M_{XYZ} P_{2XYZ} P_{3XYZ}) = \frac{[M_{XYZ} P_{2XYZ}]}{[P_{2XYZ} P_{3XYZ}]}$$

(3)

thus:

$$\theta = 4 \cdot \arcsin\left(\frac{[M_{XYZ} P_{2XYZ}]}{[P_{2XYZ} P_{3XYZ}]}\right)$$

(4)

R can be determined from $CM_{XYZ} P_{3XYZ}$ triangle, since:

$$[M_{XYZ} P_{3XYZ}] = [CP_{3XYZ}] \cdot \sin \angle CP_{3XYZ} M_{XYZ}$$

(5)

we have:

$$R = [CP_{3XYZ}] = \frac{[M_{XYZ} P_{3XYZ}]}{\sin \angle CP_{3XYZ} M_{XYZ}} \quad (5)$$

where the $CP_{3XYZ} M_{XYZ}$ angle have the value:

$$\begin{aligned} \angle CP_{3XYZ} P_{2XYZ} - \angle M_{XYZ} P_{3XYZ} P_{2XYZ} &= \left(90^\circ - \frac{\theta}{4}\right) - \frac{\theta}{4} \\ &= 90^\circ - \frac{\theta}{2} \end{aligned} \quad (6)$$

In this way we determine the arc circle parameters for each one of the segments. Using the presented algorithm, the computation time needed for one cycle (acquisition – markers position in 3D space – arc circle parameters determinations) is less than the frame grabber acquisition rate.

IV. EXPERIMENTS

In order to validate our algorithm, we conducted a series of tests. One of the tests consisted in creation of some benchmark images to determine the entire system precision. We designed some of the possible robot configurations as well known serial arc circles (fig. 9) and placed the markers on the critical points.

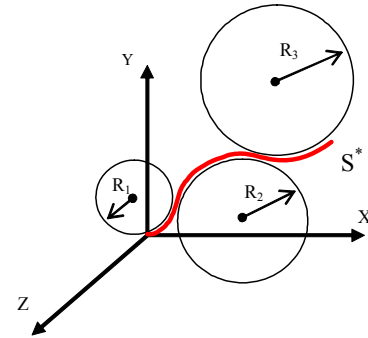


Fig 9. Creating the baseline for the benchmark image.

We printed these models and presented them to the video cameras as being the real robot (fig. 10).



Fig 10. One of the printed benchmark images.

For each one of the image benchmarks we compared the well known parameters with the measured ones and also we made more than one captures to test the system stability. In Table 1 we present a set of results. For each benchmark image, we known the (R, θ) values for each of the three segments, resulting (R_i, θ_i) where $i=1...3$, and we capture the images and run the algorithm for four times, having $M_j(R_i, \theta_i)$ results, $j=1...4$.

Table 1
System and algorithm test results

	Known (R_i, θ_i)	$M_1(R_i, \theta_i)$	$M_2(R_i, \theta_i)$	$M_3(R_i, \theta_i)$	$M_4(R_i, \theta_i)$
First benchmark image	460, 5°	460.2; 4°57"	459.2; 4°59"	457.4; 5°01"	459.2; 4°59"
	760, 3°	758.3; 3°02"	758.3; 3°02"	774.4; 2°58"	774.4; 2°58"
	229, 10°	228; 10°02"	227.4; 10°04"	229.8; 9°58"	228; 10°02"
Second benchmark image	572, 4°	568.6; 4°02"	572; 4°00"	577.2; 3°58"	577.2; 3°58"
	190, 12°	190.2; 12°03"	190.7; 12°01"	191.6; 11°58"	190.7; 12°01"
	380, 6°	381.1; 6°01"	380; 6°00"	380; 6°00"	381.1; 6°01"
Third benchmark image	208, 11°	207.7; 11°02"	207.7; 11°02"	208; 11°00"	209.1; 10°57"
	152, 15°	151.4; 15°10"	151.8; 15°06"	153.3; 14°57"	152.4; 15°02"
	127.3, 18°	126.2; 18°12"	126.4; 18°10"	128.1; 17°52"	126.6; 18°07"

From Table 1 one can observe that the entire artificial vision system offers very accurate information about the robot shape. As observations, the small differences in measured values for the same image are caused by the video system: video cameras and frame grabber small constructive imperfections, as indicated in their datasheets, and also by small deviations of the illumination system. In general, the error between known and measured values is proportional with the initial known value: for a bigger value, a bigger error. These results made us to conclude that the algorithm perform well.

V. CONCLUSIONS AND FUTURE IMPROVEMENTS

In this paper we presented a novel visual-based inverse reaction for a closed loop control, designed to command a tentacle robot in a three-dimensional operating space. We presented the current version of an entire generation of tentacle robots, and a new method to place the two video cameras using laser instruments to "see" the robot arm configuration in a three-dimensional space. We proposed a new model of visual markers, which allows the control system to determine the current robot arm configuration and to check if errors in operation occurred, resulting in an unwanted torsion of robot arm. We created a unified image space in a form of an Look-Up Table, to improve the algorithm speed and to offer near real-time information about the robot arm configuration. The parameters of robot arm configuration were determined using a geometric approach. Also, we tested the artificial video system to verify the precision of results and the stability of measurements.

As future tests we are looking to test the artificial vision system in an unstructured space, and to test the system along with the movement command system.

REFERENCES

- [1] S. Hirose, *Biologically inspired robots: Snake-like locomotors and manipulators*, Oxford University Press, 1993
- [2] M. Ivanescu, "Dynamic Control of a Tentacle Manipulator", in *Robotics and Factories of the Future*, Ed. Springer-Verlag, pp. 317-326, 1984
- [3] M. Ivanescu, "A New Manipulator Arm: A Tentacle Model", *Recent Trends in Robotics*, North Holland, pp. 51-56, 1986
- [4] M. Ivanescu, D. Cojocaru, "Visual Based Control System for a Tentacle Manipulator", *WSEAS Transactions on Information Science and Applications*, Issue 3, Volume 3, pp. 610-617, ISSN 1790-0832 2006
- [5] D. Cojocaru, T. R. Tanasie, M. Ivanescu, "Graphic Simulator for the Visual Control of Hyperredundant Robots", *WSEAS Transactions on Systems and Control*, Issue 9, Volume 3, pp. 759-768, September 2008, ISSN 1991-8763.
- [6] D. Cojocaru, T. R. Tanasie, "Calibration Application for Real-Time Hyperredundant Robot's Vision Control", *Proceedings of the RAAD 2008 17th International Workshop on Robotics in Alpe-Adria-Danube Region*, ISBN 978-88-9037-0-8, paper no. 39, September 15-17, 2008, Ancona, Italy
- [7] D. Cojocaru, F. Manta, T. R. Tanasie, C. Pana, 2008, "3D Vision for Continuum Robot Control", *Bulletin of the Transilvania University of Brasov*, Vol. 15(50) Series A, ISSN 1223-9631, Special Issue No. 1 Vol. I, 2008, p331-336, *Proceedings of the International Conference ROBOTICS'08*, ISBN 978-973-598-387-1, 13-14 November, 2008, Brasov, Romania
- [8] G. Boccolato, F. Manta, S. Dumitru, D. Cojocaru, "3D Control for a Tentacle Robot", *3rd International Conference on Applied Mathematics, Simulation, Modelling (ASM'09)*, Vouliagmeni Beach, Athens, Greece, 29-31 December 2009
- [9] D. Cojocaru, M. Ivanescu, R.T. Tanasie, S. Dumitru, F. Manta, "Vision Control for Hyperredundant Robots", *IJAA International Journal Automation Austria*, ISSN 1562 – 2703, nr 18, 2010

Management workflow for consultants in a company

Marior Dan Adrian, Constantin Cîrciumaru

Abstract— The present paper is an effort to partially automate the workflow to hire an external consultant in the cases when the existing expertise is not enough. Usually this happens in but is not limited to major worldwide companies. For the implementation we use a virtual intranet portal custom built for the Ford Motor Company plant in Craiova. Although it is used for development, the main idea was to make it deployable in production with changes related to the server environment architecture – in production we are no longer talking about a single machine.

I. INTRODUCTION

EVERY major company in the world has to consider hiring, at some point in time, one or more external consultants. The consultants may be hired for a limited or unlimited period of time. The expertise they bring can prove invaluable in certain conditions, so every company must have a workflow predefined for such situations. Their services are usually acquired because they can bring “wisdom, objectivity and expertise to the organization, and they are well accepted especially if they are credentialled, published and/or well known”[1]. For example, they become necessary when there is a need for a company to implement a project with specific requirements that the company does not have the appropriate expertise for (that actually means the company does not have the money to invest in training a number of persons from the staff or they need to get started immediately on the project) or a project needs maintenance that the existing employees cannot handle.

Although a few of the steps in the hiring or collaboration process are done by humans, most of them need to be backed by automated systems, and this is where Sharepoint excels. Microsoft’s software platform for collaboration in this case provides the means and a good alternative to partially automate this process and increase the efficiency of the company. The Sharepoint suite of technologies and development tools is sold by the Redmond company in versions appropriate for every type of business, from the Windows Sharepoint Services 3.0 suite of free tools to the Enterprise release of Sharepoint which costs tens of thousands of euros. Of course there is no need to use the enterprise version for a workflow which can be put in place with the help of the Standard release or less. Every company

This work was supported (in part) by the grant no. POSDRU/6/1.5/S/14b.

Marior Dan Adrian is with the Department of Automation in the Faculty of Automation, Computers and Electronics, Craiova, (e-mail: marior@automation.ucv.ro).

Constantin Cîrciumaru is with the IT Department of the Ford Motor Company, Craiova (e-mail: ccircium@ford.com).

which considers implementing a Sharepoint collaboration environment has to think of the version needed to solve existing problems, and to choose accordingly. Of course, if the future brings positive evolutions, companies can always activate the enterprise features. Since building a Sharepoint development environment for various occasions is a very time consuming process, we have built an infrastructure in order to leverage more than a workflow and/or other applications, which can be used for example in a factory like Ford (has actually been designed in accordance with the recommendations of the Ford Craiova plant IT Manager).

All the sections of the article serve the practical purpose of describing the foundation and the algorithm to implement a document management workflow for a large company. In section I we motivate the usage of external consultants, the object of this workflow, we describe a few aspects of the alternative we chose to have the work done and mentioned a few aspects of the implementation. In section II we presented a few theoretical and also more specific aspects of workflows in Sharepoint, with the environment’s constructs like lists and document libraries, and we also introduced aspects of x86 architecture virtualization. Afterwards, section III presents the infrastructure needed to host the workflow, the “Consulting Contracts” document library and the “Announcements” list and section IV describes the workflow in detail; conclusions are presented in the final section of the paper, in order to discuss the significance of the present work.

II. WORKFLOWS

Workflows are specific to human activity of every kind [2]; also called business processes, a term which better describes them for any economic activity, workflows can easily be discovered in our day-to-day lives, because they actually express the logic needed to carry out any activity, in any domain or situation [4]. They usually involve two or more persons, and have the purpose to create or add value to the organization. There are two types of workflows: sequential and parallel ones. In sequential workflows every step depends on the prior steps (cannot happen if the prior ones have not occurred) while in parallel workflows two or more steps can occur at the same time. A workflow represents a sequence of related steps, of related operations. It could be seen as a virtualization (in paper or digital form) of real work and of course the relationship works both ways; one particular class of workflows is employed in most companies choosing to implement Sharepoint environments.

Before building the workflow, the developer must first

build his infrastructure to support the workflows (once an infrastructure is built, it can host multiple workflows). The persons responsible with the implementation of the project have the possibility to model the structure of the portal after the organization schematics of the company or departments they are hired to work for, and that usually means the already existing users and groups structure; inside the top level site (domain), which is usually the host for the workflows, the developer can build other types of Sharepoint constructs, such as subsites, site collections, lists, document libraries, dashboard pages, announcements lists, calendars, SQL Server analysis services indicators, Excel indicators, reports, basic web pages, etc. [3]

There are two appropriate ways to build and implement a workflow project: the more laborious way, where the conception, design and building of the project is exclusively or partially the merit of the author/authors of the project, and the intelligent way, by using the already existing elements, adapting them and the environment the work is done in to the tasks that need completion. Nevertheless, this decrease in difficulty implied by the latter choice comes at a price, and that is the level of abstraction we have to assimilate and place on top of what we already know; in time this was observed to be typical for every emerging technology [2]. The two means of building workflows correspond to using Visual Studio and Sharepoint Designer respectively. The first is actually suitable for building more than one type of applications for the Sharepoint environment, for example it can be used to build features (plug-in-like functionalities).

Sharepoint, or the MOSS 2007 family of technologies (Microsoft Office Sharepoint Server), is not used only for workflow development, but also for a large amount of problems in the business and project management area of expertise. Here are some of the problems to be solved with MOSS: web content management, enterprise level search, data management and collaboration, records management, dashboards and other business intelligence tasks, workflows, metadata tasks (for example columns in lists), dynamic and customizable navigation and others [3].

In order to adhere to industry proven standards we used a custom built virtual development environment, modelled with the purpose of being transferable to production after the necessary tests and of course after all desired applications will have been implemented. Using virtual machines for development and also for other purposes is not a new idea but in the recent years it has become more of a necessity than an IT trend, due to the priceless flexibility it offers. The motivation behind the virtualization trend is enormous, and the statistics best reflect the situation: according to research published by the Gartner Group, in the year 2009 almost 18% of servers ran virtualized, in 2010 the percentage grew to 28 and this year it seems we are reaching 50 % of server loads worldwide [6]. The research field of virtualization has been very prolific and thus has continued over the years to bring the technology to its maturity which we are enjoying

nowadays. This is why we can safely say that the virtual environments can successfully replace real ones, and provide important characteristics usually more difficult to achieve with real systems.

Remaining true to best practices of working with the Sharepoint environment, we first built the infrastructure to host the workflow, the intranet portal for the Ford plant in Craiova; this is the development phase, in which it is best to use virtual machines because of their invaluable facilities such as “snapshots” of the system, capturing the state at desired moments in time. This implies an easy way to repair damage by reverting to a previous snapshot. There are a number of advantages of using virtual machines on a large scale in the IT world, such as isolation, standardization, consolidation, ease of testing and mobility. Isolation is really a business requirement, because running all the applications on a single server is a good idea but many times it is the cause for all sorts of conflicts and that does not even account for faulty software. [3]

A virtual machine is really independent from the others and from the host operating environment. Of course performance is an but the performance loss is kept to a minimum for most cases. Consolidation essentially results in fewer physical machines being used, which in turn drives financial savings.

The facilities regarding testing of various scenarios and applications are very useful for developers of any kind and consist of snapshot and rollback capabilities. System administrators can also find them useful for testing the effect of various applications on the operating environment. Virtual machines are also very easy to move between hosts, due to the fact that they are stored on disk as files.

The benefits of having a standardized operating environment are reduced support costs and increased share of resources to be reallocated to critical tasks. Although virtual machines can be hosted under a large variety of hardware platforms when built they all result in the same operating environment, and that is a great achievement for the researchers in the field.

III. THE SHAREPOINT WORKFLOW INFRASTRUCTURE

Before we dive into the details of the infrastructure it is better to mention the changes needed for deploying it. Let us consider the production environment to be an enterprise-type company such as Ford which has the needs to justify the use of 3 tier architecture and also the financial means to use it. Of course in this situation we can no longer use the initial development setup of the virtual machine, that of a single “three in one” software conglomerate. In light of the new perspective and breaking the conglomerate down into its three constitutive parts, we should install the database server (namely SQL Server 2005), the application server and the web frontend each on its own machine or server farm, in case the company has a large amount of users to accommodate.

Nevertheless, if the required amount of users and work to

do is not overwhelming we can always deploy the virtual machine directly onto a single server considering just the needs for redundancy, power-related problems and fault tolerance. Of course, the machine must possess corresponding computing power, as we can expect from custom built servers.

We shall now describe the Ford Portal which is actually a top level site connected to a Shared Service Provider

(hosted by another top level site). The top level site or domain as it is also called has a few subsites, corresponding to the departments in the company: IT, Logistics, Planning, Quality, Research and Development and others can also be added later in case they decide to take on the digital approach to document management. The portal is accessible just like the main administration console, through a browser, preferably Internet Explorer; it usually is accessible through the internal network of the company – intranet portal – but if needed, for example in cases where the company wants to connect to a business partner, some parts of it can be accessible through the internet.

Upon creation, subsites can be built with the help of templates, and Sharepoint 2007 comes with a comprehensive suite of templates organized into categories: Collaboration (team site, blank site, document workspace, wiki site, blog), Meetings (basic, blank, decision, social and multiplage meeting workspaces), Enterprise (document center, records center, site directory, search center, etc.) and Publishing (news site and publishing site with/without workflow). The template chosen for the root of the portal and for most of the subsites was the Team Site, which comprises of a number of built in elements, such as lists and document libraries. One of them is the “Announcements” list which is used in the application.

Workflows in Sharepoint can be implemented using three methods, the built in method, by means of Sharepoint Designer and with the help of Visual Studio; they are used according to the level of complexity and need for customization of the workflow. We chose to utilize Sharepoint Designer for its ease of use and powerful features, sufficient for most document workflows. From this point of view Visual Studio workflows have more flexibility than Sharepoint Designer ones, which can only be associated with one document library or list.

Besides the “Announcements” list, which is just what the name implies, a document library called Consulting Contracts was built, with the sole purpose of storing the new collaboration contracts created by the management. The document library comes with a predefined set of columns, such as Type, Name, Modified, Modified by, but in order to be able to host the workflow we had to customize it by adding two columns: EmployeeID and EmployeeDept (the ID and Department of the consultant). Views could also be created, as we wanted to see for example only the contracts pertaining to the IT Department.

The screenshot shows a web form titled "New Contract Properties - Server". At the top, there's a header "CONSULTING CONTRACT" and a subtitle "for the Ford Motor Company plant in Craiova". Below this, there are several input fields arranged in a list: "Name:", "Surname:", "Profession:", "Age:", "Sex:", "Marital status:", and "Hiring department:". Above these, there are three fields: "Title:", "EmployeeID:", and "EmployeeDept:" (which has a dropdown menu showing "IT"). The form is displayed within a browser window with a blue sidebar on the left.

Fig. 1. The contract to be filled by the Manager

We also made possible the creation of a new contract directly from the document library (custom content type), based on a document template (.dotx file) which was uploaded to the server.

The template was used to create a content type, another key concept in Sharepoint, and because content types have metadata columns we can store along with the content type various pieces of information such as the name of the document, person who created or modified it, date and time of last modification, etc., which can all be accessed just like any programming variable in Sharepoint Designer. These are only a few of the advantages of working with Sharepoint.

Preparing the way for the workflow, we also activated the requirement of approval for newly submitted items, the versioning capability was set to store only the major versions of the documents, the permission to see the draft items was given only to those who have the right to approve items, and the checking out before editing capability was activated. There is a possibility that more than one department wants to implement the same type of consulting contracts so in order to prevent them from “reinventing the wheel” we can save the document library as a site template.

IV. WORKFLOW IMPLEMENTATION

The workflow consists of a few steps which can be described as follows: the manager or a person in charge creates a new contract by pressing the appropriate button, which triggers the instantiation of a new Word document with metadata (for example the fields seen in the previous figure). After the manager saves the document, it appears in the document library with the name of the new consultant, date and time of creation, originating user, ID and Department of the consultant, initial status of the document (set to “Pending” as we shall see in Figure 3) and the status of the workflow. Consequently, in the Announcements list we should see the item automatically created in relation to the contract just created.

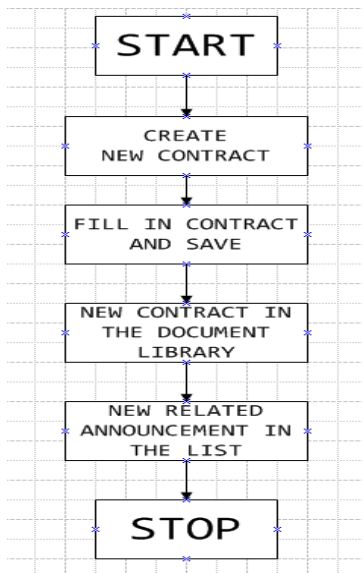


Fig. 2. The workflow diagram

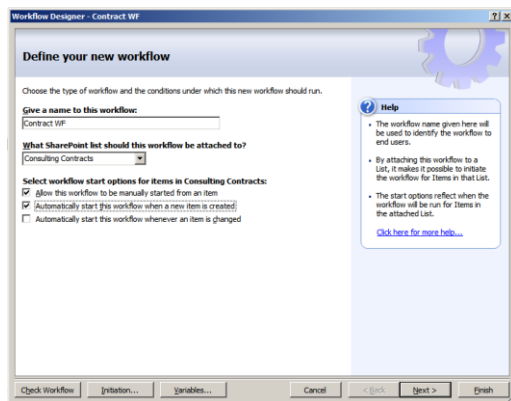


Fig. 3. The initial workflow definition

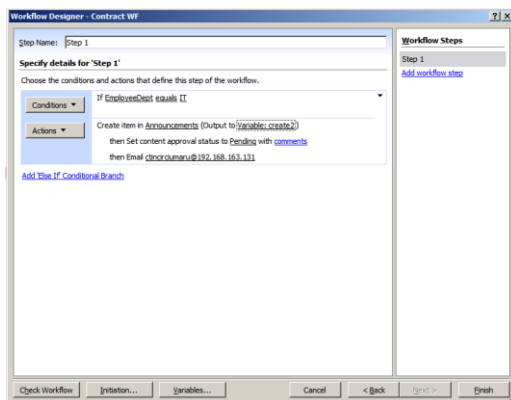


Fig. 4. First step in defining conditions and actions

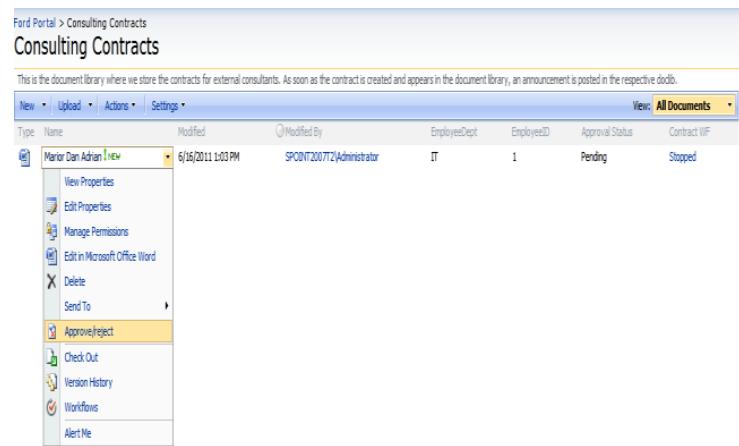


Fig. 5. Approving/ Rejecting the document

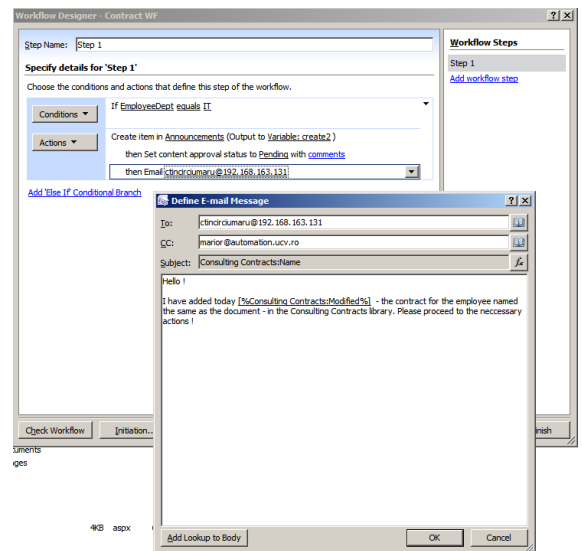


Fig. 6. The variables involved in the algorithm

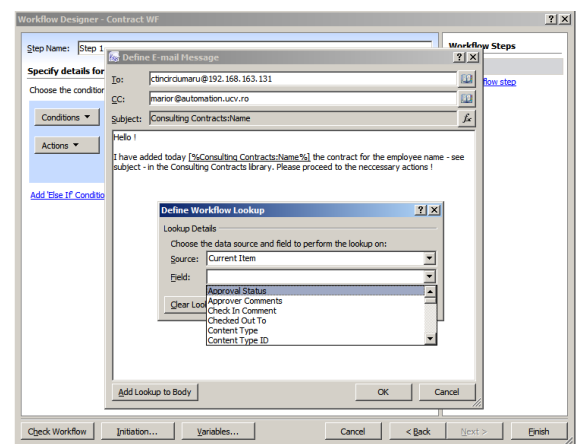


Fig. 7. The “Add Lookup to Body” menu

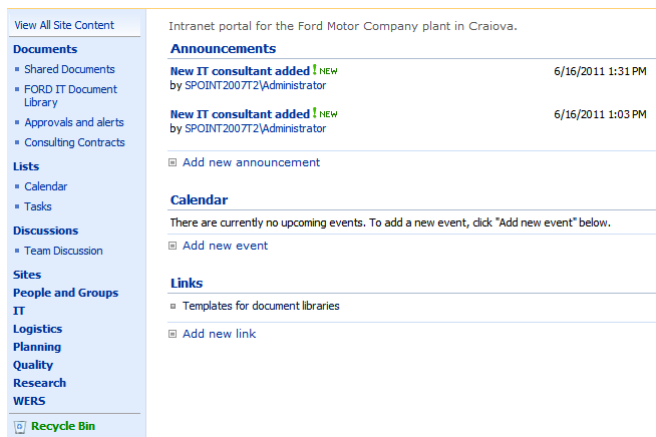


Fig. 8. The “Announcements” list in the front page

Figure 2 shows the options we can set in the first step of building the workflow, such as the list to attach the workflow to and the way to start it (by giving the manager the possibility to start the workflow manually and to have it start automatically upon contract creation we can ensure the workflow also takes into account existing documents, not only the newly created ones).

The moment the document appears in the Contracts library the workflow is set to create a new item in the “Announcements” list, with custom text in it, sets the document status to “Pending” and sends an email to the person responsible for the IT Department. These three actions are all done in the same step; as we can see in the Figure 3 we can add more actions to be executed in the same step or even add more “if” clauses.

Also, it must be noted that the two parts of a workflow definition are the conditions which must be met and the actions to take should the conditions be evaluated to true.

We set the document status to “Pending” because it has to be manually approved by the IT Manager, and he can do that easily using the contextual menu of the document, as in Figure 4. He could find more options useful, for example he can check in/check out the document, he can see the version history and see the workflows attached to the document.

In Figure 5 there is a detailed view on the email to be sent to the manager. The SMTP mail server receives the comand to send the email but just before the text is sent to the server there is some processing to be done by Sharepoint. As it can be seen in the image the variable in the square brackets is replaced by the name of the person who created the contract; the variable or variables which can be added may belong to a lot of categories but are usually chosen from the metadata of the current item (the current contract in this case).

From Figure 6 we can observe how other parts of metadata data can be chosen and introduced in the body of the email. Also we can put variables and fill them with data in the subject of the email, by pressing the button similar to the Excel function builder.

The announcements list is shown in Figure 7, and is

located on the front page of the portal, for easy access to the news and to be easily noticed, of course.

V. CONCLUSION

Although the market for management and document management workflows has a lot of solutions to offer, we chose Sharepoint, or the Microsoft Office Sharepoint Server suite as the best solution for a company looking to use an already “field tested” and widely accepted solution which has proven to be able to withstand high demand workloads with a low fault rate. This choice is limited, however, by the financial and hardware resources. No software suite is perfect, and our intention was not to promote Sharepoint as such, but, all things considered, it does have the appeal of professional Microsoft software combined with features which, at least partially, can find a place in every company. Other worth mentioning alternative tools to implement workflows we could have chosen are Visual Studio Workflow Designer, jBPM from the JBOSS community and a plethora of other Java alternatives, such as the BPM module from the Eclipse project, YAWL, Bonita, ActiveBPEL, Apache Agila, etc. A comparison between all of them is not possible in the present article but more interesting in our opinion would be the justification of using Visual Studio or Sharepoint Designer for implementing a workflow. The Visual Studio workflow designer allows for the highest degree of complexity in workflow, by using it we can achieve flexibility the Sharepoint Designer was not meant for. For example, workflow templates can be associated to a content type, workflow modifications are possible, solutions can be packaged as features and deployed into the Sharepoint sites, readily available debugging, both sequential and state workflows and others are all possible to implement only in Visual Studio. However, we chose the Sharepoint Designer because of the ease of use and the fact that the workflow we implemented did not require such important facilities; also the background of using and working with .NET solutions over the years was crucial in taking the decision. For the possible future trends of this research we mention modifying the application to host interdepartmental workflows and some Business Intelligence features such as Key Performance Indicators.

ACKNOWLEDGMENT

This work was supported (in part) by the grant no. POSDRU/6/1.5/S/14b.

REFERENCES

- [1] B. Scott and J. Hascall, “*Transitions and Partnerships: The Tao of Internal and External Consulting*”, *OD Practitioner*, vol. 33, no. 3, 2001.
- [2] D. A. Marior, and C. Circumaru, “*Workflow for the contract employees of a company*”, *E-Journal of Doctoral Research*, Vol. 1, No. 1, 2011.
- [3] D. Mann, “*Workflow in the 2007 Microsoft Office System*”, Apress, 2007.

- [4] <http://www.businessdictionary.com/definition/workflow.html>
- [5] P. Coventry, "*Sharepoint Designer Step by Step*", Microsoft Press, 2008
- [6] Barrett, D. and Kipper, G., "*Virtualization and Forensics – a digital forensic investigator's guide to virtual environments*", Syngress, 2010.
- [7] T. Zachry, B.A. McCollum, "*Constructing Online Workspaces for Collaboration: An Experience with Two Cases of Contrasting Systems*", IEEE International Communication Conference, IPCC, 1-3 oct. 2007.

On Some Properties of Diagonally Stable Polytopic Systems

Mihaela-Hanako Matcovschi, Octavian Pastravanu, and Mihail Voicu

Abstract—For polytopic matrices and polytopic systems we introduce the concept of diagonal stability with respect to a p -norm, $1 \leq p \leq \infty$. We address both the discrete-time and the continuous-time cases. We prove the equivalence between this algebraic concept and the following properties of uncertain linear dynamics described by polytopic models: • the existence of diagonal Lyapunov functions defined by weighted p -norm; • the existence of contractive sets shaped by weighted p -norm, which are invariant with respect to the system trajectories. We also discuss the problem of the best rate for the Lyapunov function decrease and, respectively, for the invariant set contraction. A numerical procedure has been developed to support this exploration.

I. INTRODUCTION

STARTING from the concept of “diagonal stability relative to a Hölder p -norm, $1 \leq p \leq \infty$,” defined and characterized by [1] for real square matrices, the paper investigates some properties related to this concept applied to polytopic matrices and polytopic systems. We address both the discrete-time and the continuous-time cases. Our investigation shows that a series of results formulated for single matrices and single matrix systems in [1] can be extended such that the generalized forms can cover the more complex situations corresponding to polytopic descriptions.

The paper is organized as follows. Section 2 presents a preparatory material that includes • the definitions of the key notions (polytopic matrices and systems, diagonal stability of polytopic matrices and systems relative to a p -norm); • some comments on diagonal stability and connections to the literature; • a brief comparative analysis between diagonal stability and stability. Section 3 exposes our main theoretical results that reveal the equivalence between the diagonal stability relative to the p -norm and two properties of polytopic-system dynamics, namely: • the existence of strong Lyapunov functions with diagonal form expressed by the p -norm, • the existence of contractive sets expressed by the p -norms that are invariant with respect to the system trajectories. Section 4 proposes a numerical method for finding the Lyapunov functions and the invariant contractive sets with the fastest decreasing rates. Section 5 illustrates by a relevant example the applicability of the numerical procedure developed in the previous section. Section 6 points out some conclusions on the importance of our research.

Manuscript received September 1, 2011. The first two authors are grateful for the support of CNMP Romania, Grant 12100/1.10.2008–SICONA.

The authors are with the Department of Automatic Control and Applied Informatics from the Technical University “Gh. Asachi” of Iași, Romania (phone/fax: +40 232 230 751; e-mail: {mhanako, opastrav, mvoicu}@ac.tuiasi.ro).

The notation and nomenclature used in the presentation of our research results are available in Appendix I. The proofs of Theorems 1 and 2 are given in Appendix II in order to ensure a fluent exposition of the main points in exploring the polytopic system properties.

II. PRELIMINARIES

A. Key notions

Definition 1.

Consider the convex hull of real square matrices

$$\mathcal{A} = \left\{ A \in \mathbb{R}^{n \times n} \mid A = \sum_{k=1}^K \gamma_k A_k, \gamma_k \geq 0, \sum_{k=1}^K \gamma_k = 1 \right\}. \quad (1)$$

(a) The convex hull (1) is called a *polytopic matrix*.

(b) The system

$$x(t+1) = Ax(t), \quad A \in \mathcal{A}, \quad x(t_0) = x_0, \quad t, t_0 \in \mathbb{Z}_+, \quad t \geq t_0, \quad (2-S)$$

is called a *discrete-time polytopic system*.

(c) The system

$$\dot{x}(t) = Ax(t), \quad A \in \mathcal{A}, \quad x(t_0) = x_0, \quad t, t_0 \in \mathbb{R}_+, \quad t \geq t_0, \quad (2-H)$$

is called a *continuous-time polytopic system*. ■

The extension S , respectively, H used in equation numbering comes from Schur (associated with discrete time) and, respectively Hurwitz (associated with continuous time).

Polytopic systems present a great interest for applications, since they incorporate information about the uncertainties of physical parameters. In model construction, the use of a polytopic system (2-S), respectively, (2-H) assumes that the entries of matrix A are fixed (not time-varying), but the values of these entries cannot be known as precise numbers.

Definition 2.

Let $1 \leq p \leq \infty$. Let $\mathcal{A}$ be a polytopic matrix.

(a) $\mathcal{A}$ is called *Schur diagonally stable relative to the p -norm* (abbreviated SDS_p) if there exists a positive definite diagonal matrix $D \succ 0$, such that

$$\forall A \in \mathcal{A} : \|A\|_p^D < 1. \quad (3-S)$$

In this case, the polytopic system (2-S) is also called SDS_p .

(b) $\mathcal{A}$ is called *Hurwitz diagonally stable relative to the p -norm* (abbreviated HDS_p) if there exists a positive definite diagonal matrix $D \succ 0$, such that

$$\forall A \in \mathcal{A} : \mu_p^D(A) < 0. \quad (3-H)$$

In this case, the polytopic system (2-H) is also called HDS_p . ■

B. Comments on Definition 2 and connections to literature

In the control literature, the “diagonal stability” in the classical sense is equally used for matrices and linear time-invariant systems; Schur diagonal stability is associated with discrete-time dynamics, whereas Hurwitz diagonal stability is associated with continuous-time dynamics. Therefore the formulation of Definition 2 encompasses both polytopic matrices and polytopic systems.

For a single matrix A , i.e. $A_k = A$, $k=1, \dots, K$, in (1), Definition 2 coincides with Definition 1 in [1]. Paper [1] proposes a generalization called *diagonal stability relative to a Hölder p -norm*, $1 \leq p \leq \infty$, which includes the classical concept of “diagonal stability” [2]–[4] as a particular case corresponding to $p=2$. This generalization considers Stein-type (respectively Lyapunov-type) inequalities with the form (3-S), respectively (3-H), defined for $1 \leq p \leq \infty$, by, which expand, *mutatis mutandis*, the sense of the standard Stein (Lyapunov) inequality (i.e. the solutions of the generalized inequalities of form (3-S), respectively (3-H), are positive definite diagonal matrices).

For a proper polytopic matrix $\mathcal{A}$, Definition 2 represents a non-trivial extension of Definition 1 in [1]. Indeed, the statement “all the matrices $A \in \mathcal{A}$ are SDS_p , respectively HDS_p ” does not necessarily imply “ $\mathcal{A}$ is SDS_p , respectively HDS_p ”, since different matrices $A \in \mathcal{A}$ may be SDS_p , respectively HDS_p , with different diagonal matrices $D \succ 0$.

For a proper polytopic matrix $\mathcal{A}$ and $p=2$, Definition 2 means the condition for *quadratic stability* of $\mathcal{A}$ [5], i.e.

$$\forall A \in \mathcal{A} : A^T P A - P \prec 0 \text{ (Stein inequality),} \quad (4-S)$$

respectively

$$\forall A \in \mathcal{A} : A^T P + P A \prec 0 \text{ (Lyapunov inequality).} \quad (4-H)$$

is satisfied by $P = (D^{-1})^2$.

C. Diagonal stability vs stability

If there exists a p , $1 \leq p \leq \infty$, such that $\mathcal{A}$ is SDS_p , respectively HDS_p , then $\mathcal{A}$ is *Schur stable* (abbreviated SS), respectively *Hurwitz stable* (abbreviated HS). Indeed, for an arbitrary matrix $A \in \mathcal{A}$, Remark 2 in [1], shows the implication “ A is SDS_p , respectively HDS_p , for a p , $1 \leq p \leq \infty$ ” $\Rightarrow$ “ A is SS, respectively HS”.

The converse statement is not true. As commented in the previous subsection, even if each $A \in \mathcal{A}$ is diagonally stable relative to a p -norm, a diagonal matrix $D \succ 0$ that satisfies (3-S) or (3-H) may not exist.

If $\mathcal{A}$ is diagonally stable relative to a given p -norm, then it is not necessarily that $\mathcal{A}$ is diagonally stable relative to another p -norm. We define the next *robustness measures*:

(a) For the discrete-time case

• the SDS_p degree (or margin)

$$\rho_{\text{SDS}_p}(\mathcal{A}) = 1 - \inf_{D \succ 0} \max_{A \in \mathcal{A}} \|A\|_p^D, \quad (5-S)$$

• the SS degree (or margin)

$$\rho_{\text{SS}}(\mathcal{A}) = 1 - \max_{A \in \mathcal{A}} \max_{i=1, \dots, n} |\lambda_i(A)|. \quad (6-S)$$

(b) For the continuous-time case

• the HDS_p degree (or margin)

$$\rho_{\text{HDS}_p}(\mathcal{A}) = \left| \inf_{\substack{D \succ 0 \\ \text{diagonal}}} \max_{A \in \mathcal{A}} \mu_p^D(A) \right|, \quad (5-H)$$

• the HS degree (or margin)

$$\rho_{\text{HS}}(\mathcal{A}) = \left| \max_{A \in \mathcal{A}} \max_{i=1, \dots, n} \text{Re}\{\lambda_i(A)\} \right|. \quad (6-H)$$

In the discrete-time case, the matrix norm properties yield

$$\rho_{\text{SDS}_p}(\mathcal{A}) \leq \rho_{\text{SS}}(\mathcal{A}). \quad (7-S)$$

whereas, in the continuous-time case, we use the matrix measure properties to get

$$\rho_{\text{HDS}_p}(\mathcal{A}) \leq \rho_{\text{HS}}(\mathcal{A}). \quad (7-H)$$

Let $p_1 \neq p_2$, $1 \leq p_1, p_2 \leq \infty$. Assume that $\mathcal{A}$ is diagonally stable relative to both p_1 - and p_2 -norm. In general, we have $\rho_{\text{SDS}_{p_1}}(\mathcal{A}) \neq \rho_{\text{SDS}_{p_2}}(\mathcal{A})$ or $\rho_{\text{HDS}_{p_1}}(\mathcal{A}) \neq \rho_{\text{HDS}_{p_2}}(\mathcal{A})$.

III. QUALITATIVE ANALYSIS RESULTS

Theorem 1.

(a) Consider a discrete-time polytopic system of the form (2-S). Let $1 \leq p \leq \infty$, a diagonal matrix $D \succ 0$ and a constant $0 < r < 1$. The following statements are equivalent:

$$(i) \forall k=1, \dots, K : \|A_k\|_p^D \leq r. \quad (8-S)$$

(ii) For the polytopic system (2-S), the function

$$\mathcal{V}_p : \mathbb{R}^n \rightarrow \mathbb{R}_+, \mathcal{V}_p(x) = \|x\|_p^D, \quad (9-S)$$

is a strong diagonal Lyapunov function, with the decreasing rate r , i.e.

$$\forall \text{ solution } x(t) \text{ to (2-S), } \forall t \in \mathbb{Z}_+, t \geq t_0 : \mathcal{V}_p(x(t+1)) \leq r \mathcal{V}_p(x(t)). \quad (10-S)$$

(iii) The contractive sets

$$\mathcal{X}_p^\varepsilon(t; t_0) = \{x \in \mathbb{R}^n \mid \|x\|_p^D \leq \varepsilon r^{(t-t_0)}\}, \quad t, t_0 \in \mathbb{Z}_+, t \geq t_0, \varepsilon > 0, \quad (11-S)$$

are invariant with respect to the solutions (state trajectories) of the polytopic system (2-S), i.e.

$$\forall \varepsilon > 0, \forall t, t_0 \in \mathbb{Z}_+, t \geq t_0, \forall x_0 = x(t_0) \in \mathbb{R}^n : \|x_0\|_p^D \leq \varepsilon \Rightarrow \|x(t, t_0, x_0)\|_p^D \leq \varepsilon r^{(t-t_0)}. \quad (12-S)$$

(b) Consider a continuous-time polytopic system of the form (2-H). Let $1 \leq p \leq \infty$, a diagonal matrix $D \succ 0$ and a constant $r < 0$. The following statements are equivalent:

$$(i) \forall k=1, \dots, K: \mu_p^D(A_k) \leq r \quad (8-H)$$

(ii) For the polytopic system (2-H), the function

$$\mathcal{V}_p: \mathbb{R}^n \rightarrow \mathbb{R}_+, \mathcal{V}_p(\mathbf{x}) = \|\mathbf{x}\|_p^D, \quad (9-H)$$

is a strong diagonal Lyapunov function, with the decreasing rate r , i.e.

$\forall$ solution $\mathbf{x}(t)$ to (2-H), $\forall t \in \mathbb{R}_+, t \geq t_0$:

$$D^+ \mathcal{V}_p(\mathbf{x}(t)) = \lim_{h \downarrow 0} \frac{1}{h} [\mathcal{V}_p(\mathbf{x}(t+h)) - \mathcal{V}_p(\mathbf{x}(t))] \leq r \mathcal{V}_p(\mathbf{x}(t)). \quad (10-H)$$

(iii) The contractive sets

$$\mathcal{X}_p^\varepsilon(t, t_0) = \left\{ \mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\|_p^D \leq \varepsilon e^{r(t-t_0)} \right\}, \quad (11-H)$$

are invariant with respect to the solutions (state trajectories) of the polytopic system (2-H), i.e.

$$\forall \varepsilon > 0, \forall t, t_0 \in \mathbb{R}_+, t \geq t_0, \forall \mathbf{x}_0 = \mathbf{x}(t_0) \in \mathbb{R}^n: \|\mathbf{x}_0\|_p^D \leq \varepsilon \Rightarrow \|\mathbf{x}(t; t_0, \mathbf{x}_0)\|_p^D \leq \varepsilon e^{r(t-t_0)}. \quad (12-H)$$

Proof: See Appendix II. ■

The constant $0 < r < 1$, respectively $r < 0$, in the statement of Theorem 1(a), respectively Theorem 1(b), represents a decreasing rate for the diagonal Lyapunov functions and for the contractive invariant sets. From the practical point of view, for a given p , $1 \leq p \leq \infty$, we are interested in finding the smallest value of r accepted by (8-S) or (8-H) in Theorem 1, which we denote by $(r_p^S)^*$ or $(r_p^H)^*$, respectively. Obviously, the value $(r_p^S)^*$ or $(r_p^H)^*$ ensures the *fastest decreasing rate* for the diagonal Lyapunov functions and for the contractive invariant sets.

Theorem 2.

Let $1 \leq p \leq \infty$.

(a) For a discrete-time polytopic system of the form (2-S) that is diagonally stable relative to the p -norm, we have:

$$\sigma_p^S = 1 - \rho_{\text{SDS}_p}(\mathcal{A}) = \inf_{\substack{D > 0 \\ \text{diagonal}}} \max_{k=1, \dots, K} \|A_k\|_p^D. \quad (13-S)$$

(b) For a continuous-time polytopic system of the form (2-H) that is diagonally stable relative to the p -norm, we have:

$$\sigma_p^H = -\rho_{\text{HDS}_p}(\mathcal{A}) = \inf_{\substack{D > 0 \\ \text{diagonal}}} \max_{k=1, \dots, K} \mu_p^D(A_k) \quad (13-H)$$

Proof: See Appendix II. ■

Let $p_1 \neq p_2$, $1 \leq p_1, p_2 \leq \infty$. Assume that a polytopic system of the form (2-S) or (2-H) is diagonally stable relative to both p_1 - and p_2 -norm. In general, the fastest decreasing rates are different, i.e. $\sigma_{p_1}^S \neq \sigma_{p_2}^S$ or $\sigma_{p_1}^H \neq \sigma_{p_2}^H$.

IV. NUMERICAL TRACTABILITY OF THEOREM 2

Consider a discrete-time polytopic system of the form (2-S), or a continuous-time polytopic system of the form (2-H). We are interested in developing a numerical procedure that provides an estimation of σ_p^S or σ_p^H for the particular norms corresponding to $p \in \{1, 2, \infty\}$.

A. Approach as a constrained minimization problem

We consider the function

$$\begin{aligned} \varphi_p^S: (\mathbb{R}_+^*)^n &\rightarrow \mathbb{R}, \quad \varphi_p^S(d_1, \dots, d_n) = \max_{k=1, \dots, K} \|A_k\|_p^D = \\ &= \max_{k=1, \dots, K} \left\| \text{diag} \left\{ \frac{1}{d_1}, \dots, \frac{1}{d_n} \right\} A_k \text{diag} \{d_1, \dots, d_n\} \right\|_p \end{aligned} \quad (14-S)$$

used by (13-S), and the function

$$\begin{aligned} \varphi_p^H: (\mathbb{R}_+^*)^n &\rightarrow \mathbb{R}, \quad \varphi_p^H(d_1, \dots, d_n) = \max_{k=1, \dots, K} \mu_p^D(A_k) = \\ &= \max_{k=1, \dots, K} \mu_p \left(\text{diag} \left\{ \frac{1}{d_1}, \dots, \frac{1}{d_n} \right\} A_k \text{diag} \{d_1, \dots, d_n\} \right) \end{aligned} \quad (14-H)$$

used by (13-H), where $D = \text{diag}\{d_1, \dots, d_n\}$, that allow writing inequalities (8-S) and (8-H) in the equivalent forms

$$\varphi_p^S(d_1, \dots, d_n) \leq r, \quad (15-S)$$

and, respectively,

$$\varphi_p^H(d_1, \dots, d_n) \leq r. \quad (15-H)$$

Let $\Delta \subset (\mathbb{R}_+^*)^n$ be a compact rectangular set, $\Delta = \{\mathbf{d} \in \mathbb{R}^n \mid 0 < d_{\min} \leq \mathbf{d} \leq d_{\max}\}$.

We can organize a search for a positive-definite diagonal matrix $D = \text{diag}\{d_1, \dots, d_n\}$, with the diagonal entries belonging to closed intervals, which minimizes the function (14-S) or (14-H). Explicitly we look for a solution to the minimization of (14-S) or (14-H) subject to the constraints

$$\mathbf{d} = [d_1 \dots d_n]^T \in \Delta = \{\mathbf{d} \in \mathbb{R}^n \mid 0 < d_{\min} \leq \mathbf{d} \leq d_{\max}\}. \quad (16)$$

Given a compact rectangular set $\Delta \subset (\mathbb{R}_+^*)^n$, since functions (14-S) and (14-H) are continuous on compact sets, each of them reaches its minimum value denoted by $\xi_p^S(\Delta) = \min_{[d_1, \dots, d_n]^T \in \Delta} \varphi_p^S(d_1, \dots, d_n)$ and, respectively, $\xi_p^H(\Delta) = \min_{[d_1, \dots, d_n]^T \in \Delta} \varphi_p^H(d_1, \dots, d_n)$, for some $\mathbf{d}^* = [d_1^* \dots d_n^*]^T \in \Delta$, meaning that the proposed constrained minimization has feasible solutions. From the properties of matrix norms and matrix measures, we have that

$$\max_{k=1, \dots, K} \left[\max_{i=1, \dots, n} |\lambda_i(A_k)| \right]^{not} = \alpha^S \leq \sigma_p^S \leq \xi_p^S(\Delta), \quad (17-S)$$

and, respectively,

$$\max_{k=1, \dots, K} \left[\max_{i=1, \dots, n} \text{Re} \{\lambda_i(A_k)\} \right]^{not} = \alpha^H \leq \sigma_p^H \leq \xi_p^H(\Delta). \quad (17-H)$$

For an adequate choice of $\Delta \subset (\mathbb{R}_+^*)^n$ (large enough), the minimum value $\xi_p^S(\Delta)$ will estimate σ_p^S with any required precision, and similarly for $\xi_p^H(\Delta)$ and σ_p^H .

From the numerical point of view, the minimization of the objective function (14-S) or (14-H) constrained by (16) is a delicate problem, since the objective function is nonlinear and nonsmooth, and, moreover, we are looking for a global minimum on the considered compact set. Even if some solvers might return accurate solutions, we prefer to avoid this approach by formulating a more robust strategy for the estimation of the fastest decreasing rates.

B. Numerical procedure that estimates σ_p^S or σ_p^H

For brevity reason, we give only the approach to σ_p^S ; the case of σ_p^H can be addressed similarly (*mutatis mutandis*).

The numerical approach is iterative and is based on the following premise.

Hypothesis: $\alpha^S < 1$ (i.e. all matrices A_k , $k=1, \dots, K$, are SS) and the rectangular compact set $\Delta \subset (\mathbb{R}_+^*)^n$ is large enough to ensure that $\xi_p^S(\Delta)$ is very close to σ_p^S . ♦

Key points of the numerical procedure

We start by looking for a solution to (15-S) corresponding to $r = 1$.

(i). If there exists no $[d_1, \dots, d_n]^T \in \Delta$ such that $\varphi_p^S(d_1, \dots, d_n) \leq 1$, then we have $1 < \xi_p^S(\Delta)$ and we can conclude that the considered system is not diagonally stable relative to the p -norm. It is worth noticing that, for a compact set $\Delta \subset (\mathbb{R}_+^*)^n$ large enough (as assumed by our hypothesis), σ_p^S is very unlikely to satisfy the inequality $\sigma_p^S < 1 < \xi_p^S(\Delta)$. In other words, the conclusion based on $\xi_p^S(\Delta)$ (instead of the precise value σ_p^S) is true, generally speaking, with extremely rare exceptions.

(ii). If there exists $d = [d_1 \dots d_n]^T \in \Delta$ such that $\varphi_p^S(d_1, \dots, d_n) \leq 1$, then we iteratively apply bisection to the interval $[\alpha^S, 1]$ and construct a strictly decreasing and bounded sequence

$$1 > r_1^S > r_2^S > r_3^S > \dots > r_h^S > \dots > \alpha^S \quad (18-S)$$

for which each inequality $\varphi_p^S(d_1, \dots, d_n) \leq r_h^S$ has a solution $d_h = [d_1^h \dots d_n^h]^T \in \Delta$. Obviously the sequence r_h^S converges to $\xi_p^S(\Delta)$. Thus we can find a subscript N such that r_N^S approximates $\xi_p^S(\Delta)$ as accurately as we want. Subsequently we get a good estimation of σ_p^S . If necessary, the precision

in estimating the fastest decreasing rate σ_p^S may be improved by enlarging the compact set $\Delta \subset (\mathbb{R}_+^*)^n$.

The algorithm for the estimation of σ_p^S is presented in Table 1, where we preserve the notations introduced above, but we drop the superscript S in order to simplify the writing.

Table 1. Algorithm for the estimation of σ_p^S

Step 0. (Initialization) For $h = 0$, give the precision $\varepsilon > 0$ and the maximum number of iteration $N \in \mathbb{N}^*$. Set

$$lb_h = \alpha, \quad ub_h = 2 - \alpha, \quad d_h = 0.$$

Step 1. (Stopping criteria)

1.1. If $ub_h - lb_h < \varepsilon$ or $lb_h = 1$, go to Step 6.

1.2. If $h = N$, go to Step 6.

Step 2. (Check the compatibility of (15-S))
Compute

$$r_h = \frac{1}{2}(lb_h + ub_h)$$

Look for a solution to the inequality

$$\varphi_p(d) \leq r_h.$$

If solution d_h is found, go to Step 3.

If no solution is found, go to Step 4.

Step 3. (Case I. (15-S) is compatible) Set

$$lb_{h+1} = lb_h, \quad ub_{h+1} = \varphi_p(d_h).$$

Go to Step 5.

Step 4. (Case II. (15-S) is incompatible) Set

$$lb_{h+1} = r_h, \quad ub_{h+1} = ub_h.$$

Go to Step 5.

Step 5. (Cycling) Set $d_{h+1} = d_h$, $h := h + 1$ and go to Step 1.

Step 6. (Finalization) Return d_h .

Note that in the initialization step the upper bound ub_0 has been set so that to obtain $r_0 = 1$. If the system under study is not SDS_p , then the algorithm will end with $lb_1 = 1$ returning $d_1 = 0$. If the system is SDS_p , then the algorithm will end either when the imposed precision ($ub_h - lb_h < \varepsilon$) or the maximum number of iterations ($h = N$) has been reached.

Practical considerations on checking the compatibility of (15-S)

In practice, when building the sequence (18-S), the search for a solution $d = [d_1, \dots, d_n]^T \in \Delta$ to an inequality of the form (15-S) (with imposed r) is replaced by an equivalent problem, namely the search for solutions $[d_1, \dots, d_n]^T \in \Delta$ to the set of inequalities

$$\left\| \text{diag}\left\{\frac{1}{d_1}, \dots, \frac{1}{d_n}\right\} A_k \text{diag}\{d_1, \dots, d_n\} \right\|_p \leq r, \quad k=1, \dots, K. \quad (19-S)$$

which represent the explicit form of (8-S).

For $p \in \{1, 2, \infty\}$ and known r , the inequalities (18-S) are equivalent to a set of linear inequalities for $p \in \{1, \infty\}$ and, respectively, to a linear matrix inequality (LMI) for $p = 2$, as detailed in the sequel:

- for $p = 1$, inequalities (19-S) are equivalent to

$$(A_k^S)^T \delta \leq r \delta, \quad k=1, \dots, K, \quad (20-S)$$

where $\delta = [\delta_1 \dots \delta_n]^T \in \mathbb{R}^n$ is the positive vector formed with the inverses of the elements of $d = [d_1, \dots, d_n]^T$, i.e.

$$\delta = [\delta_1 \dots \delta_n]^T \in \mathbb{R}^n, \quad \delta_i = 1/d_i, \quad i=1, \dots, n, \quad (21)$$

and $A_k^S \in \mathbb{R}^{n \times n}$ are nonnegative matrices defined in accordance with the notation M^S in Appendix I.

- for $p = 2$, inequalities (19-S) are equivalent to

$$A_k^T P A_k - r^2 P \preceq 0, \quad k=1, \dots, K, \quad (22-S)$$

where $P = (D^{-1})^2$ with $D = \text{diag}\{d_1, \dots, d_n\}$;

- for $p = \infty$, inequalities (19-S) are equivalent to

$$A_k^S d \leq r d, \quad k=1, \dots, K. \quad (23-S)$$

Inequalities (20-S), (22-S) or (23-S), equivalent to (19-S), are easy to handle in a software environment for scientific computation (e.g. Matlab). Explicitly, in checking the compatibility of (15-S) these equivalent forms are used as constraints in the minimization problem

$$\begin{aligned} & \text{minimize } J(d) \\ & \text{subject to} \\ & (19-S) \text{ and } d \in \Delta \end{aligned} \quad (24-S)$$

where

$$J : (\mathbb{R}_+^*)^n \rightarrow \mathbb{R}, \quad J(d) = 0. \quad (25)$$

The solution to problem (24-S) for $p \in \{1, \infty\}$ can be found through a call of the **linprog** function from the MATLAB Optimization Toolbox, whereas in the case when $p = 2$, the **solvesdp** function from the freely available Multi Parametric Toolbox for MATLAB [8].

V. ILLUSTRATIVE EXAMPLE

For the particular norms corresponding to $p \in \{1, 2, \infty\}$, we are interested in studying the HDS_p of the continuous-time polytopic system modeled by (2-H) where the convex hull $\mathcal{A}$ is given by (1) from the set of vertices

$$\begin{aligned} A_1 &= \begin{bmatrix} -4 & 1 & -1 \\ -1 & -5 & 3 \\ -2 & -2 & -6 \end{bmatrix}, \quad A_2 = \begin{bmatrix} -5 & -1 & 1 \\ -1 & -6 & -2 \\ -3 & -1 & -5 \end{bmatrix}, \\ A_3 &= \begin{bmatrix} -6 & -2 & 2 \\ 1 & -5 & -2 \\ -2 & 1 & -4 \end{bmatrix}. \end{aligned} \quad (26)$$

We adapt the algorithm discussed in Section IV for σ_p^S in order to find an estimation of σ_p^H and start by testing the hypothesis. Since $\alpha^H = -4.3026 < 0$, it makes sense to further investigate the HDS_p property.

We choose the compact set $\Delta \subset (\mathbb{R}_+^*)^3$ in which we look for solutions to (15-H) as defined by (16) with $d_{\min} = [1 \ 1 \ 1]^T$ and $d_{\max} = 10000 \cdot d_{\min}$.

We initialize the algorithm with the precision $\varepsilon = 0.0001$ and the maximum number of iterations $N = 20$.

For $p = 1$ after 13 iterations we get $\xi_1^H(\Delta) = -0.8377$, for $D_1 = \text{diag}\{1.1269 \ 1.4415 \ 1.000\}$.

For $p = 2$ after 8 iterations we get $\xi_2^H(\Delta) = -3.2644$, for $D_2 = \text{diag}\{1.0000 \ 1.3091 \ 1.3088\}$.

For $p = \infty$ after 15 iterations we get $\xi_\infty^H(\Delta) = -1.2572$ for $D_\infty = \text{diag}\{1.0000 \ 1.2033 \ 1.1679\}$.

Thus, we conclude that the considered system is HDS_p for $p \in \{1, 2, \infty\}$. The diagonal matrices given above can be used, for example, in the construction of contractive sets $\mathcal{X}_p^\varepsilon(t; t_0)$ (11-H) that are invariant with respect to the solutions (state trajectories) of the polytopic system.

VI. CONCLUSIONS

The concept of diagonal stability relative to a p -norm introduced by the authors has obvious roots within the matrix algebra researches of the decade 1990-2000. The progress brought by our work targets two types of objectives, as follows:

- (1) We expand the notional area of “diagonal stability” in the field of algebra by referring to polytopic matrices. We show that this extension is far from triviality. We comment on some issues in the already existing literature that vaguely anticipated the possibility of a generalization in the sense firmly formulated by our paper.
- (2) We reveal the connections between the algebraic context at (1) and the qualitative analysis of linear dynamics with uncertain parameters described by polytopic systems. We prove that the diagonal stability relative to a given p -norm is equivalent to the existence of contractive sets invariant with respect to the polytopic system trajectories. Then we focus on the identification of the invariant sets exhibiting the fastest decreasing rate and we develop a procedure for the numerical tractability of this problem.

REFERENCES

- [1] O. Pastravanu, and M. Voicu (2006). Generalized matrix diagonal stability and linear dynamical systems. *Linear Algebra and its Applications*, vol. 419, no. 1, pp. 299-310.
- [2] G.P. Barker, A. Berman, and R.J. Plemmons (1978). Positive diagonal solutions to the Lyapunov equations, *Linear and Multilinear Algebra*, vol. 5, pp. 249-256.
- [3] A. Berman, and D. Hershkowitz (1983). Matrix diagonal stability and its implications. *SIAM Journal on Algebraic and Discrete Methods*, vol. 4, pp. 377-382.

- [4] E. Kaszkurewicz, and A. Bhaya (2000). *Matrix Diagonal Stability in Systems and Computation*, Birkhauser: Boston.
- [5] F. Blanchini, and S. Miani (2008). *Set-Theoretic Methods in Control*, Birkhäuser: Boston, Basel, Berlin.
- [6] Y. Mori, T. Mori, and Y. Kuroe (1998). On a Class of Linear Constant Systems Which Have a Common Quadratic Lyapunov Function, *Proc. 37th IEEE Conf. Decision & Control 1998*, pp. 2808-2809.
- [7] W.A. Coppel (1965). *Stability and Asymptotic Behavior of Differential Equations*, D.C. Heath and Company: Boston.
- [8] M. Kvasnica, *Multi Parametric Toolbox (MPT)*, <http://control.ee.ethz.ch/~mpt/>, ETH Zurich, 2007.

APPENDIX I. NOTATION AND NOMENCLATURE

• For a vector $\mathbf{x}=[x_1 \dots x_n]^T \in \mathbb{R}^n$: $\|\mathbf{x}\|_p$ is the Hölder vector p -norm defined by $\|\mathbf{x}\|_p = [|x_1|^p + \dots + |x_n|^p]^{1/p}$ for $1 \leq p < \infty$, and by $\|\mathbf{x}\|_\infty = \max_{1 \leq i \leq n} |x_i|$ for $p = \infty$.

• For a square matrix $\mathbf{M}=[m_{ij}] \in \mathbb{R}^{n \times n}$: for $1 \leq p \leq \infty$, $\|\mathbf{M}\|_p$ is the matrix norm induced by the Hölder vector p -norm through $\|\mathbf{M}\|_p = \sup_{\mathbf{x} \neq 0} \frac{\|\mathbf{M}\mathbf{x}\|_p}{\|\mathbf{x}\|_p} = \max_{\|\mathbf{x}\|_p=1} \|\mathbf{M}\mathbf{x}\|_p$; $\mu_p(\mathbf{M}) = \lim_{h \downarrow 0} h^{-1}(\|\mathbf{I} + h\mathbf{M}\|_p - 1)$ is the matrix measure ([7], pp. 41) associated to the matrix norm $\|\bullet\|_p$.

For all $1 \leq p \leq \infty$, a positive definite diagonal matrix $\mathbf{D} = \text{diag}\{d_1, \dots, d_n\}$, $d_i > 0$, $i=1, \dots, n$, can be used for defining the *weighted* vector norm $\|\mathbf{x}\|_p^D = \|\mathbf{D}^{-1}\mathbf{x}\|_p$, the induced matrix norm $\|\mathbf{M}\|_p^D = \|\mathbf{D}^{-1}\mathbf{M}\mathbf{D}\|_p$, and the matrix measure $\mu_p^D(\mathbf{M}) = \lim_{h \downarrow 0} h^{-1}(\|\mathbf{I} + h\mathbf{M}\|_p^D - 1)$.

For $\mathbf{M} \in \mathbb{R}^{n \times n}$, let $\Sigma(\mathbf{M}) = \{z \in \mathbb{C} \mid \det(z\mathbf{I} - \mathbf{M}) = 0\}$ be the spectrum of $\mathbf{M}$ and denote by $\lambda_i(\mathbf{M}) \in \Sigma(\mathbf{M})$, $i=1, \dots, n$, its eigenvalues. In the complex plane $\mathbb{C}$, define the regions $\mathbb{C}_S = \{z \in \mathbb{C} \mid |z| < 1\}$ and $\mathbb{C}_H = \{z \in \mathbb{C} \mid \text{Re } z < 0\}$. If $\Sigma(\mathbf{M}) \subset \mathbb{C}_S$, then matrix $\mathbf{M}$ is said to be *Schur stable* (abbreviated SS). If $\Sigma(\mathbf{M}) \subset \mathbb{C}_H$, then matrix $\mathbf{M}$ is said to be *Hurwitz stable* (abbreviated HS).

If $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, then “ $\mathbf{x} \leq \mathbf{y}$ ” and “ $\mathbf{x} < \mathbf{y}$ ” mean componentwise inequalities. If $\mathbf{M}, \mathbf{P} \in \mathbb{R}^{n \times n}$, then “ $\mathbf{M} \leq \mathbf{P}$ ”, “ $\mathbf{M} < \mathbf{P}$ ” mean componentwise inequalities. $|\mathbf{M}|$ denotes the matrix built with the absolute values of the entries of $\mathbf{M}$.

If $\mathbf{M}=[m_{ij}] \in \mathbb{R}^{n \times n}$, then $\mathbf{M}^S \in \mathbb{R}^{n \times n}$ (S super-script from Schur) denotes the nonnegative matrix defined by $\mathbf{M}^S = |\mathbf{M}|$. $\mathbf{M}^H \in \mathbb{R}^{n \times n}$ (H super-script from Hurwitz) denotes the essentially nonnegative matrix defined by $\mathbf{M}^H = \mathbf{M}^d + |\mathbf{M}^o|$, where $\mathbf{M}^d = \text{diag}\{m_{11}, \dots, m_{nn}\}$ and $\mathbf{M}^o = \mathbf{M} - \mathbf{M}^d$.

Let $\mathbb{R}_+^* = \{x \in \mathbb{R} \mid x > 0\}$.

APPENDIX II. PROOFS OF THE MAIN RESULTS

Proof of Theorem 1

(a) Inequality (8-S) is equivalent to the statement $\forall \mathbf{A} \in \mathcal{A} : \|\mathbf{A}\|_p^D \leq r$. Indeed, if (8-S) is true, then we have

$$\forall \mathbf{A} \in \mathcal{A} : \mathbf{A} = \sum_{k=1}^K \gamma_k \mathbf{A}_k, \quad 0 \leq \gamma_k \leq 1, \quad k=1, \dots, K, \quad \sum_{k=1}^K \gamma_k = 1 \Rightarrow \|\mathbf{A}\|_p^D \leq \sum_{k=1}^K \gamma_k \|\mathbf{A}_k\|_p^D \leq \left(\sum_{k=1}^K \gamma_k \right) r = r.$$

The converse statement is obvious since the vertex matrices belong to the polytopic matrix $\mathcal{A}$.

Then we apply the discrete-time case of Theorem 2 in [1] to all matrices $\mathbf{A}$ in $\mathcal{A}$ and we get the equivalence (8-S) $\Leftrightarrow$ (10-S) $\Leftrightarrow$ (12-S).

(b) Inequality (8-H) is equivalent to the statement $\forall \mathbf{A} \in \mathcal{A} : \mu_p^D(\mathbf{A}) \leq r$. Indeed, if (8-H) is true, then we have

$$\forall \mathbf{A} \in \mathcal{A} : \mathbf{A} = \sum_{k=1}^K \gamma_k \mathbf{A}_k, \quad 0 \leq \gamma_k \leq 1, \quad k=1, \dots, K, \quad \sum_{k=1}^K \gamma_k = 1 \Rightarrow \mu_p^D(\mathbf{A}) \leq \sum_{k=1}^K \gamma_k \mu_p^D(\mathbf{A}_k) \leq \left(\sum_{k=1}^K \gamma_k \right) r = r.$$

The converse statement is obvious since the vertex matrices belong to the polytopic matrix $\mathcal{A}$.

Then we apply the continuous-time case of Theorem 2 in [1] to all matrices $\mathbf{A}$ in $\mathcal{A}$ and we get the equivalence (8-H) $\Leftrightarrow$ (10-H) $\Leftrightarrow$ (12-H). ■

Proof of Theorem 2.

(a) For any diagonal matrix $\mathbf{D} \succ 0$, the value $\max_{\mathbf{A} \in \mathcal{A}} \|\mathbf{A}\|_p^D$ is obtained for a set of coefficients γ_k^* , $k=1, \dots, K$, such that we can write

$$\max_{\mathbf{A} \in \mathcal{A}} \|\mathbf{A}\|_p^D = \left\| \sum_{k=1}^K \gamma_k^* \mathbf{A}_k \right\|_p^D \leq \left(\sum_{k=1}^K \gamma_k^* \right) \max_{k=1, \dots, K} \|\mathbf{A}_k\|_p^D = \max_{k=1, \dots, K} \|\mathbf{A}_k\|_p^D.$$

On the other hand we have the inequality

$$\max_{\mathbf{A} \in \mathcal{A}} \|\mathbf{A}\|_p^D \geq \max_{k=1, \dots, K} \|\mathbf{A}_k\|_p^D.$$

(b) For any diagonal matrix $\mathbf{D} \succ 0$, the value $\max_{\mathbf{A} \in \mathcal{A}} \mu_p^D(\mathbf{A})$ is obtained for a set of coefficients γ_k^* , $k=1, \dots, K$, such that we can write

$$\max_{\mathbf{A} \in \mathcal{A}} \mu_p^D(\mathbf{A}) = \mu_p^D \left(\sum_{k=1}^K \gamma_k^* \mathbf{A}_k \right) \leq \left(\sum_{k=1}^K \gamma_k^* \right) \max_{k=1, \dots, K} \mu_p^D(\mathbf{A}_k) = \max_{k=1, \dots, K} \mu_p^D(\mathbf{A}_k).$$

On the other hand the following inequality is satisfied

$$\max_{\mathbf{A} \in \mathcal{A}} \mu_p^D(\mathbf{A}) \geq \max_{k=1, \dots, K} \mu_p^D(\mathbf{A}_k). \quad \blacksquare$$

Fusion of Supervisors in Discrete Event Systems Using Coloured Petri Nets

F. A. de Alencar Menezes and Giovanni C. Barroso

Abstract—In this paper it is presented mathematical proof of the method called “Control on Decomposed Colors” for definition of supervisors in the context of discrete event systems modeled by coloured Petri nets as an extension of the theorem proposed by Moody and Antsaklis, using Petri nets. As a result, it is proposed to minimization the number of supervisors by the method “Fusion of Controllers”.

Index Terms—Coloured Petri net, supervisory control, discrete event system, fusion of controllers, supervision of coloured Petri nets.

I. INTRODUCTION

Taking advantage of the various qualities of Petri nets (PN), including: visual richness, compact representation of the system and simple formalism, some authors as [1], [2], [3], [4], [5], [6] and, [7], propose studies using this tool in the synthesis of controllers, for the discrete event systems. Thus, following the proposal of [7] for ordinary PN, it is presented an approach using coloured Petri nets (CPN) and, mathematical proof of the method “Control on Decomposed Colors” described in [8] and, as extension it is proposed the method for minimization the number of controllers called, Fusion of Controllers, shown in Theorem 3.

II. METHOD OF PLACE INVARIANTS FOR SYNTHESIS OF SUPERVISORS USING PETRI NETS

The place invariants correspond to the sets of places in which the sum of tokens remains constant for all markings obtainable from the PN. In [7], the controlled PN will have incidence matrix controlled and initial marking controlled, C_c and M_{c_0} , respectively, such that:

$$C_c = -LC \quad (1)$$

$$M_{c_0} = b - LM_0 \quad (2)$$

where:

- L it is the matrix of dimension equal to the number of posts in the uncontrolled PN, where each of its elements is equal to one if the place is subject to any constraints and zero otherwise.
- b it is the matrix whose elements represent the number of tokens required by the constraints.
- C it is the incidence matrix of uncontrolled PN.

Manuscript received May 3, 2011; revised September 9, 2011.

This work was supported by the IEEE.

F. A. de Alencar Menezes is with PPGETI, Ceará Federal University, on leave from Department of Mathematics of Vale do Acaraú State University, Sobral, Ce, Brasil (alencarmenezes@gmail.com).

Giovanni C. Barroso is with Department of Physics, Ceará Federal University, Fortaleza, Ce, Brasil (gcb@fisica.ufc.br)

- M_0 it is the matrix that represents the initial marking of uncontrolled PN.

Theorem 1: (Moody and Antsaklis, 1998) - Given the inequality

$$b - LM_0 \geq 0 \quad (3)$$

then, the PN to be controlled owns the incidence matrix of controller C_c and the initial marking of controller M_{c_0} .

Proof: Assuming (3) is not true, then $LM_0 > b$. If this is the case, then the initial conditions of the PN could violate the constraints, which would be absurd. On the other hand, if (3) is true, then there exists M_{c_0} , such that,

$$b - LM_0 - M_{c_0} = 0 \implies M_{c_0} = b - LM_0 \quad (4)$$

In other words, the initial condition of the controller is defined by a matrix that represents the break in each of the constraints represented by $b \geq LM_0$. Moreover, equation (1) provides an invariant, whereby incurring a closed cycle of the PN. Thus, according to (3),

$$\begin{aligned} b &= LM_0 + M = LM + M_0 \\ &\downarrow \\ b - LM_0 &= M = LM + M_0 - LM_0 \geq 0 \end{aligned} \quad (5)$$

■

III. CONTROL ON DECOMPOSED COLORS

The control on decomposed colors for the synthesis of supervisors in CPN in [8], is developed from the analysis of the incidence matrix of the CPN, along with the decomposition of this matrix in other matrices of coefficients with the same variable. Such decomposition favors the calculation of controllers, enabling it to apply the extension of Theorem 1, presented in Section 2. Thus, it is proposed a solution for the synthesis of supervisors in CPN decomposing the colored tokens in tokens of constraints, which are the tokens that are subject to some form of constraint and complementary tokens of flow, which are the tokens that do not suffer from constraints. It is used the program “CPN Tools” [9] in order to present scenarios that illustrate the method. Thus, given a uncontrolled system, the necessary constraints, and its model in CPN, the steps in order to obtain the supervisor are:

- 1) Construct the matrix D with reference to the uncontrolled CPN, whose elements represent the respective expressions associated with arcs.
- 2) Decompose the matrix D into a sum of matrices where each parcel is a matrix whose elements are relative to the same variable x_n , such that, $D = D^{x_1} + D^{x_2} + \dots + D^{x_n}$.

- 3) Construct the incidence matrix $\mathbf{C}$ of the uncontrolled CPN from matrix $\mathbf{D}$, considering only the incidence coefficients of expressions associated with arcs.
- 4) Decompose the matrix $\mathbf{C}$ into a sum of matrices whose elements (incidence coefficients) are associated with the same variable, taking as reference, the portions of from decomposition of $\mathbf{D}$, such that, $\mathbf{C} = \mathbf{C}^{x_1} + \mathbf{C}^{x_2} + \dots + \mathbf{C}^{x_n}$.
- 5) For each imposed constraints, add the number of tokens corresponding to the initial marking of the other colors (complementary tokens of flow), which are represented by the same variable.
- 6) Calculate the incidence matrices and initial markings of the controller, using an extension of equations of Moody and Antsaklis, modified and adapted to CPN.

A. Initial Marking of Supervisor

In the control on decomposed colors one works with "k" colors and "r" constraints for a given model of CPN. In this case, the inequality (3) it is extended, such that,

$$(b + \hat{b}) - \mathcal{L}\mathbf{M}_0 \geq 0 \quad (6)$$

where, $\hat{b}$ is the matrix of complementary tokens of flow inherent to each constraint imposed upon the system. Thus, it follows that there $\mathbf{M}_{C_0} \geq 0$, such that,

$$\mathbf{M}_{C_0} = (b + \hat{b}) - \mathcal{L}\mathbf{M}_0. \quad (7)$$

Hence, we can define in (7) that:

- $b = (b_i)$ is the $r \times 1$ matrix of constraints imposed on the CPN and b_i is the i th constraint, such that, $1 \leq i \leq r$. If b_i it is not a constraint, then $b_i = 0$.

$$b = (b_i)_{(r \times 1)} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_r \end{pmatrix} \quad (8)$$

- $\hat{b} = (\hat{b}_i)$ is the $r \times 1$ matrix of the sets of the complementary tokens of flow of CPN on b_i and $\hat{b}$ is the i th set of complementary tokens of flow (tokens of different colors) on b_i . If $\hat{b}$ it is relative to $b_i = 0$, then $\hat{b}_i = 0$.

$$\hat{b} = (\hat{b}_i)_{(r \times 1)} = \begin{pmatrix} \hat{b}_1 \\ \hat{b}_2 \\ \vdots \\ \hat{b}_r \end{pmatrix} \quad (9)$$

- $\mathcal{L}$ is the matrix composed of r rows $(L_{1 \times m})_i$ where each $(L_{i,m}) = 1$ defines the m -th place relative to the i -th constraint. If $b_i = 0$ then $(L_{i,m}) = 0$.

$$\mathcal{L} = [(L_{(1 \times m)})_i]_{(r \times m)} = \begin{pmatrix} L_1 \\ L_2 \\ \vdots \\ L_r \end{pmatrix} \quad (10)$$

- $\mathbf{M}_0$ it is the initial marking of CPN.

$$\mathbf{M}_0 = [\mathbf{M}_0(p_j)]_{m \times 1} = \begin{pmatrix} \mathbf{M}_0(p_1) \\ \mathbf{M}_0(p_2) \\ \vdots \\ \mathbf{M}_0(p_m) \end{pmatrix} \quad (11)$$

- $\mathbf{M}_{C_0}$ it is the initial marking of supervisor.

$$\mathbf{M}_{C_0} = [(\mathbf{M}_{C_0})_i]_{r \times 1} = \begin{pmatrix} (\mathbf{M}_{C_0})_1 \\ (\mathbf{M}_{C_0})_2 \\ \vdots \\ (\mathbf{M}_{C_0})_r \end{pmatrix} \quad (12)$$

Note that one can simplify the expression (7) by taking $B = b + \hat{b}$. Thus:

$$\mathbf{M}_{C_0} = \mathbf{B} - \mathcal{L}\mathbf{M}_0 \quad (13)$$

the initial marking of the supervisor by the method. Thus, the matrix $\mathbf{B}$, which presents the constraints b_i coupled to the complementary tokens of flow $\hat{b}_i$, inherent to each, respectively, is given by:

$$\mathbf{B} = (\mathbf{B}_i)_{r \times 1} = \begin{pmatrix} b_1 + \hat{b}_1 \\ b_2 + \hat{b}_2 \\ \vdots \\ b_r + \hat{b}_r \end{pmatrix} \quad (14)$$

Note that each $\hat{b}_i$ is a set of tokens, complement of b_i . Thus:

$$\mathbf{M}_{C_0} = \begin{pmatrix} b_1 + \hat{b}_1 \\ b_2 + \hat{b}_2 \\ \vdots \\ b_r + \hat{b}_r \end{pmatrix} - \begin{pmatrix} L_1 \\ L_2 \\ \vdots \\ L_r \end{pmatrix} \begin{pmatrix} \mathbf{M}_0(p_1) \\ \mathbf{M}_0(p_2) \\ \vdots \\ \mathbf{M}_0(p_m) \end{pmatrix}. \quad (15)$$

It follows that, of 13:

$$\mathbf{M}_{C_0} = \mathbf{B} - \mathcal{L}\mathbf{M}_0 \iff \mathbf{B} - \mathcal{L}\mathbf{M}_0 \geq 0 \quad (16)$$

B. The Proof of Control on Decomposed Colors

The result obtained in the proof of Theorem 2 ensures the achievement of supervisors in accordance with the method described in this section. Thus, considering Theorem 1 as initial hypothesis, one arrives to mathematical argument of the proposal based in the place invariants for CPN.

Theorem 2: Given the expression

$$\mathbf{B} - \mathcal{L}\mathbf{M}_0 \geq 0 \quad (17)$$

the CPN to be controlled will have an initial marking of the supervisor it is given by $\mathbf{M}_{C_0} = \mathbf{B} - \mathcal{L}\mathbf{M}_0$.

Proof: By hypothesis, equation (3) is true. Taking $\hat{b} \geq 0$ (the set of complementary tokens of flow on b), one obtains (6) and (7), preserving the accuracy of expression, which implies (17), from where:

$$\mathbf{B} \geq \mathcal{L}\mathbf{M}_0 \quad (18)$$

In this case, exists $\mathbf{M}_{C_0} \geq 0$, such that,

$$\mathbf{B} = \mathbf{M}_{C_0} + \mathcal{L}\mathbf{M}_0 \quad (19)$$

which implies (13). On the other hand, supposing the absurd that (17) is false, one can say that:

$$\mathbf{B} < \mathcal{L}\mathbf{M}_0 \quad (20)$$

Given that,

$$b + \hat{b} < \mathcal{L}\mathbf{M}_0 \quad (21)$$

which implies that,

$$b < \mathcal{L}\mathbf{M}_0 \quad (22)$$

As $\mathcal{L}\mathbf{M}_0 \geq 0$ one can conclude that:

$$b - \mathcal{L}\mathbf{M}_0 < 0 \quad (23)$$

nonsense, because it contradicts (3), so (16) is true, which confirms the theorem. ■

IV. FUSION OF CONTROLLERS

A. Proof and Contexts of Method

When implementing the control on decomposed colors in a CPN which provides different sets of colors associated with different places and expressions of arcs, composed of different variables, one can unite the specifications without prejudice for the modeled system which carries out the fusion of controllers. This is due to the identification of distinct colors in the process of separation of variables in the decomposition of the incidence matrix. Thus, the matrix $\mathcal{L}$ provides controllers with distinct sets of colors. Thus, within the fusion of these controllers, the resultant controller aggregates all action of incidence on the transitions inherent to the specifications given, whereby we may impose incidence arcs to the same transition, input or output, under the effect of different variables, without risk of blocking. This process it is proved in Theorem 3, which defines the initial marking of fusion $\mathbf{M}_{F0}$ and the incidence matrix of fusion $\mathbf{C}_{CF}$, whereby define the supervisor resulting of whereby define the supervisor resulting of *Fusion of Controllers*.

Theorem 3: Given a CPN with a set of colors represented by distinct variables, B_i and $\mathbf{M}_0$, the restrictions matrix with the relative complementary tokens of flow and the initial marking of the net, - then one can conclude that the initial marking and the incidence matrix of the fusion of supervisors are respectively

$$\mathbf{M}_{F0} = \sum_{i=1}^r \mathbf{B}_i - \mathcal{L}_i \mathbf{M}_0; \quad (24)$$

$$\mathbf{C}_{CF} = - \left(\sum_{i=1}^r \mathcal{L}_i \right) \mathbf{C}. \quad (25)$$

Proof: Let b and $\mathbf{M}_0$, be respectively admissible, then

$$\mathcal{L}\mathbf{M}_0 \leq b \Rightarrow \mathcal{L}\mathbf{M}_0 \leq b + \hat{b} \Rightarrow \mathcal{L}\mathbf{M}_0 \leq \mathbf{B}. \quad (26)$$

According to Theorem 1, without loss of generality, one can consider that

$$\mathcal{L}_1 \mathbf{M}_0 + \mathbf{M}_C = \mathbf{B}_1; \quad (27)$$

$$\mathcal{L}_2 \mathbf{M}_0 + \mathbf{M}'_C = \mathbf{B}_2. \quad (28)$$

Indeed, adding (27) and (28), we then have

$$(\mathcal{L}_1 + \mathcal{L}_2) \mathbf{M}_0 + (\mathbf{M}_C + \mathbf{M}'_C) = (\mathbf{B}_1 + \mathbf{B}_2), \quad (29)$$

which implies,

$$\mathbf{M}_{F0} = (\mathbf{M}_C + \mathbf{M}'_C) = (\mathbf{B}_1 + \mathbf{B}_2) - (\mathcal{L}_1 + \mathcal{L}_2) \mathbf{M}_0. \quad (30)$$

Generalizing for $i = 1 : r$ we have (24).

Furthermore, consider

$$\mathbf{C}_C = - \left(\begin{matrix} \mathcal{L}_1 \\ \mathcal{L}_2 \end{matrix} \right) \mathbf{C}, \quad (31)$$

- the incidence matrix of the supervisors in hypothesis. Within this context, the Petri net has different variables to represent different sets of colors, therefore, each coordinate in the $\mathcal{L}_i$ is distinctly identified. Therefore, with the summation of $\mathcal{L}_i$ defining supervisors can be carried out without conflict. Thus, the incidence matrix of fusion of supervisors is given by

$$\mathbf{C}_{CF} = - (\mathcal{L}_1 + \mathcal{L}_2) \mathbf{C}. \quad (32)$$

In general, for $i = 1 : r$, the expression (25), can be obtained, which proves the theorem. ■

Example 1— Given a CPN with two sets of distinct colors, where the same set of colors, represented by two variables. In this context, the CPN is subjected to the action of the supervisor defined by control places, **pc5** and **pc6** (Fig. 1). Following the process of Theorem 3, the Fusion of Controllers is performed, whereby the , is defined as shown in Fig. 2, which assumes the total controller required.

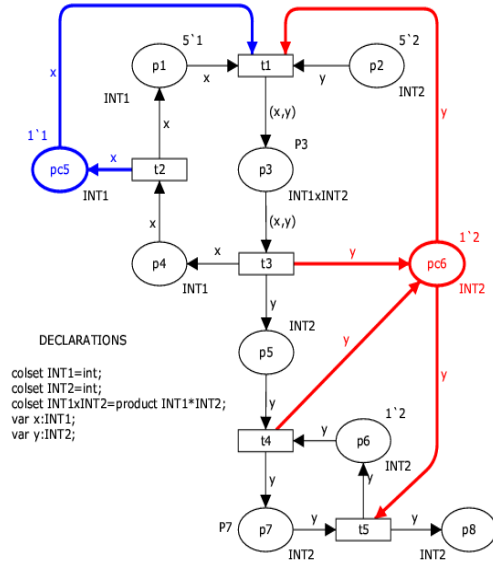


Fig. 1. The Controlled CPN with the same type colors, represented by different variables.

Consider the specifications:

$$\mathbf{M}(p_3) + \mathbf{M}(p_4) \leq 1^1 1; \quad (33)$$

$$\mathbf{M}(p_3) + \mathbf{M}(p_6) \leq 2^1 2. \quad (34)$$

Thus, from the CPN of Fig. 8, we have the matrix $\mathbf{D}$, such that,

$$\mathbf{D} = \begin{pmatrix} -1(x) & 1(x) & 0 & 0 & 0 \\ -1(y) & 0 & 0 & 0 & 0 \\ 1(x,y) & 0 & -1(x,y) & 0 & 0 \\ 0 & -1(x) & 1(x) & 0 & 0 \\ 0 & 0 & 1(y) & -1(y) & 0 \\ 0 & 0 & 0 & -1(y) & 1(y) \\ 0 & 0 & 0 & 1(y) & -1(y) \\ 0 & 0 & 0 & 0 & 1(y) \end{pmatrix}. \quad (35)$$

Therefore, the incidence matrices of the uncontrolled CPN in relation to the variables x and y are:

$$\mathbf{C}^x = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}; \quad (36)$$

$$\mathbf{C}^y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (37)$$

As the initial marking it is:

$$\mathbf{M}_0 = \begin{pmatrix} 5^1 1 \\ 5^1 2 \\ 0 \\ 0 \\ 0 \\ 1^1 2 \\ 0 \\ 0 \end{pmatrix} \quad (38)$$

and the matrix that indicates the control places that suffer constraints it is:

$$\mathcal{L} = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}. \quad (39)$$

The incidence matrix of the desired supervisor it is:

$$\mathbf{C}_C = -\mathcal{L}\mathbf{C} = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & -1 \end{pmatrix}. \quad (40)$$

In the context it is obtained:

$$\mathbf{B} = \begin{pmatrix} 1^1 1 + 0 \\ 2^1 2 + 0 \end{pmatrix}. \quad (41)$$

Note that no addition of complementary tokens of flow relating to the constraints are imposed. This is due to the fact that the uncontrolled CPN of Fig. 1 is able to present tokens that are

of the same type and represented by different variables. Thus, the initial markings of control places **pc5** and **pc6** are:

$$\mathbf{M}_{C_0} = \mathbf{B} - \mathcal{L}\mathbf{M}_0 = \begin{pmatrix} 1^1 1 \\ 2^1 2 \end{pmatrix} - \begin{pmatrix} 0 \\ 1^1 2 \end{pmatrix} = \begin{pmatrix} 1^1 1 \\ 1^1 2 \end{pmatrix} \quad (42)$$

To find the resulting supervisor, defined by the control place **pcr**, Fig. 2, which corresponds to the union of control places **pc5** and **pc6** in Fig. 1, one must establish the union of constraints in determined places, such that:

$$\mathbf{M}(p_3) + \mathbf{M}(p_4) + \mathbf{M}(p_6) \leq 1^1 1 + 2^1 2. \quad (43)$$

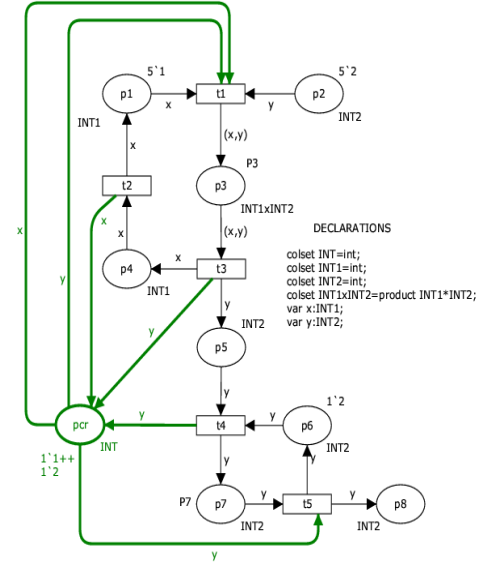


Fig. 2. The CPN controlled by the supervisor **pcr** resulting from fusion of the controllers **pc5** and **pc6**.

In (43), one should not be repeated the place **p3** which is part of (33) and (34). This would be as if there were one more place on the CPN that would bring a repetition of incidence on transition relative to the imposed constraint. Thus according Theorem 3, the incidence matrix of fusion that defines the supervisor **pcr**, from fusion of the controllers **pc5** and **pc6** it is:

$$\mathbf{C}_{CF} = -\left(\sum_{i=1}^r \mathcal{L}_i\right) \mathbf{C} = \begin{pmatrix} -2 & 1 & 1 & 1 & -1 \end{pmatrix}. \quad (44)$$

such that, the initial marking of fusion $\mathbf{M}_{F_0}$ it is:

$$\mathbf{M}_{F_0} = [(1^1 1 + 0) - (2^1 2 + 0)]$$

$$= \begin{pmatrix} 5^1 1 \\ 5^1 2 \\ 0 \\ 0 \\ 0 \\ 1^1 2 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 2 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} = (1^1 1 + 2^1 2) - (1^1 2) = (1^1 1 + 1^1 2) \quad (45)$$

Through (44) one can observe that the supervisor **pcr**, resulting from the fusion, will be the double input to the transition

t1. Obviously, with x and y , there will be an input to the **t5** transition, and subsequently an output to the transitions **t2**, **t3**, and **t4**. The calculation in (45) shows that the initial marking of **pcr** will have a token of color "1" and two tokens of color "2". Thus, the place resulting of fusion as per Fig. 2 it is now the control place that assumes the action of the controllers, represented by the places, **pc5** and **pc6**, in Fig. 1, simultaneously.

Thus, even when the CPN contains different tokens of the same type, the supervisor does not block the CPN because they possess arcs associated with distinct variables.

Example 2—Consider the CPN of Fig. 4, which is controlled by the supervisor **pcr1**, resulting from the fusion of controllers **pc1** and **pc2**, Fig. 3. In this CPN, there is a set of colors associated with a single variable. In this case, the supervisor

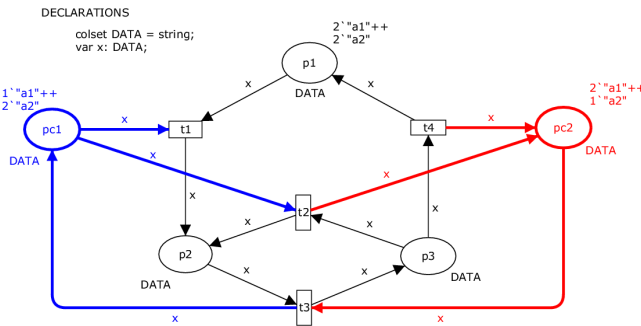


Fig. 3. The colored Petri net with controller places with only one variable associated with arcs.

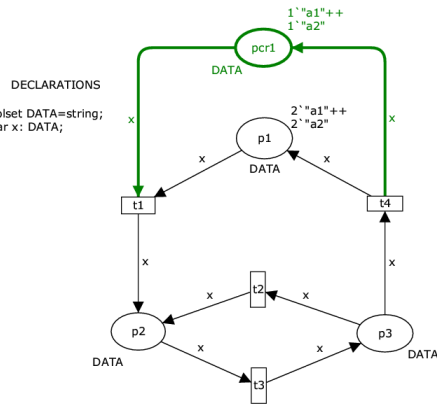


Fig. 4. The CPN controlled by the supervisor **pcr1** resulting from fusion of control places **pc1** and **pc2** referent to Example 1.

pcr1, resulting from fusion, begins to exert a major constraint upon the system, preventing the movement of the complementary tokens of flow in order to avoid conflict caused by the representation of different colors by a single variable and the constraints imposed. In this context, it is therefore necessary to ponder over the application of fusion of controllers when considering costs and benefits of this loss to the modeled system.

Applying Theorem 3:

$$C_{CF} = - \left(\sum_{i=1}^r \mathcal{L}_i \right) C = \begin{pmatrix} -1 & 0 & 0 & 1 \end{pmatrix}, \quad (46)$$

and the initial marking of fusion

$$M_{F_0} = \sum_{i=1}^r B_i - \mathcal{L}_i M_0 = (1^{a_1} + 1^{a_2}). \quad (47)$$

Note that the incidence matrix of fusion (46) of control places **pc1** and **pc2** can be obtained by the sum of rows of incidence matrices of CPN in Fig. 3.

V. CONCLUSION

The focus of this work is the method "Fusion of Controllers" in Discrete Event Systems using coloured Petri nets (CPNs). The method "Fusion of Controllers" provides the merger of the control places that define the supervisor of a coloured Petri net (CPN), in which different color sets are associated with different variables. In this case, one obtains a control place, called supervisor of fusion which complies with the constraints placed originally, with the same performance of the modeled system.

A restriction of the method "Fusion of Controllers" is when different colors are associated with a same variable. Given this fact, one can not decompose the incidence matrix of the uncontrolled CPN, in order to determine the areas of controls related to the constraints with different colors. So to avoid conflicts the action of the supervisor resulting of the fusion is focused in the tokens of constraints, and ignores the complementary tokens of flow. In this case, it is synthesized a control more restrictive of the modeled system.

As a future study, it is proposed the application of the method "Fusion of Controllers" in manufacturing cells.

REFERENCES

- [1] Barroso G. C., Lima A. M. N. and Perkusich (1996). Synthesis of Supervisors Using Petri Nets. In Proceedings of the 13th World Congress of IFAC – International Federation of Automatic Control – volume J, 449-454, San Francisco, CA, USA.
- [2] Braga R. P. F. (2006). Contribuições ao Controle Supervisório de Sistemas a Eventos Discretos Modelados por Redes de Petri. Dissertação de Mestrado em Engenharia Elétrica, Universidade Federal de Santa Catarina.
- [3] Giua A. e DiCesari F. (1994). Blocking and controllability of Petri nets in supervisory control. IEEE Trans. on Automatic Control 39(4): 818-823.
- [4] Holloway L. E. e Krogh B. H. (1990). Syntheses of Feedback Control-Logic for a Class of Controlled Petri nets. IEEE Trans. Automatic Control 35(5): 514-523.
- [5] Iordache M. V. e Antsaklis P. J. (2006). Supervisory Control of Concurrent Systems: A Petri Nets Structural Approach, Boston, Birkhäuser.
- [6] Makungu M., Barbeau M. e St-Dennis R. (1999). Syntheses of Controllers of Processes Modeled as Colored Petri nets. Discrete event dynamic systems: Theory and Applications, v.9, p. 147-169.
- [7] Moody J. O. e Antsaklis P. J. (1998). Supervisory Control of Discrete Event Systems Using Petri Nets. Norwell: Kluwer Academic Publishers.
- [8] Prata B. A., Barroso G. C. e Arruda J. B. F. (2008). Um Novo Método para Controle Supervisório de Sistemas a Eventos Discretos Baseado em Redes de Petri Coloridas. Congresso Brasileiro de Automática, 02-05 de setembro.
- [9] Jensen K. e Kristensen L.M. (2009). Coloured Petri Nets – Modelling and Validation of Concurrent Systems. Springer-Verlag, New York.

Nonlinear H_∞ Control of Variable Speed Wind Turbines for Power Regulation and Load Reduction

Jován Mérida and Luis T. Aguilar

Abstract—A control strategy is realized which solve the problem of output power regulation of variable speed wind energy conversion systems by combining a linear control for blade pitch angle with a nonlinear H_∞ torque control which mitigate the effects of external disturbances that occur at the input and output of the system. The controller exhibits better power and speed regulation when compared to classic linear controllers. We assume that the effective wind speed and acceleration are available from measurements on the wind turbine. In order to validate the mathematical model and evaluate the performance of proposed controller, we used the National Renewable Energy Laboratory (NREL) aerolastic wind turbine simulator FAST. Simulation and validation results show that the proposed control strategy is effective in terms of power and speed regulation.

I. INTRODUCTION

As a result of increasing environmental concern, more and more electricity is being generated from renewable sources. Wind energy has proved to be an important source of clean and renewable energy in order to produce electrical energy. Nowadays, wind energy is by far the fastest-growing renewable energy technology, between 2000 and 2009, wind energy generation worldwide increased by a factor of almost 9 [1]. However, the performance of wind turbine must be improved. There are two primary types of horizontal-axis wind turbines: fixed speed and variable speed [2]. In this work we choose the variable speed because although the fixed speed system is easy to build and operate, does not have the ability that the variable speed system has in energy extraction, up to a 20% increase over fixed speed [2], [3]. Moreover, the variable speed system is much more complex to control. Advanced control plays an important role in the performance of large wind turbines. This allows better use of resources of the turbine, increasing the lifetime of mechanical and electrical components, earning higher returns. On the other hand, the new technological advancements improve the prospects of wind power allowing the design of cost-effective of wind turbines. Wind turbine controllers objectives depend on the operation area [4], [5]. Variable speed wind turbine operation can be divided into three operating regions:

- Region I: Below cut-in wind speed.
- Region II: Between cut-in wind speed and rated wind speed.
- Region III: Between rated wind speed and cut-out wind speed.

In region I wind turbines do not run, because power available in wind is low compared to losses in turbine system. Region II is an operational mode where it is desirable that the turbine capture as much power as possible from the wind, this because wind energy extraction rates are low, and the structural loads are relatively small. Generator torque provides the control input to vary the rotor speed while the blade pitch is held constant. Region III is encountered when the wind speeds are high enough that the turbine must limit the fraction of the wind power captured such that safe electrical, and mechanical loads are not exceeded. If the wind speeds exceed the contained ones in the region III, the system will make a forced stop the machine, protecting it from aerodynamic loads excessively high. Generally the rated rotor speed and power output are maintain by the blade pitch control with the generator torque constant at its rated value. Region III is considered in the present work where the power available in the wind exceeds the limit for which the turbine mechanics has been designed. Much of the research work in the wind energy conversion system control have used classical controllers. This, for several reasons. First, linear control theory is a well-developed topic while nonlinear control theory is less developed and difficult to implement. Second, most wind turbine control systems, to date, is based on linear control theory, thus the implemented wind turbine controllers are the based on linearized models [6]. Although thaonsider multiple controlled variables, such as controlling tower vibration, rotor speed, and blade vibration simultaneously (see, e.g., [7]–[10]).

The aim of this paper is to design a robust nonlinear controller to work in region III where wind speeds are high and dramatic growth of load structural. In this region the primary objective of the turbine controller is to reduce electrical power and rotor speed fluctuations while reducing the control loads. To limit loads and maintaining electric power production is necessary to limit the power and the rotational speed at its nominal values. This work presents a control strategy incorporating a linear control strategy with nonlinear H_∞ control strategy for variable-speed wind turbines intended regulated the electric power while the load is limited. The control structure is shown to robustly reject disturbances acting on the system, particularly those introduced by time-varying wind profiles. The controller takes into consideration the nonlinear nature of the wind turbine behavior, the flexibility of drive train, as well as the turbulent nature of the wind. The controller is design using a single mass model of the variable speed wind turbine and its performance has been tested through simulations. The

Authors are with Instituto Politécnico Nacional, avenida del parque 1310 Mesa de Otay, Tijuana 22510 México (email: jmeridao0900@ipn.mx; laguilarb@ipn.mx).

dynamic model of a horizontal axis turbine is simulated in MATLAB-Simulink and has been validated through a complex aerostatics wind turbine simulator FAST (Fatigue, Aerodynamics, Structures and Turbulence code) [11]. The overall controller is tested on a wind turbine simulator with realistic wind profiles providing satisfactory results.

The paper is organized as follows. In Section II is given background material on nonlinear H_∞ -control synthesis, the wind turbine model and problem formulation is presented in Section III. The control design is provided in Section IV. Performance of the proposed results are given in Section V through simulations. Section VI presents some conclusions.

II. BACKGROUND MATERIAL ON NONLINEAR H_∞ -CONTROL SYNTHESIS

Consider a nonlinear system of the form

$$\begin{aligned}\dot{x} &= f(x) + g_1(x)w + g_2(x)u \\ z &= h_1(x) + k_{12}(x)u \\ y &= h_2(x) + k_{21}(x)w\end{aligned}\quad (1)$$

where $x(t) \in \mathbf{R}^n$ is the state space vector, $u(t) \in \mathbf{R}^m$ is the control input, $w(t) \in \mathbf{R}^r$ is the unknown disturbance, $z(t) \in \mathbf{R}^l$ is the unknown output to be controlled, $y(t) \in \mathbf{R}^p$ is the only available measurement on the system. The following assumptions are assumed to hold:

(A1) The functions $f(x)$, $g_1(x)$, $g_2(x)$, $h_1(x)$, $h_2(x)$, $k_{12}(x)$, and $k_{21}(x)$ are twice continuously differentiable in x locally around $x = 0$.

(A2) $f(0) = 0$, $h_1(0) = 0$, and $h_2(0) = 0$.

(A3)

$$\begin{aligned}h_1^T(x)k_{12}(x) &= 0, & k_{12}^T(x)k_{12}(x) &= I, \\ k_{21}(x)g_1^T(x) &= 0, & k_{21}(x)k_{21}^T(x) &= I.\end{aligned}$$

These assumptions are inherited from the standard nonlinear H_∞ control theory [12], [13] and they are made for technical reasons. Assumption (A1) guarantees the well-posedness of the above dynamic system, while being enforced by integrable exogenous inputs. Assumption (A2) ensures that the origin is an equilibrium point of the non-driven ($u = 0$) disturbance-free ($w = 0$) dynamic system (1). Assumption (A3) is a simplifying assumption inherited from the standard H_∞ -control problem.

A causal dynamic feedback compensator

$$u = \mathcal{K}(x) \quad (2)$$

is said to be globally (locally) admissible controller if the closed-loop system (1)–(2) is globally asymptotically stable when $w = 0$.

Given a real number $\gamma > 0$, it is said that system (1), (2) has $\mathcal{L}_2$ -gain less than γ if the response z , resulting from w

for initial state $x(0) = 0$, satisfies

$$\int_0^T \|z(t)\|^2 dt < \gamma^2 \int_0^T \|w(t)\|^2 dt \quad (3)$$

for all $T > 0$ and all piecewise continuous functions $w(t)$.

The H_∞ -control problem is to find a globally admissible controller (2) such that $\mathcal{L}_2$ -gain of the closed-loop system (1)–(2) is less than γ . In turn, a locally admissible controller (2) is said to be a local solution of the H_∞ -control problem if there exists a neighborhood U of the equilibrium such that inequality (3) is satisfied for all $T > 0$ and all piecewise continuous functions $w(t)$ for which the state trajectory of the closed-loop system starting from the initial point $x(0) = 0$ remains in U for all $t \in [0, T]$.

A. Local State-Space Solution

Assumptions (A1)–(A3) allow one to linearize the corresponding Hamilton-Jacobi-Isaacs inequalities from [14] that arise in the state feedback and output-injection design thereby yielding a local solution of the time-invariant H_∞ -control problem. The subsequent local analysis involves the linear time-invariant H_∞ -control problem for the system

$$\begin{aligned}\dot{x} &= Ax + B_1w + B_2u \\ z &= C_1x + D_{12}u \\ y &= C_2x + D_{21}w\end{aligned}\quad (4)$$

where

$$\begin{aligned}A &= \frac{\partial f}{\partial x}(0), B_1 = g_1(0), B_2 = g_2(0), \\ C_1 &= \frac{\partial h_1}{\partial x}(0), D_{12} = k_{12}(0), \\ C_2 &= \frac{\partial h_2}{\partial x}(0), D_{21}(t) = k_{21}(0).\end{aligned}\quad (5)$$

Such a problem is now well-understood if the linear system (4) is stabilizable and detectable from u and y , respectively. Under these assumptions, the following conditions are necessary and sufficient for a solution to exist (see, e.g., [12]): C1) There exists a positive semidefinite symmetric solution of the equation

$$PA + A^T P + C_1^T C_1 + P \left[\frac{1}{\gamma^2} B_1 B_1^T - B_2 B_2^T \right] P = 0 \quad (6)$$

such that the matrix $A - (B_2 B_2^T - \gamma^{-2} B_1 B_1^T)P$ has all eigenvalues with negative real part.

According to the strict bounded real lemma [15], condition C1) ensures that there exists a positive constant ε_0 such that the system of the perturbed Riccati equation

$$\begin{aligned}P_\varepsilon A + A^T P_\varepsilon + C_1^T C_1 + \varepsilon I \\ + P_\varepsilon \left[\frac{1}{\gamma^2} B_1 B_1^T - B_2 B_2^T \right] P_\varepsilon = 0\end{aligned}\quad (7)$$

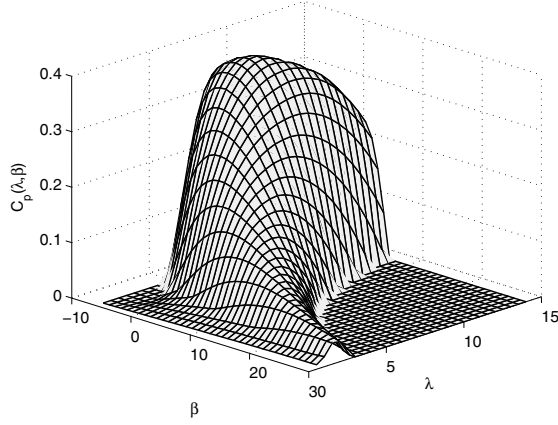


Fig. 1. Power coefficient curve.

has a unique positive definite symmetric solution $(P_\varepsilon, Z_\varepsilon)$ for each $\varepsilon \in (0, \varepsilon_0)$ where $\tilde{A} = A + \gamma^{-2} B_1 B_1^T P_\varepsilon$.

Equation (7) is subsequently utilized to derive a local solution of the nonlinear H_∞ -control problem for (1). The following result is extracted from [14].

Theorem 1: Let condition C1) be satisfied and let $(P_\varepsilon, Z_\varepsilon)$ be the corresponding positive solution of (7) under some $\varepsilon > 0$. Then the output feedback

$$u = -g_2^T(x) P_\varepsilon x \quad (8)$$

is a local solution of the H_∞ -control problem.

III. WIND TURBINE MODEL AND PROBLEM STATEMENT

The aerodynamic power captured by the rotor is given by the nonlinear expression [3]

$$P_a = \frac{1}{2} \rho \pi R^2 C_p(\lambda, \beta) v^3 \quad (9)$$

where v is the wind speed, ρ is the air density, and R is the rotor radius. The efficiency of the rotor blades is denoted as C_p , which depends on the blade pitch angle β , or the angle of attack of the rotor blades, and the tip speed ratio λ , the ratio of the blade tip linear speed to the wind speed. The parameters β and λ affect the efficiency of the system. The coefficient C_p is specific for each wind turbine. The relationship of tip speed ratio is given by

$$\lambda = R \frac{\omega_r}{v}. \quad (10)$$

The turbine estimated $C_p - \lambda - \beta$ surface, derived from simulation is illustrated in Fig. 1. This surface was created with the modeling software WTPerf [16], which uses blade-element-momentum theory to predict the performance of wind turbines [3]. The WTPerf simulation was performed to obtain the operating parameters for the CART (Controls Advanced Research Turbine).

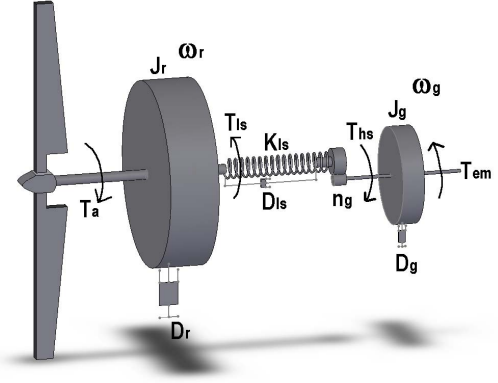


Fig. 2. Two-mass wind turbine model.

Fig. 1 indicates that there is one specific λ at which the turbine is most efficient. From (9) and (10), one can note that the rotor efficiency is highly nonlinear and makes the entire system a nonlinear system. The efficiency of power capture is a function of the tip speed ratio and the blade pitch. The power captured from the wind follows the relationship

$$P_a = T_a \omega_r \quad (11)$$

where

$$T_a = \frac{1}{2} \rho \pi R^3 \frac{C_p(\lambda, \beta)}{\lambda} v^2 \quad (12)$$

is the aerodynamic torque which depends nonlinearly upon the tip speed ratio. A variable speed wind turbine generally consists of an aeroturbine, a gearbox, and generator, as shown in Fig. 2.

The wind turns the blades generating an aerodynamic torque T_a , which spin a shaft at the speed ω_r . The low speed torque T_{ls} acts as a braking torque on the rotor. The gearbox, which increases the rotor speed by the ratio n_g to obtain the generator speed ω_g and decreases the high speed torque T_{hs} . The generator is driven by the high speed torque T_{hs} and braked by the generator electromagnetic torque T_{em} [6]. The mathematical model of the two mass wind turbine, can be described as follows:

$$\begin{aligned} J_r \dot{\omega}_r &= T_a(\omega_r, \beta, v) - K_{ls}(\theta_r - \theta_{ls}) - D_{ls}(\omega_r - \omega_{ls}) \\ &\quad - D_r \omega_r \\ J_g n_g \dot{\omega}_g &= -T_{em} n_g + K_{ls}(\theta_r - \theta_{ls}) + D_{ls}(\omega_r - \omega_{ls}) \\ &\quad - D_g n_g \omega_g \end{aligned} \quad (13)$$

where ω_{ls} is the low shaft speed, θ_r is the rotor side angular deviation, θ_{ls} is the gearbox side angular deviation, J_r is the rotor inertia, J_g is the generator inertia, D_r is the rotor external damping, D_g is the generator external damping, D_{ls} is the low speed shaft damping, and K_{ls} is the low speed

TABLE I
ONE-MASS MODEL PARAMETERS

Notation	Numerical value	Units
R	21.65	m
ρ	1.308	kg/m ³
J_t	3.92×10^5	kg m ²
D_t	400	N m/rad/s
H	36.6	m
P_e	600	kW
n_g	43.165	

shaft stiffness. The gearbox ratio n_g is:

$$n_g = \frac{\omega_g}{\omega_{ls}} = \frac{T_{ls}}{T_{hs}}. \quad (14)$$

If a perfectly rigid low speed shaft is assumed, a single mass model of the turbine may be then considered, upon using (14) and (13), one gets:

$$J_t \dot{\omega}_r = T_a(\omega_r, \beta, v) - D_t \omega_r - T_g, \quad (15)$$

where $J_t = J_r + n_g^2 J_g$, $D_t = D_r + n_g^2 D_g$, and $T_g = n_g T_{em}$ are the turbine total inertia, turbine total external damping, and generator torque in the rotor side, respectively. The parameters of the model are given in Table I. Those parameters are based on the CART which is a two-bladed, teetered, active-yaw, upwind, variable speed, variable pitch, horizontal axis wind turbine. The nominal power is 600 kW, the rated wind speed of 13 m/s, a cut out wind speed of 26 m/s, and it has a maximum power coefficient $C_{pmax} = 0.3659$. The rated rotor speed is 41.7 rpm. The required constraints for torque and rotor speed are 162 kN-m and 58 rpm respectively [17]. The gearbox is connected to an induction generator via the high speed shaft, and the generator is connected to the grid via power electronics.

Generator power will be controlled in region III when the wind speeds are high enough that the turbine must limit the fraction of the wind power captured so that safe electrical and mechanical loads are not exceeded. Then it operates at the rated power with power regulation during high wind periods by active control of the blade pitch angle or passive regulation based on aerodynamic stall [3]. The objective control in this region is to find a control law T_g and β to achieve the best tracking of rated of power while ω_r follows ω_d , as well as to reject fast wind speed variations and avoiding significant control efforts that induce undesirable torques and forces on the wind turbine structure. For variable speed wind turbine we design a controller using blade pitch and generator torque as control inputs.

Our objective is to design a regulator of the form

$$\dot{T}_g = \frac{1}{\omega_r} \left[C_0 \varepsilon_p - \frac{T_g}{J_t} (T_a - D_t \omega_r - T_g) - u \right] \quad (16)$$

where $P_e = \omega_r T_g$ is the electrical power, $\varepsilon_p = P_{nom} - P_e$, and C_0 is a positive constant while the rotor speed regulation is partly guaranteed by the pitch controller.

We confine our investigation to the velocity regulation problem where

1) The output to be controlled is given by

$$z = \rho_y \begin{bmatrix} 0 \\ w_r \\ P_{nom} - P_e \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u \quad (17)$$

with a positive weight coefficient ρ_y , and

2) The measurements

$$y = \begin{bmatrix} \omega_r \\ \varepsilon_p \\ T_g \end{bmatrix} + w_y \quad (18)$$

corrupted by the vector $w_y \in \mathbf{R}^3$, are only available.

IV. CONTROLLER SYNTHESIS

A. H_∞ Synthesis

To begin with, let us introduce the error state vector $x = (x_1, x_2, x_3) = (\omega_r, P_{nom} - P_e, T_g)$. After that, let us rewrite the state equation (15) as

$$\begin{aligned} \dot{x}_1 &= -\frac{D_t}{J_t} x_1 - \frac{1}{J_t} x_3 + \frac{1}{J_t} T_a \\ \dot{x}_2 &= -c_0 x_2 + u \\ \dot{x}_3 &= \frac{1}{x_1} \left[c_0 x_2 - \frac{1}{J_t} (T_a - D_t x_1 - x_3) x_3 - u \right]. \end{aligned} \quad (19)$$

Since the right-hand of the equations (19) are twice continuously differentiable in x locally around the origin $x = 0 \in \mathbf{R}^3$, the above H_∞ control problem is nothing else than the earlier theoretical approach to the nonlinear H_∞ control problem for the system (1) specified as follows:

$$\begin{aligned} f(x) &= \begin{bmatrix} \frac{1}{J_t} [T_a - D_t x_1 - x_3] \\ -c_0 x_2 \\ \frac{1}{x_1} \left[c_0 x_2 - \frac{T_a x_3}{J_t} + \frac{D_t x_1 x_3}{J_t} + \frac{x_3^2}{J_t} \right] \end{bmatrix}, \\ g_1(x) &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad g_2(x) = \begin{bmatrix} 0 \\ 1 \\ -\frac{1}{x_1} \end{bmatrix}, \\ h_1(x) &= \begin{bmatrix} 0 \\ x_1 \\ x_2 \end{bmatrix}, \quad k_{12}(x) = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \\ h_2(x) &= \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad k_{21}(x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned} \quad (20)$$

The subsequent local analysis involves the linear H_∞ -control problem for the system (4), (5), the state vector x contains

the deviation from the operating point

$$A_1 = \begin{bmatrix} \frac{1}{J_t} \left[\frac{\partial T_a}{\partial x_1} - D_t \right] & 0 & \frac{-1}{J_t} \\ 0 & -c_0 & 0 \\ A_{31} & \frac{C_0}{x_{1op}} & A_{33} \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 1 \\ -\frac{1}{x_{1op}} \end{bmatrix},$$

$$C_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad D_{12} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix},$$

$$C_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad D_{21} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where

$$A_{31} = -\frac{x_{3op}}{J_t} \frac{\partial(T_a x^{-1})}{\partial x_1} - \frac{c_0 x_{2op}}{x_{1op}^2} - \frac{x_{3op}^2}{J_t x_{1op}^2}$$

$$A_{33} = \frac{1}{J_t} [-T_{aop} + D_t x_{1op} + 2x_{3op}].$$

Now by applying Theorem 1 to system (1) thus specified, we derive a local solution of the H_∞ regulation problem.

B. Pitch Controller

In order to regulate the rotor speed and reduce generator torque oscillation, the torque control is aided by the pitch action. The pitch control allows to maintain the rotor speed around its nominal value. To achieve this, proportional action is used

$$\Delta\beta = K_p \varepsilon_w \quad (21)$$

where $\varepsilon_w = \omega_{nom} - \omega_r$ is the rotor speed tracking error and $K_p > 0$.

V. SIMULATION RESULTS WITH TURBULENT WIND

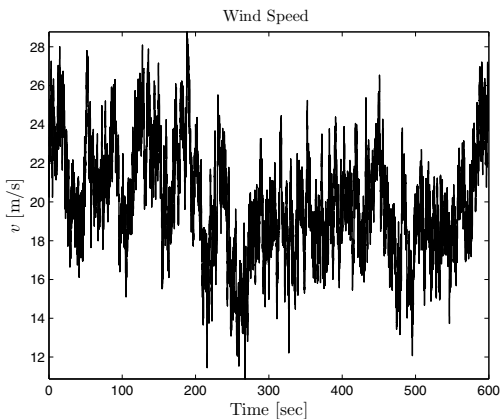


Fig. 3. Wind speed profile of 20 m/s mean value.

The proposed control approach has been simulated on based on the CART. This turbine was modelled with the mathematical model on Matlab-Simulink as well as in FAST aeroelastic simulator for validation. The wind speed is described as a slowly varying average wind speed superimposed by a rapidly varying turbulent wind speed. The model of the wind speed v at the measured point is

$$v = v_m + v_t \quad (22)$$

where v_m is the mean value and v_t is the turbulent component. The hub-reference wind field was generated using Turbsim [18]. The wind data consist of 600-sec in $v_m = 20$ m/s with 16% turbulence intensity, via the Kaimal turbulence model. Fig. 3 shows the profile of wind speed. In order to improve power regulation, rotor speed, and reduce the mechanical loads, we design H_∞ -regulation controller (8), (20) with $\gamma = 12$ and $\varepsilon = 0.001$.

A. Using the simplified Mathematical Model

First, the proposed controller is evaluated using the simplified mathematical model and then is evaluated with FAST simulator. In Figs. 4 and 5, simulation results are presented. The nominal values for regulation problem are shown in Figs. 4(a) and 4(b); it is observed that the proposed control is able to achieve precise speed and power regulation. In Fig. 4(a) the rotor speed is well regulated close to its nominal value. Fig. 4(b) shows that the electrical power P_e follows the nominal power. This value is almost equal to the nominal power P_{nom} . In this region pitch control alters the pitch of the blade, thereby changing the airflow around the blades resulting in the reduced torque capture of wind turbine rotor. Because of the pitch control, the control torque is reduced as shown in Fig. 5(a); if variation of T_g are large can be result in loads over the wind turbine affecting its behavior, but in this case its value goes up to 138.25 kNm, which is under the maximum one 162 kNm. These result in the reduction of the drive train mechanical stresses and output power fluctuations. In Fig. 5(b) we see that the collective pitch action did not exceed its limit. It was observed that a good tracking of a power reference and regulation of the rotor speed near of its nominal value can be successfully enforced with the controller used.

B. Using the simplified Mathematical Model

To verify the results obtained from the mathematical model, the proposed controller strategy performance has been tested for validation using FAST simulator. In this study, two degree of freedom are simulated: the variable generator and rotor speed. The performance increases in comparison with the mathematical model getting a better regulation speed and power tracking (Figs. 4(a) and 4(b)). The high speed shaft torsional moment with the mathematical model is greater then the one obtained with the simulator as well as the pitch angle (Figs. 5(a) and 5(b)). Checking out the results, one should conclude that the wind turbine control is satisfactory.

VI. CONCLUSIONS

Variable speed operation of wind turbine is necessary to increase power generation efficiency. As explained above, the wind turbine output power is limited to the rated power by the pitch and torque controller. We have developed the structure for nonlinear H_∞ control system in conjunction with a linear control strategy whose design procedure shown to be acceptable to solving the tracking of a power problem and regulation of the rotor speed near of its nominal value. To evaluate the performance the proposed strategy control approach has been simulated on a 600 kW two-blade wind turbine. Then, it has been validated using the wind turbine simulator FAST. Simulation results show that the proposed method is able to achieve the power and speed regulation while the load is limited. The states of the system were supposed available.

REFERENCES

- [1] R. Gelman, M. Kubik, and S. Buchanan, "2009 renewable energy data book," [Online]. Available: http://www.eere.energy.gov/maps_data/pdfs/eere.databook.pdf, 2010.
- [2] G. Ofualagba and E. U. Ubeku, "Wind energy conversion system-wind turbine modeling," in *IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, July 2008, pp. 1–8.
- [3] T. Burton, D. Sharpe, N. Jenkins, and E. Bossanyi, *Wind Energy Handbook*, 1st ed. Wiley, Dec. 2001.
- [4] L. Y. Pao and K. E. Johnson, "A tutorial on the dynamics and control of wind turbines and wind farms," in *IEEE Proc. Amer. Control Conf.*, St. Louis, MO, July 2009, pp. 2076–2089.
- [5] J. H. Laks, L. Y. Pao, and A. D. Wright, "Control of wind turbines: Past, present, and future," in *IEEE Proc. Amer. Control Conf.*, St. Louis, MO, 2009, pp. 2096–2103.
- [6] B. Boukhezzer, L. Lupu, H. Siguerdidjane, and M. Hand, "Multivariable control strategy for variable speed, variable pitch wind turbines," *Renewable Energy*, vol. 32, no. 8, pp. 1273–1287, 2007.
- [7] M. Grimble, "Horizontal axis wind turbine control: comparison of classical, LQG and H_∞ designs," *Dynamics and Control*, vol. 6, no. 2, pp. 143–161, April 1996.
- [8] A. Wright and L. Fingersh, "Advanced control design for wind turbines. part I: Control design, implementation, and initial tests," NREL, Tech. Rep. NREL/TP-500-42437, March 2008.
- [9] S. Thomsen, "Nonlinear control of a wind turbine." ME Thesis, Lyngby: Informatik og Matematisk Modellering, Danmarks Tekniske Universitet, 2006.
- [10] F. D. Bianchi, H. de Battista, and R. J. Mantz, *Wind Turbine Control Systems: Principles, Modelling and Gain Scheduling Design*, 1st ed., ser. Advances in Industrial Control, M. J. Grimble and M. A. Johnson, Eds. Springer-Verlag, London Limited, 2007.
- [11] J. J. and B. M. Jr., "Fast users guide," National Renewable Energy Laboratory, Golden, CO, Tech. Rep. NREL/EL-500-38230, 2005.
- [12] J. Doyle, K. Glover, P. Khargonekar, and B. Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Trans. Autom. Control*, vol. 34, no. 8, pp. 831–847, Aug 1989.
- [13] A. Isidori and A. Astolfi, "Disturbance attenuation and H_∞ -control via measurement feedback in nonlinear system," *IEEE Trans. Autom. Control*, vol. 37, no. 9, pp. 1283–1293, Sep 1992.
- [14] L. Aguilar, Y. Orlov, and L. Acho, "Nonlinear H_∞ -control of nonsmooth time varying systems with application to friction mechanical manipulators," *Automatica*, vol. 39, pp. 1531–1542, 2003.
- [15] B. Anderson and R. Vreugdenhil, *Network analysis and synthesis*. Englewood Cliffs NJ: Prentice Hall, 1973.
- [16] M. Buhl, "NWTC design codes WTPerf," [Online]. Available: <http://wind.nrel.gov/designcodes/simulators/wtperf/>, Sep. 2009.
- [17] L. Fingersh and K. Johnson, "Controls advanced research turbine (cart) commissioning and baseline data collection," NREL, Tech. Rep., 2002.
- [18] N. Kelley and B. Jonkman, "Turbsim user's guide: Version 1.50," National Renewable Energy Laboratory (NREL), Tech. Rep. NREL/TP-500-46198, September 2009.

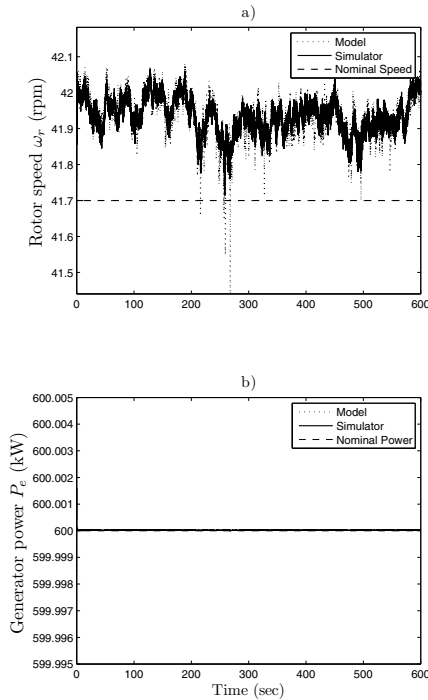


Fig. 4. Closed-loop system responses: (a) rotor speed and (b) generator power.

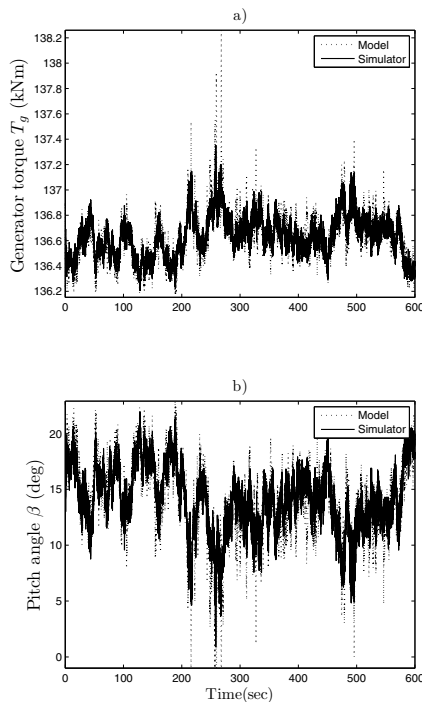


Fig. 5. Closed-loop system responses: (a) generator torque and (b) pitch angle.

Fault Diagnosis Using Hybrid Wavelet/ Elman Neural Networks

Letitia Mirea

Abstract— The paper addresses the problem of process fault diagnosis using a new kind of neural network, namely the hybrid wavelet/ Elman neural network. Hybrid learning based on a subclustering algorithm and the steepest-descent method is used to train the proposed neural network. The experiments refer to the sensor and actuator fault diagnosis of a sub-system from the evaporation station of the Lublin sugar factory, namely the evaporator. A generalised neural network based observer scheme is used to generate the residuals (symptoms) in the form of one step-ahead prediction errors. These are then classified by a Multi-Layer Perceptron neural network in order to take the appropriate decision regarding the type of the behaviour of the process (normal or abnormal).

I. INTRODUCTION

High performance process control and supervision often require the use of accurate process models. The use of modelling techniques of various forms has been the subject of extensive research in the field of control systems and fault condition monitoring, covering at least two decades [1]. There are many challenges in this important research domain. One of the most important challenge is that dynamic systems under feedback control have complex behaviour and are non-linear. Accurate and reliable model-based fault diagnosis requires the use of models that are well able to replicate the normal behaviour of the system, whilst being able to sharply and quickly isolate effects that are due to system faults [1]. Usually, a fault manifests as a deviation of at least one characteristic property or variable of the process with respect to the corresponding nominal value. A fault diagnosis system should be able to perform early detection of small faults, diagnosis of faults in different parts of the investigated process, detection of faults in closed-loop and supervision of the process in the transient states. All these requirements should be satisfied in the face of the existing measurement uncertainty, disturbances and incomplete knowledge about the process. A fault diagnosis system is efficient if it is able to avoid interpreting the unknown inputs (disturbances, measurement noise, modelling errors) as faults, i.e. to produce a reduced number of false alarms.

The current trends of research in fault diagnosis are focused on the development of *robust methods*. The robustness of a fault diagnosis system is achieved by

maximising the detectability and isolability of faults together with the minimisation of the effects of the uncertainties and disturbances on the fault diagnosis procedure.

Artificial Neural Networks (ANNs) are widely used in Fault Detection and Isolation (FDI) as technique based on process data able to cope with the robustness problem [2]. They are currently used in FDI as both dynamic non-linear models to generate the symptoms and pattern classifiers to evaluate them. These approaches do not require an accurate mathematical model of the process, but need representative training data.

Wavelet decompositions provide a useful basis for localised approximation of functions with any degree of regularity at different scales and with a desired accuracy. Wavelets can therefore be seen as a new basis for approximating functions and can be used for non-linear system identification [3], [4], [5]. This approach is based on the concept of approximating an arbitrary non-linear function in terms of dilates and translates of a single function, namely the ‘mother’ wavelet function [6]. Wavelet Neural Networks (WNNs) were introduced as advanced computational schemes for signal representation and classification. They were designed to combine the results of the wavelet decomposition theory with the learning properties characterising the neural networks. WNNs have been successfully applied to the identification of non-linear systems and adaptive control [3], [4], [5] due to the time-frequency localisation properties of wavelets and the learning ability of the neural networks.

Elman recurrent network is a well-known recurrent topology, developed by Jeffrey Elman. An Elman network has a set of context nodes. Each context node receives input from a single hidden node and sends its output to each node in the layer of its corresponding hidden node. Since the context nodes depend only on the activations of the hidden nodes from the previous input, the context nodes retain state information among inputs [7].

This paper refers to the development of a new type of neural network by combining the above mentioned neural structures. This new neural network, namely the Hybrid Wavelet/ Elman Neural Network (HWENN), is a dynamic neural network due to the context neurons in the Elman part of the net and, therefore, leads to a decrease in dimension of the its input space. Its application to sensor and actuator fault diagnosis of a sub-system from the evaporation station of a sugar factory is also considered. Process input-output data are used to train the HWENN, i.e. to determine the

Manuscript received June 18, 2011.

L. Mirea is with the "Gh. Asachi" Technical University of Iasi, Department of Automatic Control and Industrial Informatics, Blvd. D. Mangeron 53A, RO-700050 Iasi, Romania (phone: +40-232-278680, fax: +40-232-214290, e-mail: lmirea@ac.tuiasi.ro).

network parameters that would minimize a performance index. Firstly, a clustering algorithm developed by Chiu [8] is used to determine the number of wavelet nodes and their parameters' initial values. Following this, a gradient-based learning algorithm is applied in to refine the HWNN parameters.

The paper is organised in five sections as follows. Section II presents the architecture and the learning procedure for HWENN. Section III refers to the design of an FDI system based on HWENNs (residual generation) and Multi-Layer Perceptron (residual evaluation). The application of the developed HWENNs to the problem of detecting and isolation sensor and actuator faults in the evaporation station of a sugar factory is presented in Section IV. Conclusions are given in Section V.

II. HYBRID WAVELET/ELMAN NEURAL NETWORK

Based on the theory of wavelet decomposition [6], any function $f(x) \in L^2(\mathfrak{R})$ can be approximated by:

$$f(x) \cong \sum_{k=1}^K c_k \cdot \psi\left(\frac{x - b_k}{a_k}\right) \quad (1)$$

where the function ψ is called “mother wavelet”, c_k are the weighting coefficients, $b_k \in \mathfrak{R}$ are the translation parameters and $a_k \in \mathfrak{R}_+^*$ are the dilation parameters.

Relation (1) can be implemented via a feed-forward neural network structure, similar to the radial basis functions networks. This wavelet neural network has one hidden layer of neurons with the transfer functions obtained as dilated and translated versions of the “mother wavelet”. This function should have zero-mean and should be localised in both space and frequency domains. In the present approach, the following “Mexican hat” is considered as the “mother wavelet”:

$$\psi(x) = \alpha \cdot (1 - \alpha \cdot x^2) \cdot e^{-\frac{\alpha \cdot x^2}{2}}. \quad (2)$$

The constant α in relation (2) is set to the value 0.5.

In the case of a Single-Input Single-Output (SISO) WNN, the output of the net is:

$$y = c_0 + \sum_{j=1}^M c_j \cdot \psi\left(\frac{u - b_j}{a_j}\right), \quad (3)$$

where u is the WNN input and M is the number of wavelet nodes. The wavelet neuron itself has no capacity to implicitly identify offsets within measured data. Thus, the explicit estimation of a mean value possibly existing in the process-measured data becomes substantial, motivating the introduction of the constant term: c_0 , in eq. (3).

For a problem with P inputs, multidimensional wavelets must be considered. The most frequent choice [3], [4], [5] is that of separable wavelets, i.e. the product of P mono-dimensional wavelets associated with each input:

$$\Psi_j(\mathbf{u}) = \prod_{i=1}^P \psi\left(\frac{u_i - b_{ij}}{a_{ij}}\right), \quad j = 1, \dots, M. \quad (4)$$

where $\mathbf{u} = [u_i]_{i=1, \dots, P}$ is the vector of WNN inputs, $\mathbf{b}_j = [b_{ij}]_{i=1, \dots, P}$ and $\mathbf{a}_j = [a_{ij}]_{i=1, \dots, P}$, $j = 1, \dots, M$ are the translation and dilation vectors, respectively. In this case, the WNN performs static mapping between its input space and its output space. The WNN output is given by:

$$\hat{y} = c_0 + \sum_{j=1}^M c_j \cdot \Psi_j(\mathbf{u}). \quad (5)$$

In the case of monitoring a dynamic system using sampled data, a discrete time process representation is required. This can be achieved by identifying WNN models for each system output, i.e. Multi-Input Single-Output (MISO) models. To be used for dynamic system identification, the wavelet neural networks require spatial time representation.

Using time delay units is the simpler way of including dynamic elements into the WNN functionality. In this case, the static WNN has to be fed with current and delayed values of the inputs and the outputs of the system [2], [9], [10]. This leads to an increase of the dimension of the neural network input space. Another possibility is to include into the WNN architecture dynamic elements or recurrent connections. In this case, the wavelet neural network with internal dynamic elements has to be fed only with current values of the process inputs and outputs. In this way, the dimension of the WNN input space is lower than in the previous approach. This further implies a reduction in the WNN's structure complexity.

The paper introduces a new type of neural network by combining the WNN structure with the Elman neural network that possesses dynamic elements. This will be further referred as Hybrid Wavelet/ Elman Neural Network (HWENN). The proposed HWENN has the architecture depicted in Fig. 1. Here q^{-1} stands for the delay operator. The output of the HWENN is given by:

$$\begin{aligned} \hat{y}[k] &= \Phi(c_0 + \sum_{m=1}^M c_m \cdot \tilde{z}_m[k]), \\ \tilde{z}_m[k] &= \Gamma_m \left(\sum_{i=1}^M \tilde{w}_{m,i} \cdot \tilde{z}_m[k-1] + \sum_{j=1}^M w_{m,j} \cdot z_m[k] \right), \\ z_m[k] &= \Psi(\mathbf{u}[k]), \quad m = 1, \dots, M \end{aligned}$$

where $[k]$ is the sample time instant. The activation functions of the neurons in the Elman part of the network are as following: Γ_m , $m = 1, \dots, M$ are of hyperbolic tangent type and Φ is of linear type.

One considers N data pairs collected from the inputs and outputs of the process. It is desired that the output of the HWENN, $\hat{y}[k]$, would fit the system output, $y_p[k]$, where $[k]$ is the sample time instant:

$$\hat{y}[k] = f(\mathbf{u}_p[k], y_p[k-1]) \cong y_p[k], \quad (6)$$

where $\mathbf{u}_p$ denotes the process input, $y_p[k]$ represents the process output, $\hat{y}[k]$ denotes the approximating output given by the HWENN and f is the function that denotes the mapping performed by the neural net.

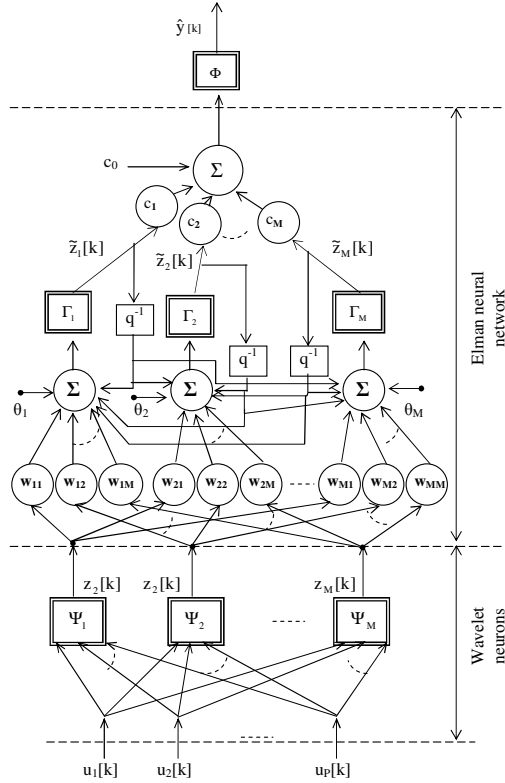


Fig. 1. Architecture of the HWENN

The HWENN network parameters $\{\mathbf{a}_m, \mathbf{b}_m, c_0, c_m, w_{m,j}, \tilde{w}_{m,i}, \theta_m\}$, $m, j = 1, \dots, M$, are adapted in order to minimise a quadratic cost-function such as the sum-squared error between the network output $\hat{y}[k, \boldsymbol{\theta}]$, and the process output, $y_p[k]$:

$$SSE = \frac{1}{2} \cdot \sum_{k=1}^N (y_p[k] - \hat{y}[k, \boldsymbol{\xi}])^2, \quad (7)$$

where $\boldsymbol{\xi}$ represents the vector of HWENN parameters.

As the network output is non-linear in its parameters, a non-linear optimisation technique must be applied. The network parameters are thus optimised iteratively based on the steepest-descent method:

$$\xi_{\text{new}} = \xi_{\text{old}} - \eta \frac{\partial SSE}{\partial \xi} \quad (8)$$

where ξ is an arbitrary network parameter and η is the learning rate. The strategy of adaptive learning rate with momentum term is also used in order guarantee a stable learning.

The problem that arises here is how the HWENN parameters should be initialised in order to avoid the

convergence of the gradient procedure to an undesirable local minimum and to insure the numerical stability of the learning algorithm. A random initialisation of all parameters to small values is not suitable, as the components of the gradient of the cost function may be very small in the area of interest. Thus, one wants to take advantage of the input space where wavelets are not zero [10]. Therefore, the HWENN parameters are initialised based on a fuzzy clustering algorithm developed by Chiu in [8].

The purpose of the fuzzy clustering algorithm is to distil natural groupings of the HWENN input data set, producing a concise representation of the system's behaviour.

A collection of N data points $\{\mathbf{u}(1), \mathbf{u}(2), \dots, \mathbf{u}(N)\}$, $\mathbf{u}(i) = [\mathbf{u}_p(i)]_{p=1, \dots, P}$ in a P dimensional space is considered.

These data points represent the input vectors of the HWENN that are used in the training stage. Without loss of generality, it is assumed that each data point has been normalised in each dimension, i.e. the data points are bounded by a hypercube. Each data point is considered as a potential cluster centre and a measure of the potential of data point $\mathbf{u}(i)$ is defined as a function of its distance to all other data points [8]:

$$P_i = \sum_{j=1}^N \exp\{-\alpha \cdot \|\mathbf{u}(i) - \mathbf{u}(j)\|_2^2\}, \quad i = \overline{1, N}, \quad \alpha = \frac{4}{r_\alpha^2},$$

where $\|\cdot\|_2$ is the Euclidean norm and r_α is a positive constant that defines the radius of a neighbourhood. Data points outside this radius have little influence on the potential. So, a data point with many neighbouring data points will have a high potential value.

The first cluster centre is considered to be that data point with the highest potential value. Let $\mathbf{b}_1$ be the first cluster centre and P_1^* be its potential value. In order to select the next cluster centre, the potential of each data point is revised according to the relation [8]:

$$P_i = P_i - P_1^* \cdot \exp\{-\beta \cdot \|\mathbf{u}(i) - \mathbf{b}_1\|_2^2\},$$

$$i = \overline{1, N}, \quad \beta = \frac{4}{r_b^2},$$

where r_b is a positive constant that defines the radius of a neighbourhood that will have a measurable reduction in potential. To avoid obtaining closely spaced clusters, r_b should be taken greater than r_α .

The second cluster centre is considered to be that data point with the highest revised potential value. Further, the potential of the remaining data points is reduced according to their distance to the second cluster centre. Generally, after the k -th centre has been selected, the potential of the remaining data points is revised according to the relation :

$$P_i = P_i - P_k^* \cdot \exp\{-\beta \cdot \|\mathbf{u}(i) - \mathbf{b}_k\|_2^2\},$$

where $\mathbf{b}_k$ is the k -th cluster centre and P_k^* is its potential value.

The process of acquiring new cluster centre repeats and finally, a number of M cluster centres $\mathbf{b}_k = [\mathbf{b}_{p,k}]_{p=1,\overline{P}}$, $k = \overline{1,M}$ are obtained. For each data point $\mathbf{u}(i)$, $i = \overline{1,N}$ a degree of membership μ_{ki} to each obtained cluster k , $k = \overline{1,M}$ is computed using the relation [6]:

$$\mu_{ki} = \frac{1}{\sum_{j=1}^M \left(\frac{\|\mathbf{u}(i) - \mathbf{b}_k\|_2}{\|\mathbf{u}(i) - \mathbf{b}_j\|_2} \right)^2}, i = \overline{1,N}, k = \overline{1,M}.$$

Using the obtained membership degrees, the standard deviations of each cluster are obtained using the relation:

$$a_{p,k} = \frac{\sum_{i=1}^N (\mu_{ki})^2 \cdot (u_p(i) - b_{p,k})^2}{\sum_{i=1}^N (\mu_{ki})^2},$$

$$\mathbf{a}_k = [a_{p,k}]_{p=1,\overline{P}}, k = \overline{1,M}$$

These values are used as dilation parameters for the wavelet nodes. The initial values for the Elman neural network's weights are set to small, positive, random values.

III. FDI SYSTEM BASED ON HWENN

A. Residual generation

For the generation of symptoms, the HWENN is used to model the considered process. For each output, a Multi-Input Single-Output (MISO) neural model is identified. Since the control system tends to hide the faults, both process inputs and outputs are used as inputs to the HWENN.

The developed HWENN models are further used in an observer scheme [10] to generate sets of residuals in the form of the one step-ahead prediction errors. The Extended Neural Generalised Observer Scheme (E-NGOS) is considered.

The extended Neural Generalised Observer Scheme (E-NGOS). For the present application, one considers a MISO process with I inputs $u_i[k]$, $i = \overline{1,...,I}$ and the output $y_p[k]$, all known at sampling time k . At first, the considered process is identified by one HWENN that is driven by all inputs and the output of the system. The network estimates the output of the process:

$$\hat{y}_0[k] = f_0(\mathbf{u}[k], y_p[k-1]), \mathbf{u}[k] := [u_i[k]]_{i=1,...,I} \quad (9)$$

Then, the Neural Generalised Observer Scheme (NGOS) is developed. It consists of as many HWENN as process inputs are available. Each neural network of the NGOS is driven by the process output and all inputs except one input. The output of the MISO process is approximated by one HWENN model of the NGOS as:

$$\hat{y}_j[k] = f_j(\mathbf{u}_j[k], y_p[k-1]),$$

$$\mathbf{u}_j[k] := [u_i[k]]_{i=1,...,I; i \neq j}; j = \overline{1,...,I} \quad (10)$$

The resulting set of neural models approximates all outputs of the process. The training of the HWENN is based on the system data corresponding to its normal behaviour. The residuals are then generated by means of the *extended NGOS* given by eqs. (9), (10). One step-ahead prediction errors are obtained:

$$\epsilon_j[k] = y_p[k] - \hat{y}_j[k]; j = \overline{0,...,I}. \quad (11)$$

The residual ϵ_0 is affected by all process variables. Instead, each residual ϵ_j , $j = \overline{1,...,I}$ is sensitive to the process output and all inputs but the j^{th} input. In case of a faulty input variable, the decoupled residual remains small, while the other are influenced. This is valid if all used process inputs are uncorrelated. If some of those variables are correlated, more residuals are not affected by a certain fault. In this case, the residual ϵ_0 , based on Eq. (11), is important by reflecting all kinds of faults. These patterns that reflect all the changes in the process functioning are used to detect and locate the faults.

B. Residual evaluation

Residual evaluation means to match each pattern of the residual vector with one of the pre-assigned classes of faulty behaviour, if available, and the fault-free case, respectively.

The main problem in the residual evaluation stage is the uncertainty in classification of the patterns that may arise from the overlapping nature of various classes. Therefore, a robust decision can be achieved by using a neural network as pattern classifier [10]. The static Multi-Layer Perceptron with sigmoid neurons is considered in the present approach.

The neural classifier maps the patterns (11) from the residual space into a decision space, such as the patterns belonging to a class of process behaviour (normal or abnormal) to cluster around pre-selected points, optimally chosen in the decision space [11]. A class of process behaviour is detected and isolated if an unknown input pattern is mapped closest to one of the target vectors. That multi-dimensional point corresponds to a certain class that reflects one of the considered process behaviours (normal or faulty).

Usually, the target vectors of decision space $\{\mathbf{t}_c\}$, $c = \overline{1,...,C}$ are chosen to be the set of vertices of C -unit (Euclidean) vectors. Here C represents the number of the considered classes of process behaviour. An efficient way to choose the target vectors is suggested in [11] as a subset of C vectors with K binary-valued components in the set $\{0.1, 0.9\}$, among the vertices of a hypercube. The number of the components is $K = 1 + [\log_2(C)]$, where $[\cdot]$ denotes the function of greatest integer. This will lead to a reduced number of neurons in the output layer of the neural classifier, thus a reduce in the complexity of the neural

A fault is only detected if the input pattern is mapped far from all learned classes. This situation corresponds to new (faulty) process behaviour and only the synthesis of the classifier must be reconsidered for further fault diagnosis. In the present approach, the decision logic is based on the Euclidean distance between the actual output and the target vectors of the classifier.

A. Process Description

The process is used to reduce the water content of sucrose juice. The liquid goes through a series of five sections. In each passage, the sucrose concentration increases. The juice steam recovered from one stage is used as a heating source for the next section. The sugar evaporation control should be performed in such a way that the used energy is optimised and the required quality of the final product is achieved [13].

A number of 3000 rows of measurements from another month of plant functioning have been used to test the quality of identified models. Another testing set has been created by introducing artificial single faults in the first testing data set. Faults are generated by adding or subtracting 5, 10 and 15 percent of the whole possible range

The investigated process is decomposed in five MISO sub-processes [2]. In this paper, only one sub-process has been considered, namely the evaporator. A diagnosing unit has been developed to detect and isolate actuator and sensor faults.

The selection of an appropriate learning data set has been done as a trade-off between the following criteria: - the selection of data from a day where the inputs have significant variation in order to ensure the maximum possible excitation of the process; - the number of missing values to be as small as possible; - the number of uncertain values to be as small as possible. The isolated missing and uncertain values have been replaced by means of polynomial interpolation. For each model, a training data set of 3000 rows has been selected. This corresponds to a period of time of about 8 hours, i.e. a production shift.

B. Experimental Results

Fig. 3 presents some of the obtained identification results. Fig. 3a refers to the extended training data set, while Fig. 3b refers to the testing data set (different than the training data).

In these figures, the process output has been represented with solid line and the output of the HWENN model has been represented with dotted line. The HWENN models are characterised by: $M \in \{4, 5\}$.

Altogether, 9 classes of sub-process behaviour are diagnosed: 1 - normal behaviour; 2 and 3 - positive and negative deviations of the output, respectively; 4 and 5 - positive and negative deviations of the first input, respectively; 6 and 7 - positive and negative deviations of the second input, respectively, and 8 and 9 - positive and negative deviations of the third input, respectively. The results of classification and final decision (diagnosis) are presented in Table I.

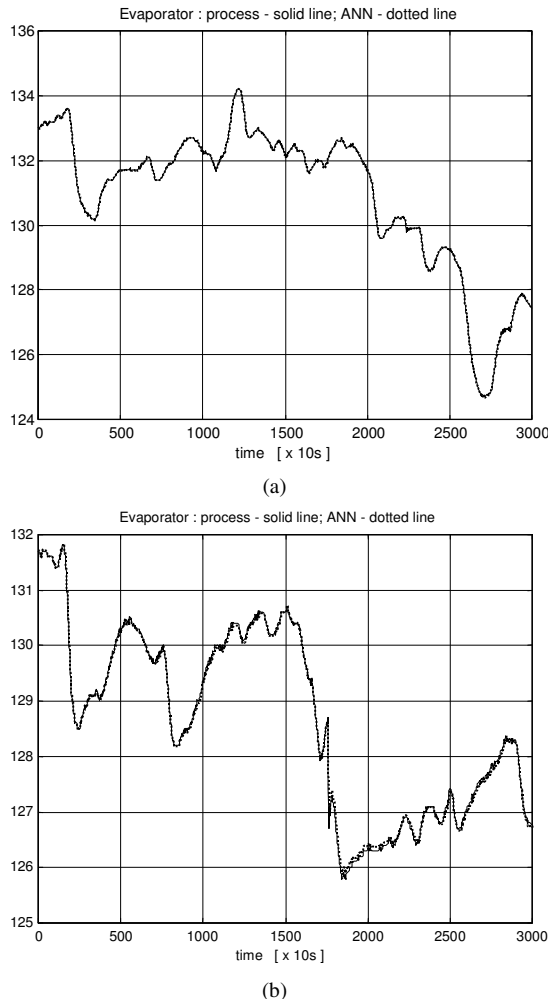


Fig. 3. Evaporator: identification results

TABLE I
PERFORMANCE OF THE DIAGNOSIS SYSTEM

Recognition rates [%]	Classification	Diagnosis
Normal behaviour	99.14	99.67
Faulty behaviours	98.56	98.72
Global	98.63	99.31

V. CONCLUSIONS

The present paper investigates the development and application of a new type of dynamic neural network to robust fault diagnosis of an industrial plant. Thus, the Hybrid Wavelet/ Elman Neural Network was developed combining the good local properties of the wavelet nodes with the recurrent structure of the Elman neural network that gives the HWENN memory of its past states. A neural approach to the generalised observer scheme is used. Residuals are generated by using the developed HWENN. The obtained symptoms are classified using a static Multi-Layer Perceptron. One remark the good results obtained with the FDI system based on the HWENN, reflected in recognition rate of around 99 %.

In the framework of fault diagnosis, further research will investigate the possibility to generate symptoms that reflect the behaviour of the process based on discrete wavelet decomposition. It is expected a performance improvement due to the increase of the active robustness of the FDI system. This would be achieved because wavelet decomposition basically performs an analysis in both frequency and time domain.

REFERENCES

- [1] Patton R.J., Clark R., Frank P.M., *Issues of Fault Diagnosis for Dynamic Systems*, Springer-Verlag, New York, Inc., 2000
- [2] Mirea L., Marcu T., "System identification using functional-link neural networks with dynamic structure", *IFAC Congress b'02*, Barcelona, Spain, CD-ROM Proceedings, 2002.
- [3] Zhang Q., Benveniste A., "Wavelet networks", *IEEE Trans. on Neural Networks*, **3**, 6, pp889-898, 1992.
- [4] Oussar Y., Rivals I., Personnaz L., Dreyfus G., "Training wavelet networks for non-linear dynamic input-output modelling", *Neurocomputing*, **20**, pp. 173-188, 1998.
- [5] Coca D., Billings S. A., "Non-linear system identification using wavelet multi-resolution models", *Int. J. Control*, **74**, 18, p1718-1736, 2001.
- [6] Daubechies I., *Ten Lectures on Wavelets*, SIAM Press, Philadelphia, 1992.
- [7] Adem Kalinli, Seref Sagiroglu, Elman network with embedded memory for system identification, *Journal of Information Science and Engineering*, **22** (6) (2006), pp. 1555-1568.
- [8] Chiu, S.L. (1994), A Cluster Estimation Method with Extension to Fuzzy Model Identification, *Proc. of the 3rd IEEE Int.Conf. on Fuzzy Systems*, Orlando, Florida, USA, 1240-1245.
- [9] Haykin S., (1994), *Neural Networks – A Comprehensive Foundation*, MacMillan, New York, 1994.
- [10] Mirea L., Patton R. J., "Component Fault diagnosis using wavelet neural networks with local recurrent structure", *SAFEPROCESS 2006*, Beijing, Popular Republic of China, CD-ROM Proceedings, pp. 91 – 96, 2006.
- [11] Marcu T., "Pattern Recognition Techniques Using Fuzzily Labeled Data for Process Fault Detection", *Int. J. of Applied Mathematics & Computer Science*, **6**, 4, pp815-840, 1996.
- [12] Bartys M., Patton R. J., Syfert M., Heras de las Sera, Quevedo J., "Introduction to the DAMADICS Actuator FDI Benchmark Study", *Control Engineering Practice*, (Invited Special Issue Paper), **14**: pp. 577-596, 2006.

Two Server Topologies for a Distributed Fingerprint-Based Recognition System

Radu F. Miron, Tiberiu S. Letia and Mihai Hulea

Abstract—Fingerprint based techniques are the most well developed biometric recognition methods. This paper presents the architecture and some implementation details for a distributed fingerprint based recognition system. Two different topologies for interconnecting the servers of the proposed system are presented: a peer-to-peer based architecture and MOM based one. The main objectives are to present these two distributed architectures and their communication protocols, together with comparative test results. The main downfall of the peer-to-peer based topology is its low scalability due to the great number of inter-server connections. The MOM based topology inherits the disadvantages of centralized systems.

I. INTRODUCTION

BIOMETRIC recognition systems are the most advanced automated techniques used to establish someone's identity. They are based on physiological and/or behavioral characteristics, identifying or verifying the identity of the individuals by "who they are", rather than by "what they possess" or by "what they remember" [1].

A number of different biometric identifiers are used in various applications. Among the technologies based on the biometrics, the fingerprint-based technology is the undisputed leader, considering its market share of more than 67% [2].

The reason behind the large use of fingerprint-based recognition systems, to the detriment of the other biometric identifiers, is that they better satisfy the following requirements: universality, acceptability, immutability, individuality and collectability [3].

Like all biometric systems, usually, an *Automated Fingerprint Identification System* (AFIS) should allow three different tasks: *the enrollment*, *the verification* and *the identification*. The enrollment process is responsible for saving the identity of subjects and their fingerprints into the database of the system. Usually the fingerprint undergoes a series of image processing steps before being saved into the DB. These steps have the goal of extracting the main features of a fingerprint, also known as the template. The difference between the last two tasks consists of the number of comparisons needed in order to establish the identity.

Manuscript received August 31, 2011.

Radu-Florin Miron is with The Technical University of Cluj-Napoca, Cluj 400114 Romania (phone: +4 0264 401432; fax: +4 0264 599893; e-mail: radu.miron@aut.utcluj.ro).

Tiberiu Stefan Letia is with The Technical University of Cluj-Napoca, Cluj 400114 Romania (e-mail: tiberiu.letia@aut.utcluj.ro).

Mihai Hulea is with The Technical University of Cluj-Napoca, Cluj 400114 Romania (e-mail: mihai.hulea@aut.utcluj.ro).

Identity verification implies a one-to-one comparison of the input fingerprint with the pre-stored template for the claimed identity of individuals. The identification task tries to find a match for a given fingerprint, by scanning the entire template database (one-to-many comparison). Usually, when there is not necessary to discriminate between verification and identification, the term *recognition* is used [4].

This paper proposes two different topologies for interconnecting the servers of a distributed fingerprint-based recognition system. The proposed system is composed of a number of mobile devices connected to centralized processing units (servers), which are responsible for the enrollment and recognition tasks. The mobile devices are used for collecting the personal data, for sending them to its assigned server along with the needed request identifier (*enroll*, *verify* or *identify*) and for receiving the answers to their requests. The servers communicate with each other, being able to forward unsolved recognition requests. A simplified model of the system is presented in Fig. 1.

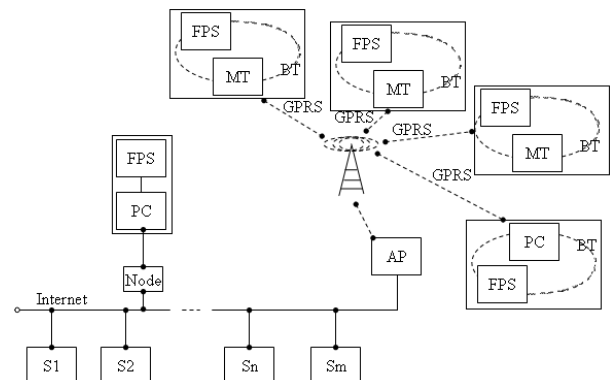


Fig. 1. System's general hardware architecture. The system's servers are denoted by S1,...,Sm. The fingerprint sensors (FPS) are connected via blue-tooth (BT) to the mobile terminals (MT) and via wire (or BT) to the personal computers (PC). AP denotes possible access-points found in the network.

This paper focuses on the architecture and on the implementation of the servers. The communication protocols between the servers and between the software modules are mainly described. The architecture of the mobile devices and the client application of the system were described in a previous paper [5]. Both ways of interconnecting the servers of the system are based on the classical *client-server model*. The first proposed topology is a *peer-to-peer system*, in which each server is connected to all the others. The second solution is based on the *Message Oriented Middleware (MOM)* architecture.

II. RELATED WORK

A *distributed system* comprises a collection of computers, interconnected through a network (physically) and through the middleware (logically), which are able to coordinate their activities and to share their resources, so that users perceive the system as a single computing entity [6].

The *middleware* concept was developed in the early 90's as a solution to the problems caused distributed heterogeneous systems. It represents the software component that connects the distributed modules and processes of a system [7]. The middleware evolved as a response to the need of implementing efficient distributed systems, which enables code and modules reusability.

According to [8], the middleware is defined as the intermediate software that resides on top of the operating system, used by applications to communicate inside a network. It cannot be strictly delimited by the operating system (e.g. the TCP/IP stack) and it performs the following tasks are: *hiding distribution, hiding heterogeneity, providing uniform, high-level interfaces* to software developers and *supplying a set of common services* for general purpose functions.

The *Message Oriented Middleware (MOM)* is a software infrastructure that focuses on sending/receiving messages between the distributed components of a system. Unlike *Remote Procedure Call (RPC)* and Java's similar middleware, *Remote Method Invocation (RMI)*, MOM's communication is asynchronous, which leads to a more robust distributed system [9]. *Java Message Service (JMS)* represents the specifications for MOM-based architectures. JMS-based applications are event-driven systems with asynchronous communication, which ensures deadlock avoidance [10].

Reference [11] proposes a web based remote fingerprint recognition system. The system has a 3-tier architecture: *presentation tier, functionality tier* and *data tier*. The main layer is the functionality tier, which is responsible for fingerprint image processing and recognition. The presentation layer is represented by the graphical user interface shown in the web-browser.

Another example of a distributed fingerprint recognition system is the *sensor-client-server model*. The client application, running on a PC, is connected to a fingerprint sensor. The personal data are sent to one server (responsible for enrollment/recognition tasks) via Internet [12].

The use of mobile devices (e.g. mobile phones, PDAs etc.) fitted with fingerprint sensors, for capturing and for sending fingerprint images to a central processing unit is presented in [13]. An experimental mobile device capable of scanning fingerprints, which is able to connect to a mobile phone or a PC by using Bluetooth technology, is presented in [5]. This device is used to capture fingerprints that are then sent to the client (mobile or PC application), which then forwards this file to a central server for recognition or enrollment, along with the necessary information.

III. THE FUNCTIONALITY AND THE TOPOLOGIES OF THE PROPOSED SYSTEM

A. The Functionality of the System

A use case diagram presents a graphical overview of the functionality provided by the proposed system (refer to Fig. 2). In this diagram the actor *User1* is able to select the desired task it wants to perform. Although the request is made on the client side, the actual task is executed onto the server side. All the use cases (functions) available to a user are dependent on the *Login* task (a user has to login in order to be able to request any other tasks). The other servers of the system can also become actors, but they send their request directly to the current server and they are not able to enroll subjects.

Starting from the tasks the system has to perform, two

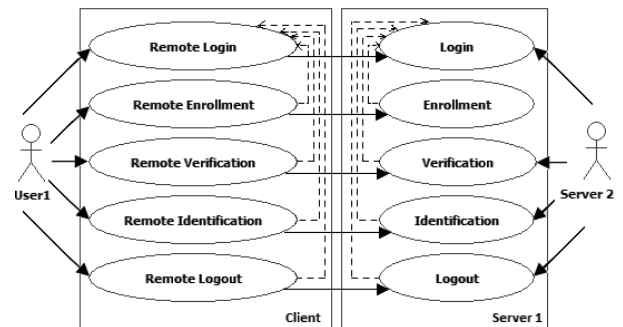


Fig. 2. System's functionality described by UML use case diagram. The system's main functions available to a standard user are depicted inside the *Client* boundary. Each selected operation is executed on the server side (*Server1*). Another server (*Server2*) can also be a client for *Server1*, except for the *Enrollment* task.

different topologies for interconnecting the servers of the system were considered: a classical *peer-to-peer* topology, in which each server connects to all the other available servers in order to solve a recognition request that couldn't be solved locally, and a *MOM architecture based* topology. In the second case only the MOM architecture was borrowed; even though the system was implemented in Java, the *JMS API* itself was not used [14].

B. Peer-to-Peer Topology

This architecture implies a connection between each server, as in Fig. 3.

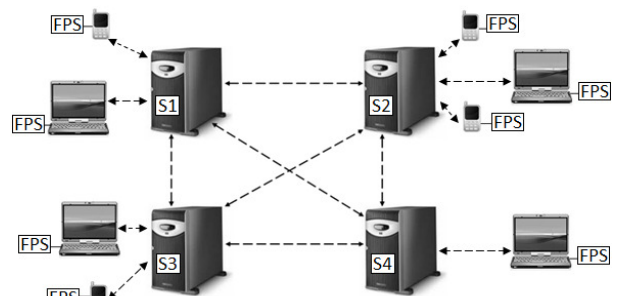


Fig. 3. Peer-to-peer servers' topology. *S1,...,S4* denote the system's servers. Each terminal (client) that connects to a server is fitted with a fingerprint sensor (*FPS*).

The main characteristics of this topology are:

- 1) Because of the large number of required connections, the system's scalability is low.
- 2) System's defect tolerance is high. If a connection is lost, the system doesn't suffer.
- 3) The requests received from standard clients and from other servers are handled almost the same, resulting in code reusability.

C. MOM Based Topology

The MOM topology has a central server, which is used to reduce the number of connections between the production servers, thus increasing its scalability (refer to Fig. 4).

The JMS API supports two models: *point-to-point* and *publisher-subscriber* [15]. The proposed system uses both models, for broadcasting an unsolved request and for replying to a remote request, respectively.

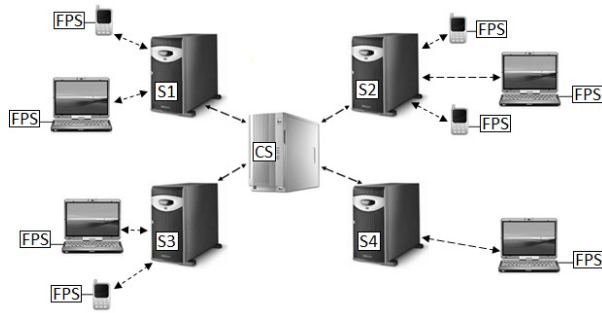


Fig. 4. MOM based servers' topology. $S1, \dots, S4$ denote the system's servers. Each terminal (client) that connects to a server is fitted with a fingerprint sensor (FPS). The central server (CS) is used to reduce the number on connections between the servers

The main characteristics of this topology are:

- 1) By using this topology, a higher scalability of the system is ensured, because a newly entered server has to connect to a single central unit.
- 2) Central server redundancy might be needed in order to increase the system's defect tolerance.
- 3) In the current implementation, queues are used in order to ensure asynchronous communication.

IV. THE SOFTWARE ARCHITECTURE OF THE SYSTEM

A. Software Architecture for the Peer-to-Peer Topology

The servers of the system are multithreaded, being able to handle multiple clients at the same time. If a server is not able to solve a recognition request locally, it forwards the

request, thus becoming a client for all the other servers. There is a slight difference between the regular client handler and the server handler. The second is not allowed to forward an unsolved recognition request for obvious reasons (infinite message loop).

The software architecture of the system has a modular design. The main software modules for the proposed peer-to-peer topology, presented in Fig. 5, are:

- 1) *ClientHandler1* – is a thread responsible for the communication with the client-application. It receives the requests and fetches them to the other modules. It is also responsible for replying back to the client.
- 2) *ServerHandler1* – performs the same tasks as the previous module, except for the fact that it receives only recognition requests.
- 3) *DBManagement* – creates the connection and it implements the access to the database.
- 4) *FingerprintProc&MatchUnit* – is divided into two sub-modules, *FeatExtraction* and *Matcher*, responsible for extracting the fingerprint features and for comparing the inputs with the pre-stored template, respectively.
- 5) *C1ReqSolver* – is the module responsible for forwarding an unsolved client recognition request. It also waits for the responses from the other servers and sends that response to the *ClientHandler* module.
- 6) *S1ReqSolver* – responds to a remote recognition request from another server.

B. Software Architecture for the MOM Based Topology

Some of the software modules, designed for the first topology, were reused for the current architecture (refer to Fig. 6). These modules are: *ClientHandler1*, *ClientHandler2*, *DBManagement*, *FingerprintProc&MatchUnit*.

The other modules of a server application are:

- 1) *C1ReqSolver*, *C2ReqSolver* – verify the recognition result and if the fingerprint is unidentified, they forward the request to the central server through *RemReqSender*. They also receive the remote responses from the central server through *RemRespReceiver*. A remote request is identified in the entire system by two identifiers: *issuing_server_name* and *request_id*.
- 2) *RemReqReceiver* – receives recognition requests from the other servers, sent through the central server.

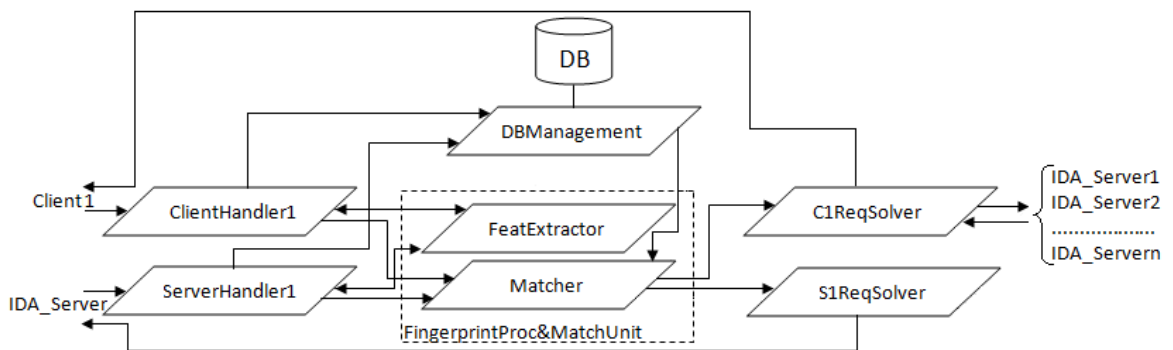


Fig. 5. Server's software architecture in the peer-to-peer topology

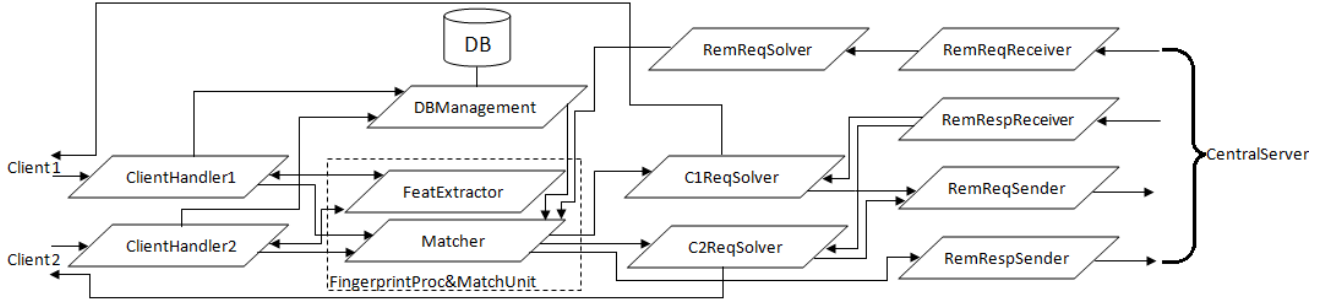


Fig. 6. . Server's software architecture in the MOM based topology

3) *RemReqSolver* – handles all the requests received from the central server.

4) *RemRespSender* – sends the responses for the remote requests to the central server.

The software architecture of the central server application is presented in Fig. 7, considering the fact that three servers are connected to it. The central server software consists of the following modules:

1) *ReqReceiver* – receives remote recognition requests from the other servers. These requests are queued in *URQ* and the *URSem* (the semaphore's value) is incremented.

2) *ReqSender* – takes unsolved requests (one at a time) from *URQ* and sends them to the other servers.

3) *RespReceiver* – is responsible for receiving the responses sent by the central server.

4) *ReqSolver* – decides when a request is solved (whether all the servers responded negatively or a server identified the print). It deletes the request from *URQ* and it queues it in *SRQ*, incrementing and decrementing the semaphores (*SRSem* and *URSem*, respectively).

V. THE VERIFICATION OF THE MODELS

Before implementing the proposed applications, a verification of the models is required, in order to avoid deadlocks, since each distributed application deals with a fairly large number of synchronized threads. The best suited method for this purpose is the use of Petri Nets.

A. Peer-to-Peer Model Verification

The Petri Net corresponding to the first proposed model is presented in Fig.8. The logic of the net is the following: the token from the PSC starts the client application. The

execution of TC2 transition is equivalent to a user recognition request and it puts a token into PPC. After TC1 executes, a token is put in PAC (*request_accepted place*). Afterwards, TC3 is able to be executed, putting a token in PTC (*request_ready_to_send place*). The execution of TR1 corresponds with the request being sent through the network. The server receives the token (*the request*) in PACS1, it verifies the fingerprint image file in PVFS1 and if it can't be processed an error message is sent back to the client (through TS13-PTRS1-TR2 sequence). If the file is acceptable, the template is extracted (in PETS1). It is then compared (in PRS1) against the n templates stored into the database (PTBDS1). If the fingerprint is identified, the recognition result is sent back to the client (through PERS1-TS17-PTRS1-TR2 sequence), else the request is forwarded to the other servers, which run the same as Server1. The client is considered to have received the response, when a token is put in PPR.

B. MOM Based Model Verification

The client and the server to which it is connected have the same Petri Net as in the first model. In the second model (see Fig. 9), if the print is not recognized, it is sent from Server1 (PERS1), through the network (TR3), to Central Server (PRCSC) and queued into the *unsolved requests' queue* (PCCNSC). Each unsolved request is sent to Server2 and Server3, in the receiving order (through PTCSC-TR4). The responses received from Server2 and Server3 are combined (into PRCSC) into a final response, queued into the *solved requests' queue* (PCCRSC). From here, the response is sent back to Server1 and finally to the client (through PTCSC-TR7-PPRS1-TS16-PERS1-TS17-PTRS1-TR2 sequence).

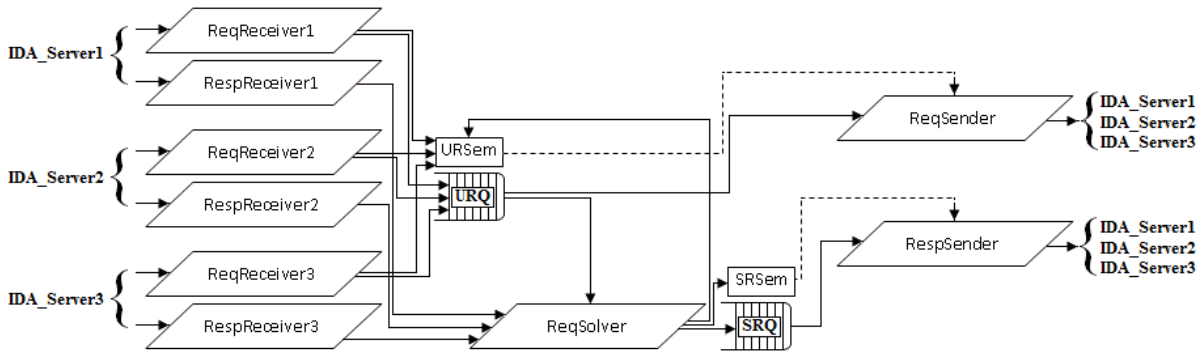


Fig. 7. Central server's software architecture in the MOM based topology

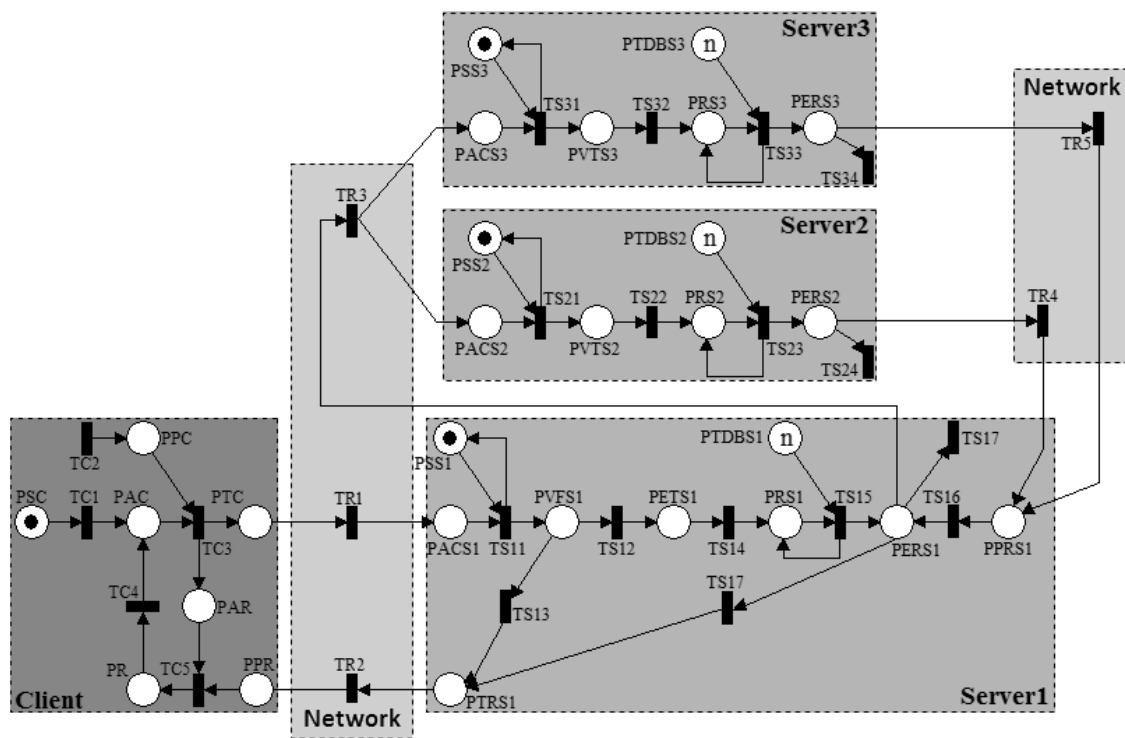


Fig. 8. Verification of the peer-to-peer model by using Petri Nets.

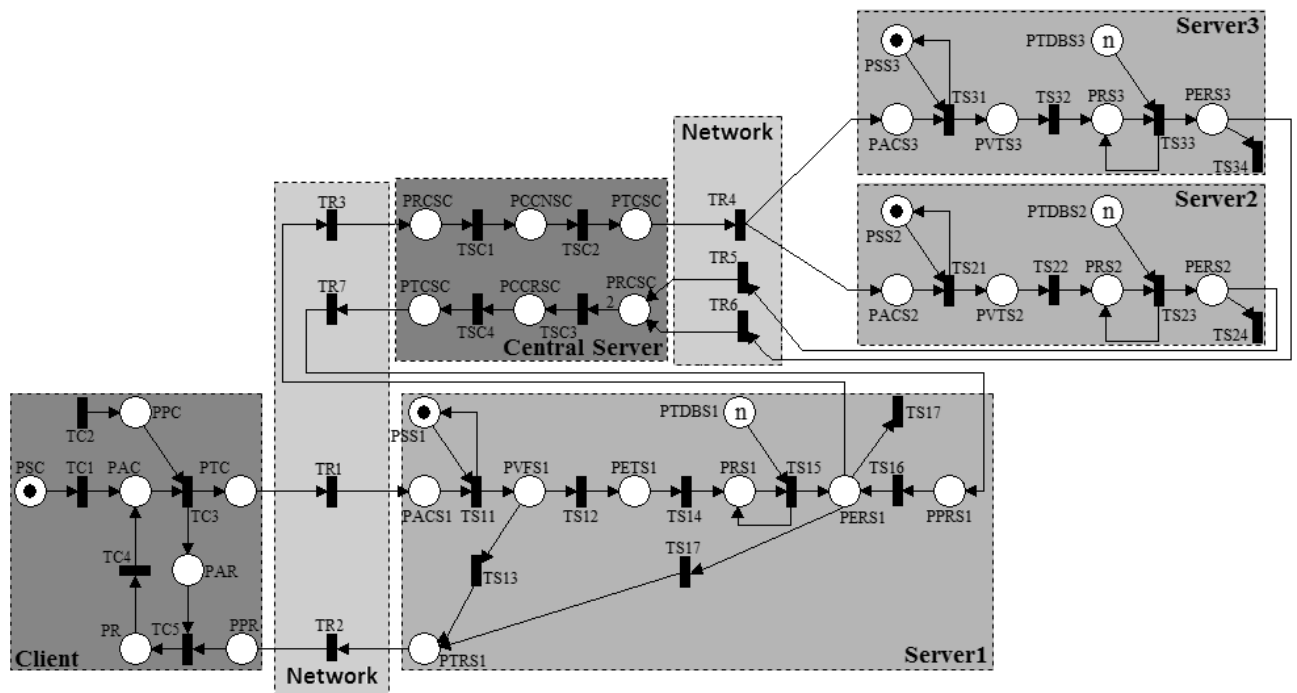


Fig. 9. Verification of the MOM based model by using Petri Nets.

VI. TESTS AND RESULTS

Both architectures, presented in this paper, were implemented in *Java 2 Standard Edition*. The client application used to test the proposed systems was implemented in the same programming language.

The chosen database management system is *MySQL Server 5.5*. *DBManagement* module uses *Java Database Connectivity API* (the *JDBC driver*) to connect to the DB.

The *FingerprintProc&MatchUnit* module, responsible for the fingerprint feature extraction and for matching the

templates, is based on *Neurotechnology Verifinger 6.0 SDK* libraries.

A. FingerprintProc&MatchUnit Temporal Performances

The module responsible for fingerprint feature extraction and matching is the same for both proposed topologies. It was tested on a PC having a dual core processor (*Intel(R) Core(TM)2 Duo T7250 @ 2.00GHz*). Tests have shown that the main two module functions have the following temporal performances:

- 1) feature extraction + template file generation: 190ms,
- 2) matching 2 templates: 0,7ms.

Because Verifinger 6.0 Standard SDK provides a single feature extraction unit and a single matching unit, received concurrent recognition requests are processed sequentially. Fig. 10.a) shows the time durations needed to solve a variable number of concurrent identification requests.

B. Temporal Performances of the Two Architectures

For comparing the proposed architectures, two fingerprint processing units (servers), connected in both ways in a LAN, were considered. A variable number (from 1 to 10) of remote identification requests are sent to the first server at the same time. The 100 enrollments database of the first server contains no match and the requests are forwarded to the second server, which performs around 100 comparisons for each request. The time durations needed for the response to reach the clients are presented in Fig. 10.b).

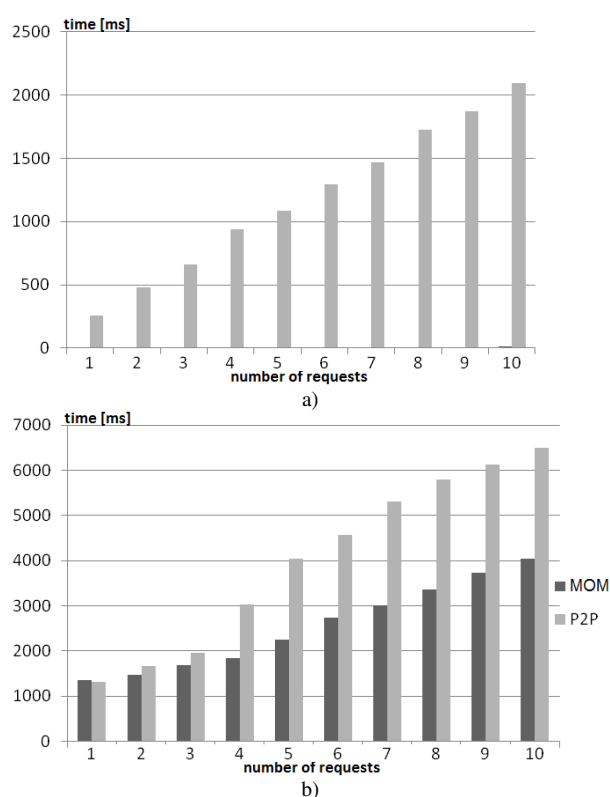


Fig. 10. a) Time durations for concurrent identification requests. The database was created so that all the requests can be solved locally (the server can find a match in its own database) and the *Matcher* module performs around 100 comparisons before finding a match for each request. b) Time durations for concurrent identification requests of the proposed MOM and peer-to-peer (P2P) architectures.

VII. CONCLUSION

This paper proposed two different topologies for interconnecting the processing units (or servers) of a distributed fingerprint recognition system. The central components of the system are the fingerprint processing and matching modules (implemented on each server). Tests have shown that a bottleneck occurs because of the fact that the

libraries that were used have a single processing/matching unit, thus reducing the temporal performances that could be provided by a multiprocessor server.

Tests have also shown that the MOM based architecture is superior to the peer-to-peer architecture, surprisingly obtaining better results, even if the requests/responses pass through a central server. This is because a server, implemented for the peer-to-peer architecture, reconnects (recreates the socket) to the other servers each time it has to forward an unsolved request. Another important MOM topology advantage is its higher scalability.

REFERENCES

- [1] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *Circuits and Systems for Video Technology*, IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, Jan. 2004, pp. 4-20.
- [2] *Biometrics Market and Industry Report 2009-2014*. International Biometric Group, New York, NY (PRWEB), Nov. 12, 2008.
- [3] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer, New York, 2003, pp. 8-12.
- [4] D. Maltoni, "A Tutorial on Fingerprint Recognition", *Biometric Systems Laboratory - DEIS - University of Bologna*, 2005, pp. 1-3.
- [5] M. Hulea, A. Astilean, T. Letia, R. Miron, and S. Folea, "Fingerprint Recognition Distributed System," IEEE International Conference on Automation, Quality and Testing, Robotics, AQTR 2008-THETA 16th edition, May 22-25 2008, Cluj-Napoca, Romania, Tome III, pp.423-428.
- [6] O. Babaoglu, and K. Marzullo, "Consistent Global States of Distributed Systems: Fundamental Concepts and Mechanisms," Laboratory of Computer Science, University of Bologna, Italy, Tech. Rep. UBLCS-93-1, Jan. 1993, pp. 1-5.
- [7] H. Xiao, *Middleware and its Applications Research Survey*, CISC 833 Advanced Operating System, Mar. 2005. Available: <http://research.cs.queensu.ca/home/xiao/DS/index.html>
- [8] S. Krakowiak, *Middleware Architecture with Patterns and Frameworks*. Distributed under a Creative Commons license, Feb. 2009, pp. 1-6. Available: <http://creativecommons.org/licenses/by-nc-nd/3.0/>
- [9] D. A. Menasce, "MOM vs. RPC: communication models for distributed applications," *Internet Computing, IEEE*, vol. 9, 2005, pp. 90 - 93, ISSN: 1089-7801.
- [10] M. Kaddour, and L. Pautet, "Towards an adaptable message oriented middleware for mobile environments," *3rd IEEE workshop on Applications and Services in Wireless Networks*, Berne (Suisse), 2003, pp. 251-262.
- [11] H. Ammar, X. Zhen, D. Nassar, and Y. Jin, "Web-Based Test Bed for Fingerprint Image Comparison," *The World Internet and Electronic Cities Conference*, May 2001. KISH ISLAND, IRAN, Received Best Paper Award.
- [12] Y. Chung, D. Moon, T. Kim, and J. W. Park, "Workload Dispatch Planning for Real-Time Fingerprint Authentication on a Sensor-Client-Server Model," in *Proc. 5th International Conf. Parallel and Distributed Computing: Applications and Technologies*, Dec. 2004, pp. 831-837.
- [13] T. Bulka, C. Agarwal, and P. Pozdnyakov, *Secure Mobile Fingerprint Identification System*, Santa Clara, CA, 2005. Available: <http://www.scu.edu/engineering/cse/ugrad/upload/COEN-2005-PROJECT-01.pdf>
- [14] JSR 914: JavaTM Message Service (JMS) API, Nigel Deakin, ORACLE, (02.dec.2003). Available: <http://jcp.org/aboutJava/communityprocess/final/jsr914/index.html>
- [15] L. Wang, S. Wei and Z. Zhang, "Research and Design of Dead Message Handling Mechanism in JMS," *Journal of Communication and Computer*, vol. 7, no. 6, Jun. 2010, ISSN 1548-7709, USA, pp. 14-18.

Modeling and Simulation of the billet's heating process in a furnace with rotary hearth

V. Mureşan, M. Abrudean, and T. Coloşi

IEEE Members

Abstract—This paper presents a method for the numerical simulation of the billets temperature into an industrial furnace. A very important aspect in approaching the problem was the fact that there were considered both the temperature variations from the exterior medium of the billet from the furnace and the temperature variations from its volume.

I. INTRODUCTION

THE furnace with rotary hearth is one of the basic aggregates in the technologic flow for manufacturing seamless pipes. Its purpose is to warm up the blocks of billets, previously cut at the required lengths, from the environment's temperature to the rolling temperature, leading the material from the elastic domain to the plastic one. The furnace is heated using methane gas.

The furnace can be charged on one row or on two rows, according to the length of the billets that are charged. In case of one row charging, the billet should be placed with the centre right on the circle that represents the middle of the hearth [1].

The furnace has five regulating temperature sectors, which have a total of 48 burners, a sector for preheating and a sector for loading-unloading (sector 6). While working in service duty, sector 6 is thermally inactive. The temperatures corresponding to each sector are controlled by adjusting the gas flow on the corresponding pipes. The temperature in each sector is measured using 2 thermocouples (for redundancy). The signal from the thermocouples output is sent to the flow controllers. Each controller (there is one for each of the 5 active sectors) is of PID type and controls the flow on the pipes operating some pneumatic actuators as valves situated on the five supply gas pipes.

The actual control system can assure a constant temperature in the exterior area of the billets for each sector. The thermocouples are placed in the centre of the active sector of the furnace. In this context, the value of temperature can be strictly controlled only in the thermocouples neighborhood. Due to the turbulences from the furnace, in general, it can be considered that the temperature on the median circle of the hearth (the equidistant circle both from the internal wall and external wall of the furnace) is constant for each sector. However, there are temperature variations on the median circle inside of each sector and they influence the heating of the billets.

In this paper is proposed a solution to simulate the temperature of the billets in the furnace, considering the fact that they are passing through 6 sectors (the preheating sector and the five active sectors) with different temperatures. It has been considered both the temperature variation between two consecutive sectors and from the billets volume.

II. MODELING THE THERMAL PROCESSES OF HEATING THE BILLETS

Modeling the thermal processes of heating the billets, in this case, can be done considering the temperature variation between the billet exterior area and the billet interior part. In this way on the exterior area of the billet we can measure the highest temperature and in the interior part of the billet (in the billet symmetry axis area) we can measure the lowest temperature. This phenomenon appears due to the fact that the heating source has a direct effect on the exterior area of the billet. The dependence of temperature on length can mathematically be expressed as an exponential function.

Also it has to be taken in consideration the temperature variation between the furnace interior and exterior wall, variation that influences the billet too. In the internal centre of the furnace we can measure the highest temperature and near the walls of the furnace we can measure the lowest temperature of that sector [2].

The analogue model of the processes with the distributed parameters can be expressed using equations and systems of equations with partial derivatives. In this way, besides the time variable (t), in the cartesian space case we introduce the independent variables (p), (q) and (r) in the model of processes. The axes that determine the cartesian space are $0p$, $0q$ and $0r$.

In figure 1 is presented a billet in the furnace. It is considered a billet with the length $L=2m$ and with the diameter $D=0,15m$. The value of the billet length implies that the furnace is charged on one row. The $0p$ axis is the vertical line on the billet that passes trough its centre. The $0r$ axis is parallel with the symmetry axis of the billet and intersects the $0p$ axis at exterior area of the billet. Due to the turbulences from the inside of the furnace and due to the small height of work space it is not necessary to discuss the problem on direction $0q$ (the vertical line on $0p$ and $0r$) and is not necessary to consider a temperature variation in the vertical plane of the furnace. In this case, if we consider a circle with the centre on the symmetrical axis of the billet,

the value of the temperature in every point of the circle is the same.

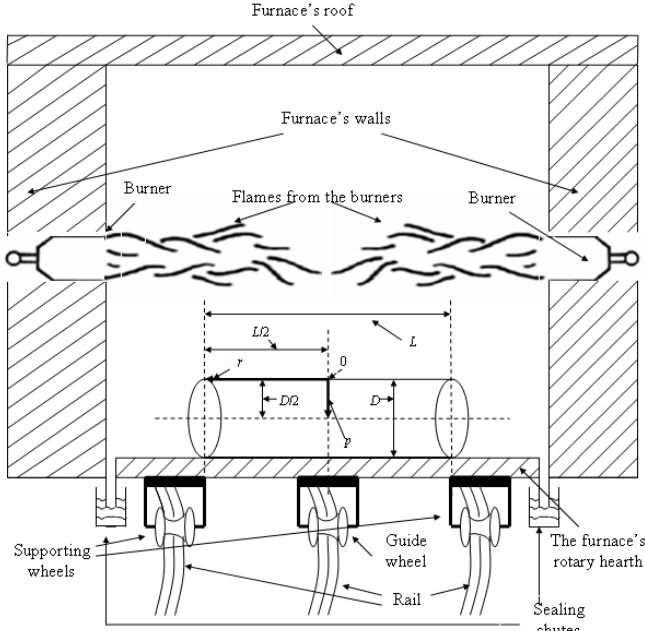


Fig. 1. Heating of a billet in the furnace with rotary hearth.

In the model of the thermal process of heating the billet is considered only the temperature variation on the $0p$ axis. This simplification it is possible because the weight of the temperature variation on the axis $0r$ is very small comparing to the $0p$ axis case. Also this simplification can be applied because the length of the billet is smaller than the hearth width (5m), the temperature variation being stronger in the furnace's walls neighbourhood.

The five thermal processes corresponding to the active sectors of the furnace are modeled through equations with partial derivatives. All of them have the same structure (the equation order is 2 and the number of independent variables is 2) but have different coefficients. Introducing variable (p), the equation with partial derivatives that describes the process work (PDE II.2) is:

$$a_{00} \cdot y + a_{10} \cdot \frac{\partial y}{\partial t} + a_{01} \cdot \frac{\partial y}{\partial p} + a_{20} \cdot \frac{\partial^2 y}{\partial t^2} + a_{11} \cdot \frac{\partial^2 y}{\partial t \partial p} + a_{02} \cdot \frac{\partial^2 y}{\partial p^2} = \varphi(t, p) \quad (1)$$

In relation (1) the functions $y(t, p)$ (the billet's temperature) and $\varphi(t, p)$ respect Cauchy conditions of continuity. Also the $a_{...}$ coefficients can be determined with the following relations: $a_{00} = 1$, $a_{10} = T_B + T_{BB}$, $a_{20} = T_B \cdot T_{BB}$, $a_{01} = P_1 + P_2$, $a_{02} = P_1 \cdot P_2$ and $a_{11} = (T_B + T_{BB}) \cdot (P_1 + P_2)$, where T_B is the heating time constant of the considered billet and P_1, P_2 can be called "the length constants" of the billet. P_1 and P_2 are the parameters of the exponential function that models the temperature variation related with the axis $0p$. For a higher generality and to have the possibility to introduce later in the

model the heating time constant of the sector, we worked with an equation (PDE II.2). As the process has only a significant time constant T_B , the time constant T_{BB} has got an insignificant value controlled through the very small value of λ_T parameter ($\lambda_T = 0,01$; the parameter λ_T is a coefficient that shows the proportion between the two time constants). If the following notation is introduced:

$$y_{TP} = \frac{\partial^{T+P} y}{\partial t^T \partial p^P}, \quad (2)$$

where $T=0,1,2,\dots$, and $P=0,1,2,\dots$, relation (1) can be written as:

$$a_{00} \cdot y_{00} + a_{10} \cdot y_{10} + a_{01} \cdot y_{01} + a_{20} \cdot y_{20} + a_{11} \cdot y_{11} + a_{02} \cdot y_{02} = \varphi_{00} \quad (3)$$

The schematic representation of the process is realized in figure 2, where the significance of the notations is:

u_0 - actuator signal representing the sector's temperature;

$\tilde{u}_0$ - disturbance signal (the temperature variation due to the variations introduced by the gas supplying system) and

y_{00} - output signal (billet's temperature).

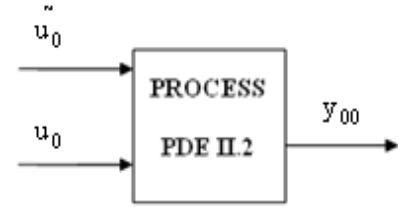


Fig. 2. The black box representation of the process.

Only a subscript attached to the functions from figure 2 signifies the derivative order of those elements related to the independent variable (t).

The analogue modeling of the system presented in figure 2 starts from the following equation:

$$\begin{cases} y_{00} \\ y_{10} \\ y_{20} = \frac{1}{a_{20}} \cdot [\varphi_{00} - (a_{00} \cdot y_{00} + a_{10} \cdot y_{10} + a_{01} \cdot y_{01} + a_{11} \cdot y_{11} + a_{02} \cdot y_{02})] \end{cases} \quad (4)$$

From the relation (4) are obtained the elements of the state vector x that in transposed form is being presented in relation (5):

$$x^T = \begin{bmatrix} y_{00} & y_{10} \end{bmatrix} \quad (5)$$

The M_{pdx} (the matrix of partial derivatives of the state vector) is presented in relation (6). We have considered the values $M=8$ and $N=5$, definitive for the dimension of M_{pdx} .

$$\begin{array}{c}
 \begin{array}{|c|c|} \hline \mathbf{x} & \mathbf{x}_p \\ \hline \mathbf{x}_T & \mathbf{x}_{TP} \\ \hline \end{array} \\
 \begin{array}{l} \xrightarrow{1} \xrightarrow{M} \\ \uparrow n=2 \\ \downarrow N=5 \end{array} \\
 \mathbf{M}_{pdx} = \begin{array}{|c|c|c|c|c|c|c|c|} \hline y_{00} & y_{01} & y_{02} & y_{03} & y_{04} & y_{05} & y_{06} & y_{07} & y_{08} \\ \hline y_{10} & y_{11} & y_{12} & y_{13} & y_{14} & y_{15} & y_{16} & y_{17} & y_{18} \\ \hline y_{20} & y_{21} & y_{22} & y_{23} & y_{24} & y_{25} & y_{26} & y_{27} & y_{28} \\ \hline y_{30} & y_{31} & y_{32} & y_{33} & y_{34} & y_{35} & y_{36} & y_{37} & y_{38} \\ \hline y_{40} & y_{41} & y_{42} & y_{43} & y_{44} & y_{45} & y_{46} & y_{47} & y_{48} \\ \hline y_{50} & y_{51} & y_{52} & y_{53} & y_{54} & y_{55} & y_{56} & y_{57} & y_{58} \\ \hline y_{60} & y_{61} & y_{62} & y_{63} & y_{64} & y_{65} & y_{66} & y_{67} & y_{68} \\ \hline \end{array} \\
 \begin{array}{l} \xrightarrow{1} \xrightarrow{M=8} \\ \uparrow n=2 \\ \downarrow N=5 \end{array}
 \end{array} \quad (6)$$

The matrices and vectors that occur in relation (6) are:

- the state vector $\mathbf{x}$ of the whole system with dimension $\mathbf{x}(2 \times 1)$;
- the vector of partial derivatives related to time (t) of the state vector $\mathbf{x}_T$ with dimensions $\mathbf{x}_T(5 \times 1)$;
- the matrix of partial derivatives related to independent variable (p) of the state vector $\mathbf{x}_p$ with dimension $\mathbf{x}_p(2 \times 8)$;
- the matrix of partial derivatives related to time (t) and to the independent variable (p) of the state vector $\mathbf{x}_{TP}$ with the dimension $\mathbf{x}_{TP}(5 \times 8)$.

Thus it results that the matrix $\mathbf{M}_{pdx}$ has the dimension $(\mathbf{M}_{pdx}(7 \times 9))$.

The analytical solution that verifies relation (1) is:

$$y_{00AN}(t, p) = K_y \cdot F_{0T}(t) \cdot F_{0P}(p) \cdot u_0(t), \quad (7)$$

where

$$F_{0T}(t) = 1 - \frac{T_B}{T_B - T_{BB}} \cdot e^{-\frac{t}{T_B}} - \frac{T_{BB}}{T_{BB} - T_B} \cdot e^{-\frac{t}{T_{BB}}} \quad (8)$$

and

$$F_{0P}(p) = \frac{P_1}{P_1 - P_2} \cdot e^{-\frac{p}{P_1}} + \frac{P_2}{P_2 - P_1} \cdot e^{-\frac{p}{P_2}}. \quad (9)$$

In the relation (7) the proportionality constant of the process $K_y = 1$ because in the steady state regime the temperature value of the exterior area of the billet becomes equal with the sector's temperature. The work of the real process is accurately approximated through the simulation of the analytical solution.

The analytical solution is used only to calculate the initial conditions (IC) (for $t=t_0=0$) for the $\mathbf{M}_{pdx}$ matrix. After computing $(y_{00AN}(0, p), y'_{00AN}(0, p), y_{0\rho AN}(0, p), y'^{(t)}_{0\rho AN}(0, p))$ it results: $y_{00,IC}=0, y_{10,IC}=0, y_{0\rho,IC}=0$ and $y_{1\rho,IC}=0$, where $\rho \in \{1, 2, \dots, M\}$. The vector $\mathbf{x}_{T,IC}$ is obtained from the state vector, if we derive successively its elements in relation to the independent variable time (t). The

first element of the vector ($\mathbf{x}_T$) is the pivot element. There is also the condition that $N \geq n$. The matrix $\mathbf{x}_{TP,IC}$ is obtained from the elements of the matrix ($\mathbf{x}_p$) through their successive derivation in relation to the independent variable time (t), starting from the left side to the right side of a line and finishing the calculation for a line, before passing to the calculation of the line below. The elements that result represent partial derivatives in relation to time (t) of the elements of matrix ($\mathbf{x}_p$), and $T \in \{n, n+1, \dots, n-1+N\}$.

After doing these calculations we can make the matrix $(\mathbf{M}_{pdx})$ for the initial conditions $((\mathbf{M}_{pdx})_{IC})$ that correspond to the start sequence ($k-1$).

III. NUMERICAL SIMULATION OF THE BILLET'S TEMPERATURE USING TAYLOR SERIES

In order to advance from sequence ($k-1$) to sequence (k) we need to use the Taylor series. According to the method that has been presented in [3], in order to calculate the elements of the state vector ($\mathbf{x}$) and of the matrix with partial derivatives of the state vector ($\mathbf{x}_p$), at moment (k), we should apply the formula:

$$y_{g,k} \cong y_{g,k-1} + \sum_{T=\tau+1}^{N+n-1} \frac{\Delta t^{T-\tau}}{(T-\tau)!} \cdot y_{Tj,k-1}, \quad (10)$$

where $\tau \in \{0, 1, \dots, n-1\}$, $j \in \{1, 2, 3, \dots, M\}$, $n=2$, $N=5$ and $M=8$. The formula presented above is valid if the integration is being done in relation to time (t). Due to the fact that in relation (6) the maximum derivation order in relation to time (t) of all the functions is 6 (y_{6i}), where $i \in \{1, 2, \dots, M\}$ and $M=8$ and due to the fact that $N+n-1=5+2-1=6$, the truncation errors problem does not appear.

The elements ($y_{g,k}$) are given by the relation:

$$y_{g,k} \cong y_{g,k-1} + \sum_{T=\tau+1}^6 \frac{\Delta t^{T-\tau}}{(T-\tau)!} \cdot y_{Tj,k-1}. \quad (11)$$

In the relation (11) $\tau \in \{0, 1\}$, $j \in \{0, 1, 2, \dots, 8\}$. Using the calculated values we get the elements of the vectors ($\mathbf{x}_T$) and ($\mathbf{x}_{TP}$) using the algorithm presented in the previous chapter. The numerical simulation is finished when $t \geq t_f$ (final simulation time). In all the relations from this subchapter we considered that the integration step (Δt) has a value that is small enough, so that the numerical integration is being done correctly. The value of (Δt) is in general smaller than the 10th part of the smaller time constant from the system (in this case T_{BB}). This value is chosen according to the specific of the application [4].

IV. THE SIMULATION RESULTS

The simulation application was developed in MATLAB environment [5].

After simulation we have compared the response that

resulted through numerical integration and the analytical response of the system. The formula of the relative error cumulated in percents is presented in relation (12):

$$\text{recpy} = 100 \cdot \left(\sum_{i=1}^{i_f} |y_{Ni} - y_{Ai}| \right) / \left(\sum_{i=1}^{i_f} |y_{Ai}| \right). \quad (12)$$

In the relation (12): i represents the integration sequence; i_f represents the integration final sequence; y_{Ni} represents the value of the system response resulted through numerical integration, that corresponds to sequence i ; y_{Ai} represents the value of the system analytical response that corresponds to sequence i . The value that resulted after calculating the error is in percents.

A. The simplified case (the temperature is considered homogenous in each sector)

The value of the temperature is controlled in the centre of each sector, in the thermocouples area. In this subchapter the following simplification is made: in a sector the temperature's value is constant along the medium circular arc of the hearth (the circular arc that is equidistant both from the internal and the external wall of the furnace). In this hypothesis, in the close areas of the neighbor sectors of the considered sector, the value of the temperature is equal to the controlled value of the temperature from the centre of that sector. These remarks are valid for all five active sectors and for the preheating sector. It results the fact that at the sectors borders a big "jump" of temperature appears, phenomenon that does not correspond in large with the real case.

The objective of the experimental part of this paper is to simulate the billet's temperature in all the period of its residence in the furnace (from the charging door to the discharging door). If the previous mentioned simplification is applied, the input signal (the temperature of the exterior medium of the billet, inside the furnace: u_0) is composed by six consecutive step signals (corresponding to a preheating sector + 5 active sectors). For the preheating sector, sector that does not have an independent heating source, the value of the corresponding input signal is $u_{00} = 900^\circ\text{C}$ (the approximate value of the burned gases in the centre of the preheating sector, burned gases that are evacuated from the five active sectors through a tap hole that is placed in the preheating sector). For the active sectors, the values of the step signals correspond to the values of the outputs of each automatic control system in steady state regime. Assuming that the control systems functioning is correct, the values imposed by the technological specifications are: for sector 1, $u_{01} = 1240^\circ\text{C}$, for sector 2, $u_{02} = 1280^\circ\text{C}$, for sector 3, $u_{03} = 1300^\circ\text{C}$, for sector 4, $u_{04} = 1310^\circ\text{C}$ and for sector 5, $u_{05} = 1300^\circ\text{C}$. The temperature of the fifth active sector is maintained smaller than in the fourth sector case because is necessary the billet's temperature homogenization (at the

piercer's input the temperature of the billet's symmetry axis area have to be bigger than the temperature of the billet's exterior area). If all the specifications are respected, a billet of 150 mm diameter have to be heated in the furnace for 90 min [6]. The area of the six sectors are equal and the geometrical form of them is the same. It results the fact that the billet covers every sector in $90/6=15$ min. In this context, in the simulation application, each step signal (u_{0i} , where $i \in \{0, \dots, 5\}$) is applied for 15 min. The imperfections due to the fact that the hearth is rotated with constant steps (with the purpose of charging-discharging the furnace) are insignificant and are not considered in this paper.

If the billet's initial temperature is 0°C (the ambient temperature) and the simulation is made in the symmetry centre of the billet (in figure 1: at 7,5 cm distance of the origin on the Op axis), the comparative graph between the analytical response (resulted by simulating the analytical solution from relation 7) and the numerical response of the billet heating process in the furnace is presented in figure 3.

On the graph we cannot differentiate the two responses due to the very small error between them.

Also, on the graph, it can be observed that the billet's temperature "jumps" at the passing from a sector to the next one (at every 15 min). The "jump" is bigger at the limit between the preheating sector and sector 1 due to the big differences of the temperature between them and smaller at the limit between sector 4 and sector 5 due to the fact that the temperatures in these two sectors are almost equal.

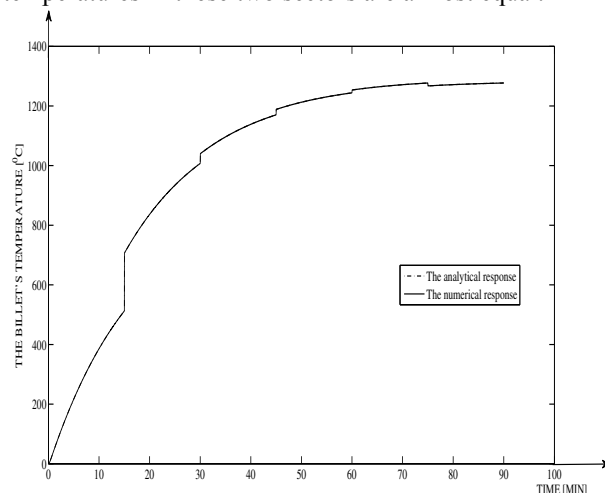


Fig. 3. The comparative graph between the analytical and the numerical response of the billet heating process in the furnace.

The cumulated error in percents between the analytical and the numerical response of the process, for the studied case, is presented in figure 4.

The maximum value of the cumulated error in percents is of $3,97 \cdot 10^{-5} \%$, right at the beginning of the numerical integration, when the temperature variations related to time are high and we get high values of the derivatives, then it falls suddenly and remains constant at the value of $3,65 \cdot 10^{-7} \%$. Also the passing from a sector to the next one

is made without significant variations of the error.

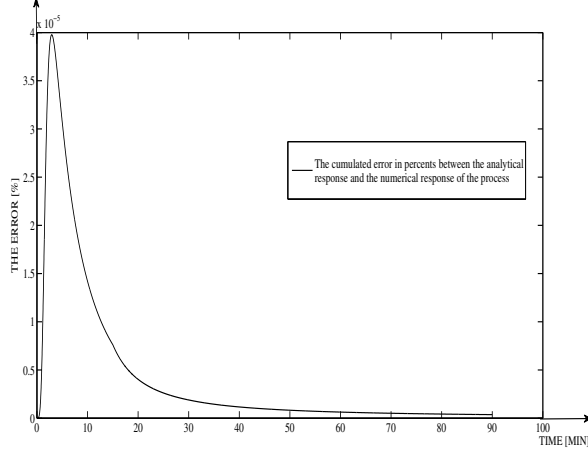


Fig. 4. The cumulated error in percents between the analytical and the numerical response of the process.

These results that are very close to 0 show the very good performances of the method of numerical simulation.

B. The real case (the temperature is not considered homogenous in each sector)

In this chapter the temperature variation inside of the furnace's sectors is considered. Along the median circular arc of the hearth, from the charging door to the inclusively active sector 4, the gases temperature in the furnace has an increasing evolution. Also, from the limit between the active sector 4 and active sector 5, until to the discharging door, the temperature has a decreasing evolution.

The first problem is the fact that the temperature evolution along the median circular arc of the furnace can be expressed like a function depending on the length variable. In the numerical simulation application, the temperature have to be introduced like a function depending on the time variable. In the previous chapter was shown the fact that the billet covers every sector in 15 min. If we consider that the hearth's stops do not influence in a significant manner the billet's heating process (their weight tends to 0 in the treated context), the length of each sector median circle arc (9,63 m) can be associated with the residence time of the billet in every sector (15 min).

The temperature variation in each sector can be modeled using a combined signal step + exponential function. In this case the input signal (the sector's temperature) becomes:

$$u_{0i} = u_{A_i} + u_{B_i} \cdot \left(1 - \frac{\tau_{1i}}{\tau_{1i} - \tau_{2i}} \cdot e^{-\left(\frac{t-i\tau}{\tau_{1i}}\right)} - \frac{\tau_{2i}}{\tau_{2i} - \tau_{1i}} \cdot e^{-\left(\frac{t-i\tau}{\tau_{2i}}\right)} \right) \quad (13)$$

, where u_{A_i} and u_{B_i} represent temperatures, τ_{1i} and τ_{2i} are time constants, $i \in \{0, \dots, 5\}$ (the sector's number; $i=0$ for the preheating sector), $\tau=15$ min and t represents the time variable. For each sector, the values of u and τ_i coefficients were determined using an identification method through interpolation. For the case of the preheating sector ($i=0$) it resulted: $u_{A_0}=200^\circ\text{C}$, $u_{B_0}=800^\circ\text{C}$, $\tau_{10}=2.1748$ min,

$\tau_{20}=3,2622$ min, or for the case of active sector 5 ($i=5$) it resulted: $u_{A_5}=1311,7^\circ\text{C}$, $u_{B_5}=-22,7^\circ\text{C}$ (the temperature's value is decreasing in this sector), $\tau_{15}=7$ min, $\tau_{25}=10,5$ min.

If the simulation is made in the same conditions like in the previous case, with the difference that the input signal is composed by six consecutive signals with the form described by relation (13), the comparative graph between the analytical response and the numerical response of the billet heating process in the furnace is presented in figure 5.

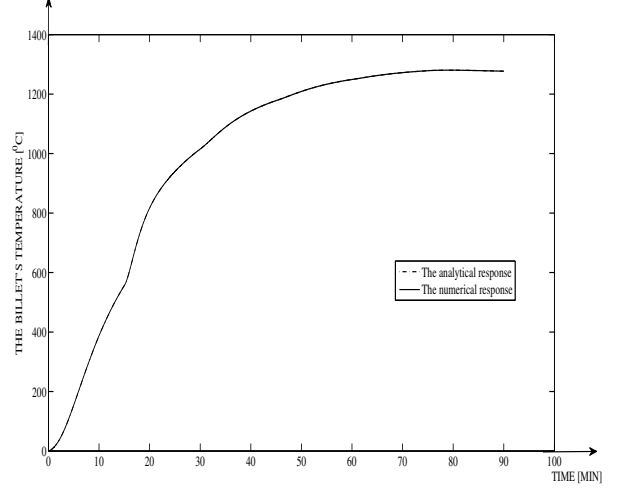


Fig. 5. The comparative graph between the analytical and the numerical response of the billet heating process in the furnace, if the input signal from relation (13) is applied.

As opposed to the case from figure 3, the temperature “jumps” are not appearing, the process's functioning being described in a very precise manner. In this case, on the graph we cannot differentiate the two responses either.

The corresponding cumulated error in percents, between the two responses, is presented in figure 6.

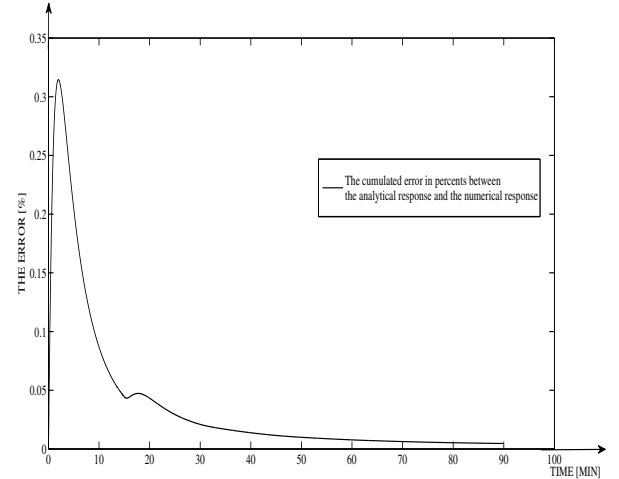


Fig. 6. The cumulated error in percents between the analytical and the numerical response of the process, for the case presented in figure 5.

In this case, the maximum value of the cumulated error in percents is 0,314%, right at the beginning of the numerical integration, from the same reasons like in the previous case,

then it falls suddenly and remains constant at the value of $4,6 \cdot 10^{-3} \%$. The values of the error are very small considering the fact that they are expressed in percents, but they are bigger than in the previous case (the process is treated in much more severe conditions). On the graph, it can be observed that only the passing of the billet from the preheating sector to the first active sector (at 15 min after starting the simulation) introduces a determinable local increasing of the error. This phenomenon appears due to the fact that the temperature variation is quite big in the first part of the active sector 1.

Another important problem is the problem of the active sector 5. This sector's purpose is to decrease the temperature of the billet in its exterior part, in such a way that the temperature in its symmetry axis area to become bigger. In this case the billet can be seen like a heating source. In order to simulate this phenomenon is necessary to introduce a correction in the billet's heating process model, more exactly a correction of the proportionality constant. This correction is made along Op axis from figure 1 and it is presented in the relation (14):

$$K'_y = K_y + 10^{-3} \cdot \left(\frac{0,075 - p}{p + 2,3} \right), \quad (14)$$

, where $K_y = 1$ and p (expressed in meters) is the distance on the Op axis direction from the origin of the considered cartesian system to the point where the simulation is made. Using this correction, the simulation can be made only above the symmetry axis of the billet, but the temperature's value is considered symmetrically distributed related to the centre of the billet. In other words, if we consider a circle inside the billet, with the centre on its symmetry axis, temperature value is constant on that circle (due to the turbulences from the interior part of the furnace).

The simulation is made in the same conditions presented in this subchapter, but the correction from relation (14) is introduced. The comparative graph between the process response if the simulation is made at the exterior area of the billet (in the origin of the cartesian system) and the process response if the simulation is made in the internal centre of the billet (at 7,5 cm distance by the origin of the cartesian system in the direction of Op axis) is presented in figure 7.

In the figure 7, the process response in the internal centre of the furnace is represented with dash-dot line and the process response in the origin of the cartesian system is represented with continuous line. The graphs shows the fact that the temperature in the exterior area of the billet is higher than the temperature in the internal centre of the billet in the preheating sector and in the active sectors 1, 2, 3 and 4, but in the active sector 5 the homogenization phenomenon appears and the temperatures invert their values.

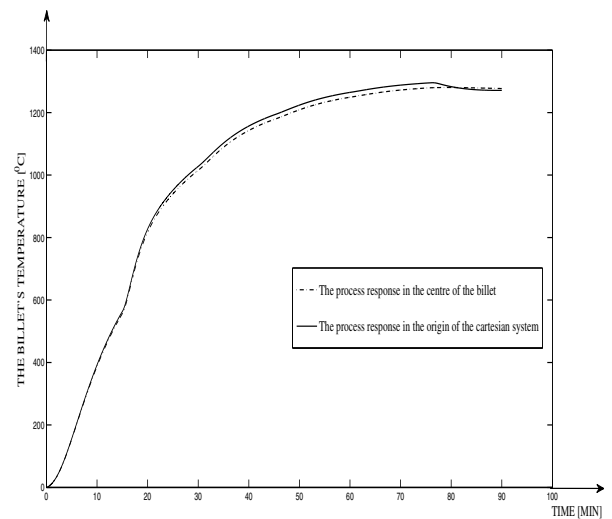


Fig. 7. The comparative graph between the process response in the internal centre of the furnace and the process response in the origin of the cartesian system.

V. CONCLUSION

1. This paper presents a method of analogue modeling and numerical simulation of the thermal process of heating the billets in the furnace with rotary hearth.

2. In order to check the performances of numerical integration, we used the analytical model of the process as comparison standard. In all cases that were studied, the performances were very good and the cumulated relative error in percents had very small values. In all the simulations the value of the integration step was $\Delta t = 0,001$ min.

3. The simulations were made considering the billet's entire layout from the furnace. Also the simulations were made considering both the simplified case (fig. 3) and the real one (fig. 5). The homogenization phenomenon of the billet's temperature from the fifth active sector of the furnace was studied too.

4. The use of the equations with partial derivatives allows a representation that is very close to reality of the way the thermal processes of heating the billets work. The model of the billet's heating process was elaborated with the purpose of including this process in an automatic control system.

REFERENCES

- [1] I. Pența, D. Szeredai, F. Major, "The technology of preparing the loading of the CVR and perforating the blocks in the Mannesmann perforator", Bucharest, 1987, 157 pages.
- [2] V. Mureșan, M. Abrudean, T. Colosi, "Temperature control of the billets in a furnace with rotary hearth", Proceedings 18th International Conference on Control Systems and Computer Science (CSCS 18) – Vol 1, May 24-27, 2011, București, România, pp. 347-354.
- [3] Colosi T., M.-L. Ungureșan, E. Dulf, R. C. Cordos, "Introduction to Analogue Modeling and Numerical Simulation with (Mpdx) and Taylor Series Distributed Parameters Processes", GALAXIA GUTENBERG Publishing house, Cluj-Napoca, 2009, 361 pages.
- [4] Abrudean, M., "Systems theory and automatic regulation", MEDIAMIRA Publishing house, Cluj-Napoca, 1998, 170 pages.
- [5] User Guide, Matlab 7.5.0 (R2007b).
- [6] Different technical papers of Tenaris Silcotub Zalău.

Design and implementation of a reliable gas leak detection system for domestic environments

Pal-Stefan Murvay and Ioan Silea

Abstract—Gas leaks represent the cause of many accidents both in the industry sector and domestic environments. Due to the deployment of complex supervisory control and data acquisition (SCADA) systems, accident rates have decreased in the industry sector but they are still high in domestic environments. Here, we propose a gas leak detection system based on redundant sensing units and investigate ways of integrating it in the distributor's SCADA system. We also look at some off-the-shelf gas leak detection devices for home usage and present results of the study made on them. The performance of each studied device is analyzed in comparison to our solution. Results of this analysis show that the proposed implementation is suitable for a more reliable detection of gas leaks.

I. INTRODUCTION AND MOTIVATION

The worldwide natural gas infrastructure is constantly expanding and the number of customers connected to the distribution network is continuously increasing. This emphasizes the importance of assuring the network reliability. Leaks represent the main problem related to gas pipelines and the main cause of incidents. The most damaging leaks are the ones small in size that occur below detection levels [7] leading to gas buildup which can cause explosions. The majority of accidents caused by gas installations come from domestic environments [2] denoting the need for a reliable and widely used solution that focuses on this category of customers.

Many gas leak detection methods have been studied and proposed including: acoustic monitoring, optical monitoring, gas sampling, soil monitoring, flow monitoring and model-based methods. Reviews of the most common leak detection methods presenting specific advantages and disadvantages of each method can be found in [10] and [9]. The main disadvantages of these techniques are the high cost and the increased probability of them generating false alarms. Due to these disadvantages and the fact that the majority of methods were designed for industrial applications, only some of them are suitable for domestic gas leak detection. Many gas leak detectors for home usage are available on the market offering different features ranging from simple alarms to devices that switch off the gas supply when a leak is detected. We studied some of these off-the-shelf devices and present our survey later in the paper.

The main focus of the work done in detecting gas leaks was on providing reliable monitoring systems for the infras-

tructure of the gas provider. Natural gas providers employ complex SCADA systems to monitor the state of the production, transport and distribution network without including the consumer's installation beyond the connection point.

Supervisory Control And Data Acquisition (SCADA) systems are used in order to provide real-time control and monitoring functions and are being widely deployed in critical infrastructures becoming a crucial element for the safe operation of these systems. SCADA can be used for relatively simple systems like an environmental conditions monitoring system for an apartment building, or more complex systems such as one monitoring all the activity in a thermal power plant. A basic SCADA system uses several Remote Terminal Units (RTUs) and/or Programmable Logic Controllers (PLCs) which gather data from sensing units and can perform local control actions. The collected data is sent to a SCADA central host computer where it is analyzed and control decisions are made. Operator terminals usually called Human Machine Interfaces (HMIs) are also used to provide human input to the system.

The SCADA system of the gas provider extends up to the regional distribution points or at most up to the meters installed at the entrance on the consumer's property. With the increased interest for replacing conventional meters with smart meters due to their multiple benefits [4] the supplier's SCADA system could extend to include these smart gas meters. Smart meters are not intended only for remote meter reading but also as a mean of providing a wide range of useful functionalities. Therefore, a gas leak detection system could be directly integrated in the meter. Smart meters equipped with such leak detection components (mainly based on flow and pressure monitoring) are already used as legislation requires it in some countries [6]. Both domestic customers and distribution companies could benefit from this extension of the monitored area. On one side the customer would benefit from the ability to prevent any damaging accident by using the gas monitoring system. On the other side the distributor would benefit from having the possibility to generate detailed usage statistics and, by protecting against gas leak incidents, will assure the safety of its infrastructure and equipment. As recent work shows [3] there is indeed interest in providing accurate gas usage statistics but to the best of our knowledge a proposal for integrating the home gas monitoring system in the SCADA distribution infrastructure hasn't been made.

This paper describes the design and study of a method for providing reliable gas leak detection system and presents possibilities for integrating it into the distributor's SCADA

Pal-Stefan Murvay and Ioan Silea are with the Department of Automatics and Applied Informatics, Politehnica University of Timisoara, Romania Email: stefan.murvay@gmail.com, ioan.silea@aut.upt.ro

infrastructure. In what follows we will first discuss in section II important issues that influence the design of a reliable gas leak detector. In section III we present our proposal for gas leak detection system and in section IV we evaluate some off-the-shelf solutions. The results of the performance tests are shown in section V followed by the conclusion of our work in section VI.

II. DESIGN ISSUES FOR RELIABLE GAS LEAK DETECTION SYSTEMS

When building a gas leak detecting system there are several aspects to consider in order to assure reliability some of which are also considered in [13]: availability, leak sensitivity, false alarm rate, optimum positioning, maintenance autonomy and alarming method.

The *availability* (A) property of a gas leak detection system can be defined as the degree to which it is in a functional state:

$$A = \frac{Uptime}{Uptime + Downtime}$$

The gas detector has to be able to perform its function in any possible situation. One common problem of safety critical process monitoring devices is the loss of power supply. If powered from the electrical grid the availability of a detector is directly proportional with the electricity availability. This dependence can be reduced or completely removed with the introduction of a battery-based back-up power module and careful electronic design to reduce power consumption. Reducing the dependence on other systems results in an increased availability.

The *leak sensitivity* represents the minimum level of the gas concentration that can be detected. This sensitivity affects how early can a leak be detected. Taking corrective actions on time depend on a good leak sensitivity. Gas detectors have their sensitivity usually specified for different types of substances.

One property that has great impact on the level of confidence in a gas leak detector is the *false alarm rate* (FAR). To define the FAR we must first consider that the process of detecting a gas leak can have four possible results:

- *True positive (TP)* - the presence of a leak is correctly detected;
- *False positive (FP)* - the alarm is sounded when a leak is not present;
- *True negative (TN)* - the absence of a leak is correctly detected;
- *False negative (FN)* - the presence of a leak is not detected;

According to [5] the false alarm rate is defined as:

$$FAR = \frac{FP}{FP + TN}$$

False alarms can be caused by a number of factors like:

- exposure to substances (other than the ones in the scope of the detector) for which the device has an increased sensitivity;

- changes in the environmental conditions (temperature, humidity, etc.) that may affect sensor readings;
- calibration problems or faulty sensing elements.

For a reduced FAR, changing environmental conditions must have a small (preferably none) influence on the employed sensing elements which should respond only to changes in concentration of the target substance. Therefore sensing elements should be chosen with respect to this remark. The FAR can be further reduced by introducing redundant leak detection elements based on different sensing techniques.

Finding the *optimum location* for positioning the sensing elements is a challenge for an accurate detection of the leak location. Generally accepted rules for the installation of gas detectors state that such devices should be positioned at a distance of 0.3 – 1.0 meters from the floor or ceiling. The former in the case of liquid propane gas (LPG) and the latter in the case of liquefied natural gas (LNG). The detector should also be no farther than 1.5 meters from the gas source. If more than one sensing element is used for detecting gas leaks, by positioning each of them in different locations the location of the leak can be found with greater accuracy.

We define the *maintenance autonomy* of a gas leak detector as the degree of maintenance required by the device to function properly. If we refer to maintenance in the general sense there are two main types of maintenance. There is on one hand a preventive maintenance which involves inspecting, detecting and correcting small malfunctions before they occur or before they can lead to major failures. On the other hand, corrective maintenance acts upon the of a malfunction which in the case of gas leak detection could be too late. Therefore, we refer here only to preventive maintenance. One possible reason for the need of maintenance is the recalibration that may be required by some devices. This is a mandatory action for sensing units which are known to drift away from their original calibration. Sensing elements operating in normal environments can become clogged by accumulating dust. The use of filters can increase the maintenance autonomy but eventually they will have to be changed.

A gas leak detection process triggers an alarm in case of a correct detection. Ultimately, the whole system is as efficient as it's *alarming method*. There are several methods proposed for "sounding" an alarm which can be divided in two categories: local and remote. Local alarming methods such as audio and visual signals have a limited action range. If some corrective action is needed from behalf of a person, this person has to be in the close proximity of the alarming unit. A remote alarming method signals the detected gas leak to a remote device which alerts a person that is not near the gas detector.

All solutions that could solve the previous points are limited by the *cost* factor. Some solutions may provide perfect reliability but at an unaffordable cost. Therefore choosing the right solution is a matter of making the best trade-offs between reliability and cost.

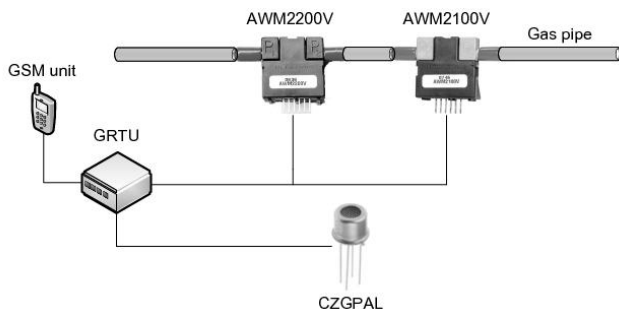


Fig. 1. Local gas monitoring system

III. DESIGN OF THE LOCAL GAS MONITORING SYSTEM

As being part of a SCADA system the local gas monitoring system includes several basic elements:

- sensing units (SUs) capable of acquiring data, communicating to other devices and performing local processing and control
- a RTU which we refer to as a Gas Remote Terminal Unit (GRTU)
- a communication unit for sending remote information
- control units or actuators for taking corrective measures

In designing the local gas monitoring system we addressed the factors discussed in the previous section focusing on those features that improve reliability. Our purpose at this point was not to provide a complete solution but to present design guidelines and to evaluate the performance of the main system components. Figure 1 presents the implemented monitoring system design.

A. Sensing units

For an improved leak sensitivity and a reduced FAR we opted for using redundant SUs. By using this approach, problems caused by a faulty SU can be avoided. A false alarm given by a faulty SU can be detected by comparison with the readings from the other SUs. This decrease the probability of the device producing a false alarm. As a proof of concept we used three SUs each of which utilizes a different gas sensor: one universal gas sensor (CZGPAL), one gas flow sensor (AWM2100V) and one gas pressure sensor (AWM2200V). An even greater number of SUs would decrease the probability of a false alarm but at the expense of an increased cost.

The CZGPAL is a universal gas sensor operating at 5V which indicates the change in gas concentration as a variation of the circuit resistance. These types of sensors need to operate at high temperatures and thus they are equipped with a heater element which for the CZGPAL sensor heats up to 450°C. At this temperature a reaction of the gas with the oxygen chemically absorbed at the sensor's surface changes the conductivity of the sensing element as a result of differences in the amount of electrons at the surface [1]. Gas sensors based on this functioning principle are the most commonly used in gas detection applications as we found for the studied off-the-shelf solutions.

The AWM2100V and AWM2200V from Honeywell are two microbridge sensors from the AWM2000 series each comprised of two Wheatstone bridges. The first one is a mass air flow sensor and the second is a low differential pressure sensor. These sensors were designed for detection of gas flows and pressures smaller than the ones present in real gas installations. So it is clear that some adaption have to be done when installing them in a real system for reducing the pressure/flow to fit the sensor limits. Another approach would be to use sensors with larger measuring ranges. We chose these two sensors to fit our smaller scale test setup. Two additional circuits are needed for the operation of the sensors: one heater control circuit and a supply circuit for the sensing bridge.

As gas leaks are most likely to appear in several common places like: fittings, connections, valves and gas appliances [8], we aim to position the SUs close to these potential leak points. In particular pressure and/or flow sensors can be used directly at the appliance outlet. For example in an oven, when the gas is switched off the pressure and flow at the burner should be equal to 0 (from the sensor point of view). If the switch is faulty gas will come out of the burner resulting in a change of pressure and flow. The leak detection will of course be activated only when the corresponding knob is set to the off position.

SUs can be connected to the GRTU trough a wireless link like ZigBee or trough a wired bus. While a wireless link would be comfortable and easy to set up it is disadvantageous in what regards the power consumption. This could be a problem in the case of a power outage when the system has to rely on back-up battery power. We chose to use a wired solution to keep power consumption at a minimum. Because usually gas pipes do not extend for long distances in homes (for practical and aesthetic reasons) all sensors could be placed relatively close to each other and the GRTU without causing wiring problems.

B. Gas RTU

A 16-bit Freescale S12 microcontroller designed for industrial applications was used to implement the GRTU. The MC9S12DJ256 microcontroller from the S12 family has 256Kbytes of flash memory and 12Kbytes of RAM and can operate at a frequency of 25 MHz. These features makes the microcontroller suitable for implementing more complex functions like the ones of a smart meter.

The GRTU receives data from all SUs and processes it in order to decide if a gas leak is present or not. This decision is taken depending on the gas concentration in the air and data on gas flow and pressure. Leaked gas is in danger of igniting only when it's concentration in the air is situated in a certain interval bounded by two values called lower explosive limit (LEL) and upper explosive limit (UEL). A concentration lower than LEL will not ignite because the it is too small. Concentrations higher than UEL will also not ignite because there would be too little oxygen. The LEL for LNG is approximately 5% and 2.5% for LPG. The UEL of LNG

is around 14% while the UEL for LPG is approximately 9.5%.

Our implementation supports two alarming levels. A pre-alarm is triggered when the gas concentration reaches a warning level followed by the alarm trigger if the concentration reaches the alarm level. The prealarm has the purpose of drawing attention to the possible existence of a leak while the alarm is triggered only when the leak exceeds the second threshold.

C. Communication unit

Upon detection of a gas leak two actions have to be taken: an alarm has to be triggered and, if possible, corrective measures have to be applied.

As stated in the previous section a local alarm can be emitting a specific sound and/or visual signals. As this type of alarms are ubiquitous in current solutions we will not insist on these methods. We implemented a remote alarming method using a GSM unit to transmit alarms trough SMS messages to a list of mobile devices. We used an old Nokia 6210 mobile phone as the GSM unit which we connected to the GRTU trough a serial RS232 bus. The phone was controlled by sending basic AT commands.

D. Control units

In what concerns the corrective measures to be taken one possible course of action, present in solutions on the market, is to turn off the gas supply using an electrovalve. This ensures that no further gas will be leaked but does not reduce the concentration of gas in the room. To reduce the hazardous gas concentration the proper ventilation of the room is mandatory. This can be done by an automatic fan ventilator like the one presented in [12] as part of an air quality monitoring system. In kitchens a hood, if present, can be used for ventilation instead of installing an additional device only for this purpose.

E. Integrating in the SCADA system

A general schematic of a SCADA network is depicted in Figure 2. Although it may seem to be straightforward the integration of local gas monitoring systems in the SCADA network requires careful planing in what concerns the support used for transmitting data and the protocols that are to be used. In this respect we investigate available solutions to find the ones that are suitable.

Whether we consider the gas monitoring system as part of a smart meter or as a standalone entity the communication technologies employed for the integration of smart meters in SCADA networks can be also used in our case. Different mediums have been used to transmit data from individual meters to centralized data bases [11]: internet, radio waves, telephone (both mobile and fixed line), etc. One way communication involving these technologies were used for automated meter reading (AMR) to reduce the workload of data collectors or to completely remove the need for a data collecting person in the field (in the case of fixed network AMR). Adding additional features to AMR networks results

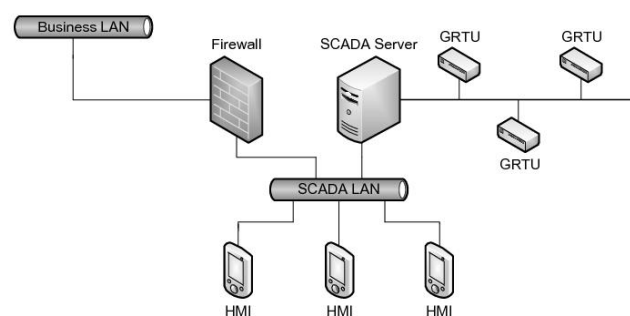


Fig. 2. Gas monitoring SCADA network

in automatic meter management (AMM) systems which are based on two way communication. This allows not only reading meter data but also controlling it providing a suitable solution for a gas monitoring system.

IV. OFF-THE-SHELF GAS LEAK DETECTORS

In order to have an understanding of the available products and some points of comparison for our solution, we studied several off-the-shelf gas leak detection solutions: one integrated in a home automation surveillance system, one detector connected to an electrovalve and one simple alarm device. In what follows we shortly present the features of each device as they are specified in their documentation as well as features that we found by dissecting the devices.

A. GA-2S wireless gas leakage detector for LS-30

The GA-2S detector can spot Liquefied Natural Gas (LNG) and Liquefied Petroleum Gas (LPG) leaks with a sensitivity of 8000ppm for LNG and 3300ppm for LPG. It has a maximum power consumption of 2W. This module is equipped with a PIC12F629 8-bit microcontroller and a Nemoto NAP-55A gas detecting sensor which needs approximately 60 seconds to heat up before it can be properly used. The detector module communicates with the LS-30 Security and Home Automation System trough RF signals. When the LS-30 receives a detection warning signal from the GA-2S it can (depending on how it is programmed) sound an alarm and make phone calls or even send messages trough ethernet or GSM if the user has some additional modules connected.

B. SicurGAS detector with electrovalve

The second device we evaluated is a gas detector with an electrovalve that can also detect both LNG and LPG leaks. It has a maximum consumption of 3W and uses a MQ-2 gas sensor. This sensor has a heating time of around 180 seconds. When the gas concentration level reaches 10% of LEL, the device starts the alarm and comands the electrovalve to close the gas supply. After this the valve can only be reopened manually.

C. LX-GS gas alarm

We also studied a simple gas alarm device which has the only purpose of signaling an increased concentration of gas by auditory and visual signals. The device's specification

TABLE I
SENSITIVITY TO DIFFERENT SUBSTANCES

Subst.	Detecting concentration (ppm)		
	NAP-55A(GA-2S)	MQ-2(SicurGAS & LX-GS)	CZGPAL
LPG	1250-11000	200-5000	-
LNG	2500-25000	5000-20000	800-20000
H_2	1500-12500	300-5000	100-5000
Alcohol	2000-17500	100-2000	-

TABLE II
POWER CONSUMPTION OF GAS DETECTORS DURING START-UP AND NORMAL EXECUTION

Value significance	Power consumption (W)			
	GA-2S	SicurGAS	LX-GS	Our solution
Expected	≤ 2	< 3	< 4	≤ 2
During startup	4.006	2.314	3.583	1.744
Average	3.933	2.296	4.740	1.712

states that it can detect LPG and LNG having a power consumption of up to 4W. After power-on the device needs approximately 4 minutes to warm-up the MQ-2 sensor before entering the waiting state. The alarm level is reached when the concentration rises up to a value between 3% LEL and 20% LEL. Several alarming levels are signaled by a one digit seven segment LED display and different sound patterns. Working environment -10° and 40°C , relative humidity $< 97\%$ RH. This device is controlled by MDT2051, an 8-bit microcontroller running at frequencies lower than 20MHz.

V. PERFORMANCE ANALYSIS

For evaluating the proposed solution we made an analysis of our implementation and of the previously described off-the-shelf devices. First we looked at factors that have a negative effect on the FAR, like sensitivity to other substances. Table I shows the sensitivity of each device to different substances including some that should not be considered by the sensor like alcohol and H_2 . In particular our tests revealed that all sensors used in the studied devices are sensible to alcohol as all detectors triggered alarms when alcohol vapors were generated in their proximity. This is not a problem in a system that used several different sensing units like our implementation but increases the FAR for detectors with only one sensor.

Next we measured the power consumption of the devices during start-up and average value during normal functioning. The startup time over which we made measurements is the one given for each detector in previous sections. The average value was obtained from one second samples during a period of two hours for each device. Table II presents the measured values and the values stated by the specification of each instrument. For our proposed solution the expected consumption represents the theoretical value calculated according to the absolute maximum ratings given in the specification of utilized components. In order to have a correct comparison, only the power consumption of each detector was measured, excluding the home automation station for GA-2S, the electrovalve of SicurGAS and the GSM unit of our proposed solution. The first thing we observed was that the consump-

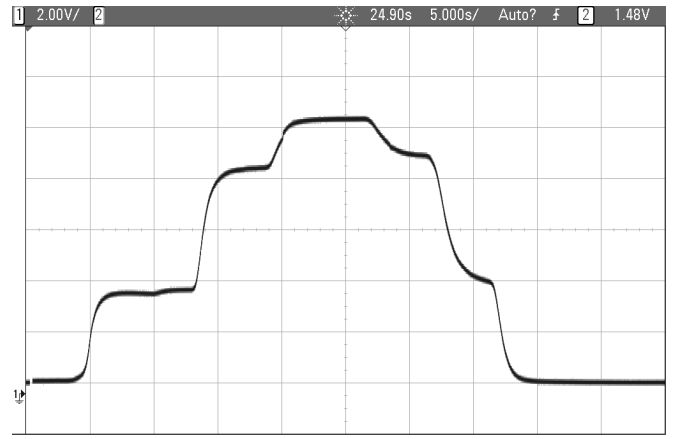


Fig. 3. Response of the AWM2200V sensor to different pressure levels

tion specified in the manual is not the real consumption of the device in all cases. Only the measurements made for the SicurGas detector were in accordance with the specification values. Although the detectors are not battery powered a user might want to connect them to an UPS in which case the real power consumption has to be known in order to choose the right battery size. The measurements also show that SicurGAS has the lowest consumption from the three off-the-shelf devices. Our implementation proved to have a slightly lower consumption but it is very likely that this will increase when using pressure and flow sensors with larger sensing range for a real-life setting.

Tests were also made for comparing the responses given by the gas flow and pressure sensors for variations of the pipe pressure. Our test environment was composed of a compressor generating a constant air flow in a plastic pipe on which the sensors were installed. Figures 3 and 4 shows the response of the flow and pressure sensors to small pressure variations (in the order of tenths of millibars). We introduced these variations by adjusting the compressors's pressure regulator. The oscilloscope plots show that the sensitivity and response time of the two sensors make them suitable for detecting gas leaks since in domestic environments pipeline pressure is in the order of tens of millibars.

VI. CONCLUSION AND FUTURE WORK

In this paper we propose a gas leak detection system based on redundant sensing elements. The evaluation of our system in comparison with several off-the-shelf solutions shows that it has a lower power consumption and that such a system has some advantages when trying to distinguish between true and false alarms. The combination of several detection techniques also has the advantage of providing the means to generate detailed statistics regarding gas usage and pipeline state. This would be useful for smart gas meters or for integrating the monitoring system in the distributor's SCADA network.

As tests made in laboratory conditions on smaller scale models are not sufficient, further tests have to be made in a real scale system and over longer periods of time to determine the suitability of the proposed system for

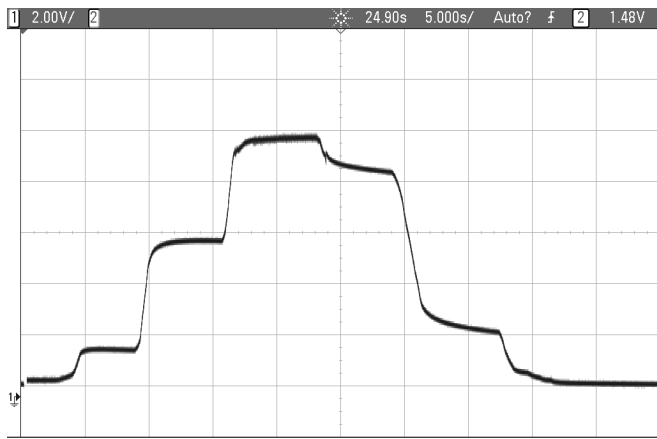


Fig. 4. Response of the AWM2100V sensor to different pressure levels

wide scale deployment. Future work also includes analyzing security aspects of such solutions in order to make them tamper proof.

ACKNOWLEDGMENT

This work was partially supported by the strategic grant POSDRU 107/1.5/S/77265, inside POSDRU Romania 2007-2013 co-financed by the European Social Fund - Investing in People and by CNCSIS-UEFISCDI project number PNII-IDEI 940/2008.

REFERENCES

- [1] N. Barsan and U. Weimar. Understanding the fundamental principles of metal oxide based gas sensors; the example of CO sensing with SnO₂ sensors in the presence of humidity. *Journal of Physics: Condensed Matter*, 15(20):R813R839, 2003.
- [2] P. Burgherr and S. Hirschberg. Comparative assessment of natural gas accident risks. PSI Report 05-01, Paul Scherrer Institut, Jan. 2005.
- [3] G. Cohn, S. Gupta, J. Froehlich, E. Larson, and S. Patel. Gassense: Appliance-level, single-point sensing of gas activity in the home. In P. Floren, A. Krger, and M. Spasojevic, editors, *Pervasive Computing*, volume 6030 of *Lecture Notes in Computer Science*, pages 265–282. Springer Berlin / Heidelberg, 2010.
- [4] S. Darby. Smart metering: what potential for householder engagement? *Building Research Information*, 38(5):442–457, 2010.
- [5] T. Fawcett. ROC graphs: Notes and practical considerations for researchers. *Machine Learning*, 31(HPL-2003-4):1–38, 2004.
- [6] S. Kono, N. Nagai, K. Yuasa, T. Fujimoto, M. Suzuki, Y. Fujii, and T. Hiroyama. Development of intelligent domestic ultrasonic gas meters. *Proceedings of the 23rd World Gas Conference*, 2006.
- [7] W. K. Muhlbaumer. *Pipeline Risk Management Manual*. Elsevier, Burlington, MA, 2004.
- [8] L. P. Pham and D. Pryor. Key considerations when designing a gas detection system ... *Chemical Engineering*, July 1998.
- [9] S. L. Scott and M. A. Barrufet. Worldwide assessment of industry leak detection capabilities for single & multiphase pipelines. Project Report for the Minerals Management Service 8/03A120, Offshore Technology Research Center, Aug. 2003.
- [10] Y. Sivathanu. Natural gas leak detection in pipelines. Technology status report, En'Urga Inc, West Lafayette, IN, 2003.
- [11] J. Vasconcelos. Survey of regulatory and technological developments concerning smart metering in the european union electricity market. RSCAS Policy Paper, Robert Schuman Centre for Advanced Studies, Jan. 2008.
- [12] N. Yamazoe. Toward innovations of gas sensor technology. *Sensors and Actuators B*, 108(1-2):2–14, 2005.
- [13] J. Zhang. Designing a cost effective and reliable pipeline leak detection system. *Pipes and Pipelines International*, 42(1):20–26, 1997.

Investigating F# as a development tool for distributed multi-agent systems

Alex Muscar and Costin Bădică

Abstract— In this paper we explore the use of F#, a functional language, as a means of developing distributed multi-agent systems. In order to achieve this goal, we started with the implementation of the MapReduce algorithm. We also implemented the algorithm in JADE, hence allowing for an initial comparison of the language-platform combination from an usability point of view with respect to the development of distributed agent systems. The results include an analysis of their underlying threading models as well as of the overall system performance. We show how the algorithmic specification can be easily mapped onto the constructs offered by the language. In the final discussion we identify some interesting features of F# that make it suitable for implementing distributed multi-agent systems.

I. INTRODUCTION

Recently we have set up a research project to investigate the development of software systems based on the metaphor of distributed multi-agent organizations, starting from modeling up to design and implementation. This problem has applications in various fields ranging from crisis and disaster management [1], [2] to network traffic monitoring [3]. While there are many methodologies and modeling approaches for multi-agent systems, some of them also emphasizing the organizational aspect, only relatively few address system development from a multi-agent organization perspective, and even less recognize the importance of the design and programming of distributed algorithms for the proper investigation and understanding of the computational aspects of multi-agent organizations. In particular we think that our approach could later on allow us to rigorously address more interesting computational properties of agent organizations, like self-configuration and adaptivity. Therefore, within this context we have setup an initial task of evaluating available programming frameworks with support for software agents, concurrency and distribution.

In what follows we focus on our initial task concerning the design and implementation of distributed algorithms using two programming frameworks that we consider appropriate: JADE and F#¹, a modern, statically typed functional language. For

experimental evaluation we have considered the distributed version of the Depth-First Traversal – DDFT hereafter – algorithm introduced in [4].

Firstly, we start by abstracting a multi-agent system as a distributed computing environment composed of many computational entities that communicate and coordinate by exchanging messages over a suitable communication infrastructure (for example a computer network) [4]. Second, we adopt a very weak notion of “agency” by understanding a (“software”, rather than “intelligent”) agent as a computational entity that (i) has its own thread of control and can decide autonomously if and when to perform a given action; (ii) communicates with other agents by asynchronous message passing. We consider asynchronous programming as being characterized by many simultaneously pending reactions to internal or external events [5]. Note that this weak concept of agent shares similar properties with the actor concept that was introduced in the early ’80s by the community of object-oriented concurrent programming [6].

It is also interesting to note that this view of software agents is compatible with the definition of *computational entities* introduced by [4]. This allows for the definition of a natural mapping between a distributed algorithm and an underlying programming and/or middleware platform – F# and JADE in this particular case – that can be easily applied to simple algorithms, like DDFT, as well as can be further generalized to more complex distributed algorithms specific to multi-agent organizations [7].

Other functional languages that take a similar approach to software agents are Erlang [8], [9], which is well known for writing distributed programs, and more recently Scala [10] (note that both languages use the term “actor” rather than “agent”). The first reason we preferred F# over Erlang or Scala is that F# seamlessly integrates with the Common Language Runtime so it has a far richer runtime system than Erlang. Since Scala targets the JVM it’s on par with F# with respect to the runtime system, but in the recent years the development process of the JVM has been slow, whereas .NET and Mono² have continued to develop at a steady pace.

In addition to the vast runtime, the language features built in support for asynchronous programming [5] and a rich and expressive type system [11]. Both features will prove useful in implementing distributed algorithms. Amongst the language’s declared strengths are parallel I/O programming, parallel CPU programming and algorithmic development. Thanks to the fact that it also runs on Mono, F# is a cross-platform solution.

This work was partially supported by the strategic grant POS-DRU/CPP107/DM11.5/S/78421, Project ID 78421 (2010), co-financed by the European Social Fund Investing in People, within the Sectoral Operational Programme Human Resources Development 2007 - 2013.

A. Muscar is with Software Engineering Department, University of Craiova, Bvd. Decebal 107, Craiova, 200440, Romania amuscar@software.ucv.ro

C. Bădică is with Software Engineering Department, University of Craiova, Bvd. Decebal 107, Craiova, 200440, Romania cbadica@software.ucv.ro

¹The F# 2.0 Language Specification – <http://research.microsoft.com/en-us/um/cambridge/projects/fsharp/manual/spec.html>

²The Mono Project – <http://www.mono-project.com>

Our choice of JADE³ was motivated by the fact that JADE is one of the most popular agent frameworks. Another important advantage is that JADE employs the weak notion of agent with respect to the semantics that we adopted, thus making agents suitable as generic entities in a distributed system. The fact that JADE is Java-based and Java is a well known imperative language, makes it a good baseline for our F# solution.

The rest of the paper is structured as follows. In section II we outline the DDFT algorithm that we considered in our experimental implementations using both F# in section III and JADE in section IV. We proceed with an experimental evaluation and comparison of those two implementations in section V. Finally we draw some conclusions in section VI.

II. BACKGROUND AND PROBLEM FORMULATION

We have implemented the DDFT algorithm introduced in [4] using F# and JADE with the goal of assessing the two platforms. The choice of the DDFT algorithm is motivated as follows. Firstly, visiting each agent from a set of agents is a basic operation in a multi-agent distributed environment. For example, this operation is a step of more complicated algorithms for self-organizing networks [12]. Secondly, this algorithm is simple enough to allow a straight forward implementation, while being sufficiently complex to use a non-trivial state transition system for representing an agent's life cycle.

The algorithm operates over a graph in which vertices are entities and edges are communication links between entities. Each entity has a set of neighbors which does not necessarily coincide with the set of all vertices. In the context of multi-agent systems this can be seen as a simplified form of organization. The key point of DDFT is to visit the graph by forwarding a *token* for as long as possible and *backtracking* when forwarding is not possible. In this scenario there is exactly one *initiator*. The other entities will be *idle* waiting for the token. When visited, an entity forwards the message to each of its neighbors in turn and when it runs out of neighbors it sends the token back to the sender. If the entity has already been visited it will just reply signaling that the link is a *back-edge* and it will not proceed to forwarding the token.

The pseudocode can be found in the algorithm listing 1.

```

procedure INITIATOR // Spontaneously
  Unvisited ← N(x)
  initiator ← false
  Visit
end procedure

procedure IDLE
  receiving T do
    entry ← sender
    Unvisited ← N(x) \ {sender}
    initiator ← false
    Visit
  end receiving
end procedure

procedure VISITED

```

```

receiving T do
  Unvisited ← Unvisited \ {sender}
  send(Backedge) to sender
end receiving

receiving Return ∨ Backedge do
  Visit
end receiving
end procedure

```

```

procedure VISIT
  if Unvisited ≠ ∅ then
    next ← Unvisited
    send(T) to next
    become Visited
  else
    if ¬initiator then
      send(Return) to entry
    end if
    become Done
  end if
end procedure

```

Listing 1. Distributed Depth-First Traversal

We can identify four states of the DDFT algorithm: *Initiator*, *Idle*, *Visited* and *Done*. As stated above only one entity will start in the *Initiator* state, whereas the rest will have the state *Idle*. When it is visited, an entity enters the state *Visited*. After the algorithm has ran, all the entities will be in the *Done* state. Also, in keeping with the terminology in [4] we will make use of three types of messages: (i) *T* representing the token that is forwarded, (ii) *Backedge* to signal that a certain link is a back-edge and (iii) *Return* to return the token to the agent it received it from.

III. THE F# APPROACH

F#'s rich type system provides for both expressive and safe programming. More information can be encoded at the type level and it can be statically checked by the compiler. As an example, the three message types we defined earlier can succinctly be expressed using a *union type*:

```

type Message =
  | T of MailboxProcessor<Message>
  | Backedge
  | Return

```

Listing 2. The F# Message type

The code shown in listing 2 defines a new type that has only three *constructors*. Before discussing how this type can help us write safer code, let us first focus on the *T* type constructor. We declared type *T* to take a parameter of type *MailboxProcessor* < *Message* >. Here *MailboxProcessor* is the type of the mailboxes that F#'s agents use. Note that a mailbox type is parameterized by the defined type *Message*. This has the effect of constraining the type of messages that can be sent and received only to those messages we are interested in. This approach is similar to that used in Erlang where agents who need to get a response to a message also pack their identifiers within the message and to that of JADE since it's FIPA⁴ compliant.

Note that the definition of a fixed number of constructors helps us write safer code. This is achieved by using the *match*

³Java Agent Development Framework – <http://jade.tilab.com/>

⁴The Foundation for Physical Intelligent Agents – <http://www.fipa.org/>

construct on type constructors. This technique is called *pattern matching* and it can be used to decompose complex structures. As we have an exhaustive list of constructors, the compiler can check if we handled all of them and eventually issue a warning if one was omitted.

As an example of pattern matching in action let's take the *Idle* function. In our example we are only interested in handling the *T* message since we are in the *Idle* state, so we try to match the message we received against a *T* constructor (the first case in the *match* construct). If the message we received is indeed of type *T* the *sender* variable will be bound to the value carried by the message (the sender of the message in our case). After the arrow we have the body of the case which is executed if the match is successful. We handle a *T* message by calling the helper function *Visit* that forwards the message to one of the agent's neighbors. Otherwise, when a *Backedge* or *Return* were received (which do not trigger valid state transitions), we just ignore them by using the wildcard `_` that matches anything and generates a recursive call to *Idle*.

```
let rec Idle () = async {
  let! msg = mailbox.Receive()
  match msg with
  | T sender ->
    return! Visit neighbours sender
  | _ -> return! Idle () }
and Visited unvisited sender = async {
  let! msg = mailbox.Receive()
  match msg with
  | T sender ->
    sender.Post Backedge
    return! Visited unvisited sender
  | _ -> return! Visit unvisited sender }
and Visit unvisited entry = async {
  let unvisited = remove entry unvisited
  match unvisited with
  | n::ns ->
    n.Post (T mailbox)
    return! Visited ns entry
  | [] ->
    if not initiator then entry.Post Return
    return () }

if initiator then Visit neighbours mailbox
else Idle ()
```

Listing 3. The F# version of the DFT algorithm

Programs that implement message passing software agents in F# follow a certain pattern: sets of recursive functions, each defining an asynchronous computation, are used to model the states of a state machine out of which only one is identified as the initial state [11]. This idiom can be used to write any agent who can be modeled as a state machine.

The part of the implementation that models the state machine is shown in listing 3. It follows the idiom described above and it maps pretty closely to the pseudocode version of the algorithm. The functions implementing the possible states of the agent and the auxiliary function *Visit* are defined such that they are easily readable, similarly to the pseudocode notation. Note that we have omitted to define an explicit function for the *Initiator* state and instead we used the *Visit* function that is invoked in the last *if* statement of the listing.

The *Visit* function is the most interesting part of the code since this is where we visit a neighbor and remove it from the unvisited neighbors list. We do this in 2 steps. First we call *remove* to remove the sender of the previous message from our unvisited neighbors list. This is a non-destructive operation; instead of changing the list itself it returns a new one that doesn't contain the sender. The second step is to decompose the *unvisited* list argument using pattern matching. Again, this is a non-destructive operation, i.e. we don't alter the underlying structure of the list. We simply take the first element (the *head* of the list) and bind it to the variable *n* as well as the rest of the list (the *tail*) and bind it to the *ns* variable. If the list is empty (this situation is handled by the second case of the *match* construct) it means we ran out of neighbors to visit so we return the token to the sender and use *return* (note the lack of an exclamation mark) to terminate the agent. This corresponds to a transition to the *Done* state. Note that the state transitions are implicitly coded by letting the functions that implement each state to call each other.

A. Asynchrony

F# has built in support for asynchronous programming via blocks of the form *async {...}* which in F# parlance are called *asynchronous computation* [5]. Inside *async* blocks the some control constructs (e.g. *let!*, *return!*) of the language take a special meaning which helps in composing asynchronous computations in a manner that's close to the core language. When executed, an asynchronous computation will eventually produce a value.

As an example let us consider the function *Idle* from listing 3 that describes the computation associated to state *Idle*. Binding the result of an asynchronous computation can be achieved by using *let!*. In our code *let!* executes an asynchronous receive operation on the mailbox, *mailbox.Receive()*, and binds its result to *msg*. Thus our code doesn't block while waiting for a message. As for *return!*, it can be understood as making a tail call to another asynchronous function – note that because the body of *Visit* is all inside of an *async* block we call it an asynchronous function. This leads to efficient recursive patterns (wasting neither heap nor stack space) which are natural ways of expressing looping behavior in functional languages. In the context of the DDFT algorithm, the *return!* construct is used to implement the state update after a state transition.

IV. THE JADE APPROACH

JADE is a very flexible agent platform that provides the basic functionalities for the development of distributed multi-agent systems. A JADE-based system is composed of a set of software agents. Each software agent has its own thread of control. The actual program performed by a JADE agent is defined as a set of plugin components known as *behaviors*. A behavior consists of a sequence of primitive actions. Behaviors are executed in parallel using interleaving of actions on the agent's thread with the help of a non-preemptive scheduler, internal to the agent [13].

Our JADE implementation of the DDFT algorithm is fairly straight forward. We have extended the generic *Behaviour* class to represent our agent's behavior. State transitions are implemented by updating the value of the *state* instance variable in a *switch* statement that checks the current value of the *state* instance variable. Similarly to the F# implementation, we have implemented the *Visit* function separately.

The relevant part of the JADE implementation can be found in listing 4.

```

public void action() {
    MessageTemplate tokenMessage =
        MessageTemplate.MatchContent("T");
    ACLMessage msg;
    switch (state) {
        case INITIATOR:
            visit();
            break;
        case IDLE:
            msg = receive(tokenMessage);
            if (msg != null) {
                entry = msg.getSender();
                unvisited.remove(entry.getName());
                visit();
            } else
                block();
            break;
        case VISITED:
            msg = receive();
            if (msg != null) {
                if (msg.getContent() == "T") {
                    String sender =
                        msg.getSender().getName();
                    unvisited.remove(sender);
                    ACLMessage reply =
                        msg.createReply();
                    reply.setContent("Backedge");
                    send(reply);
                } else
                    visit();
            } else
                block();
            break;
    }
}

public boolean done() {
    return state == DONE;
}

private void visit() {
    if (unvisited.size() > 0) {
        String next = unvisited.remove(0);
        ACLMessage msg =
            new ACLMessage(ACLMessage.INFORM);
        AID aid =
            new AID(next, AID.ISLOCALNAME);
        msg.addReceiver(aid);
        msg.setContent("T");
        send(msg);
        state = VISITED;
    } else {
        if (!initiator) {
            ACLMessage msg =
                new ACLMessage(ACLMessage.INFORM);
            msg.addReceiver(entry);
            msg.setContent("Return");
            send(msg);
        }
        state = DONE;
    }
}

```

Listing 4. The JADE version of the DFT algorithm

Whenever an agent must receive a message, we use the following JADE idiom: (i) invoke the *receive()* method, (ii) test if there is a message in the mailbox, (iii) do the work if an appropriate message was received, otherwise invoke *block()*. The invocation of *block()* is necessary because the *receive()* method is non-blocking, i.e. it immediately returns either a message or *null* if the mailbox is empty.

In the *visit()* method we use the destructive method *Remove()* that operates on the unvisited neighbors list. This method updates the list by physically deleting the entry and returning it. The other destructive operations are: (i) the assignments to the *state* instance variable, and (ii) the assignment to the *entry* instance variable that happens while the agent is in *Idle* state.

V. RESULTS AND DISCUSSION

So far we have seen two approaches for implementing the same algorithm: (i) one based on a functional programming language and (ii) the other using the more familiar imperative paradigm combined with a middleware framework. We are now going to take a closer look at the implications of our choice in each case.

A. Safety

In F# we use statically typed messages and we constrain our mailboxes to operate only on those message types that are relevant to our agent. By doing so we ensure that we won't send messages that we can't handle or that our agent can't understand. This method is also applicable when using serialization since the .NET runtime embeds type information in the serialized entities so they can safely be re-constructed at the destination.

In JADE there are three possibilities: (i) plain strings, the least safe method since strings cannot be checked to be of an appropriate type; (ii) serialization which is a convenient way of packing messages that are sent between agents written in Java, and (iii) ontologies which set up a vocabulary and a nomenclature of elements that can be used as message contents.

B. State and mutability

In sections III and IV we have talked about destructive and non-destructive operations. The F# code doesn't change any data structure, it only creates new ones by filtering old ones (i.e. the unvisited neighbor list). Also the state transition is not modeled by changing a variable, but rather as calls to methods.

In the Java version on the other hand we modify the unvisited neighbor list every time we call *visit()*. We also change the value of the *state* instance variable which is used to model state transitions in a couple of places.

We make this distinction since paraphrasing Donald E. Knuth "shared state is the root of all evil". Even if this is not an issue in our simple example, having global shared state can be a hindrance for concurrency. JADE tries to alleviate this

by using one thread per agent (which we'll see in section V-D causes other kinds of problems). One could easily imagine a scenario where the programmer spawns her own threads that interact with JADE's agents thus circumventing JADE's attempt at safety if no proper locking mechanism is used. Nevertheless some times using shared state is the natural way of modeling the problem. The same kind of issues can manifest in solutions implemented in F#, it's just that thanks to the language's functional nature and the abstractions it offers such issues are easier to avoid.

C. Usability

The structure of the programs derived using both approaches follows the algorithm specification closely, but the F# solution (including the code to generate a tree shaped hierarchy of agents) is nearly half the size of the JADE solution. The agent logic itself spans over 30 lines of F# code whereas in JADE it spans over 70 lines of code. This is due to the brevity and expressivity of F# thanks to which the programmer is spared of writing too much boilerplate (code that follows a certain pattern and could easily be abstracted away) code. Note that this is the case in which we used strings for messages in JADE. If we were to take the solution using serialization or the one using ontologies the amount of code would grow further.

Another advantage we get from using F# is that we don't have to explicitly manage the state of the agent ourselves. Instead of keeping track of it in a variable, like we do in Java, we let the runtime take care of that. We just call functions corresponding to states whenever we want to transition to a new state. This makes the code more declarative, allowing the programmer to state *what* we want done instead of *how* we want to do it.

D. The threading model

While the code structure of the solutions is quite similar, following closely the algorithm, the underlying systems behave quite differently. Whereas JADE uses one thread per agent [13], F#'s runtime uses the thread pool approach [5] that it is known to have as consequence the improvement of system's scalability.

Threads are expensive because they consume significant system resources. So the number of threads that can be spawn by an application should not be too large. Thus the number of agents that can be created by an application using JADE is limited by the number of threads that the guest operating system supports. Since this number is usually not very large, this proves to be a problem for any system when a fine-grained solution is needed.

F#'s approach of using lightweight tasks which are scheduled for execution in a thread pool leads to a more scalable solution. In this scenario a (rather small) number of OS threads run tasks that are scheduled in the thread pool. Thus a very large number of tasks can be ran without paying the price of spawning a thread per task. Depending on the implementation, new threads can be spawned if necessary up to a maximum number which is implementation dependent.

In the thread pool case a task may be executed by multiple threads. This is known as *thread-hopping*, a task can be executed by any number of threads, depending on the availability of such threads in the thread pool. F# asynchronous computations can be suspended when making an asynchronous call and they are resumed when the result of the call is available. At this point the computation is re-scheduled and it continues where it left off. Even though the code is written in a sequential fashion its execution model is not the same as of regular programs. This is not an issue for local state since state is not kept in the thread's stack but in heap memory.

There are some assumptions that the F# model makes. The most important one is that asynchronous operations don't block the executing thread except when performing useful computations [5]. This is a key point because a thread can only execute one task at a time and it's the task who decides when to release the thread. As an example say we have a web server written using asynchronous computations. If the socket I/O operations were blocking then the program would end up consuming a thread per client, thus making the use of asynchronous computations useless.

JADE does offer a *block()* method that marks a behavior as being blocked, but it has a different purpose. While blocked a behavior does not poll for messages but the agent continues taking up a thread. When a message is delivered all blocked behaviors will be marked as active again.

E. Performance

While DDFT is a good example for getting a feel of developing distributed systems using both solutions and the means they offer for this task, in order to evaluate the performance of both platforms we implemented an example inspired from the real world: Google's *mapreduce* [14]. In *mapreduce* a machine—the *master*—splits the work in chunks and assigns each chunk to a *mapper*—more than one chunk can be assigned to a mapper if there are more chunks than mappers. When done, the intermediary results generated by mappers are forwarded to one or more *reducers* for final processing.

Using the *mapreduce* model we implemented an application that counts the number of occurrences of each word in a collection of documents—4 documents, 11.8 Mb each. In our setup we had a master, a reducer and between 1 and 4 mappers, each running on its own machine. In order to simplify the implementation each mapper had the 4 documents available locally. In both the F# and JADE implementations the following steps are performed:

- 1) The master waits for m mappers and r reducers—in our case m varies between 1 and 4 and r is always 1;
- 2) When a new client connects to the master it will be considered a mapper if the master does not have m mappers yet, otherwise it will be classified as a reducer;
- 3) When all the mappers and the reducers are connected the master assigns one or more documents to be processed to each mapper in a *round-robin* fashion;

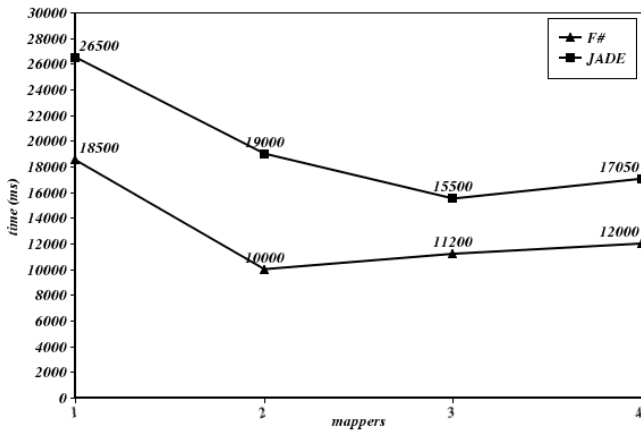


Fig. 1. mapreduce performance for F# and JADE.

- 4) Each mapper counts the number of appearances of each word in the document(s) it has been assigned and for each tuple $\langle \text{word}, \text{count} \rangle$ a message is passed back to the master;
- 5) When all the mappers are done, for each tuple $\langle \text{word}, \text{list}(\text{counts}) \rangle$, the master sends a message to the reducer;
- 6) For each tuple received in the previous step the reducer adds the counts and sends back a message $\langle \text{word}, \text{sum}(\text{counts}) \rangle$;
- 7) When the reducer is done the master shows the final counts.

We ran the test on network of computers with Core 2 Duo processors running at 2.2 GHz each with 4 GBs of RAM. The computers are connected by a Myrinet-2000 network⁵. The F# benchmark was run on .NET 4.0 and the Java one on Java 1.6.0.22. Figure 1 shows a comparison of the total time for each solution with a varying number of mappers—from 1 to 4.

VI. CONCLUSIONS AND FUTURE WORK

In light of our experiment we think F# is a viable option for writing distributed algorithms. Its succinctness and expressivity make the implementation stay close to the algorithmic form. Our F# implementation of the DDFT algorithm is half the size of the JADE version.

F#'s asynchronous computation model that makes use of a thread pool proved to be a scalable option. This is confirmed by the results shown in [5] as well as by our results from section V-E where the F# solution performed consistently better than the JADE implementation. Also the fact that F# doesn't use one thread per agent means that a larger number can be spawned thus allowing for finer grained solutions to be implemented.

Based on our initial results we would like to use F# to investigate computational properties of distributed multi-agent organizations, with a focus on advanced properties like self-configuration and adaptivity. One line of research is to see

if F#'s advantages outlined in this paper will scale-up to significantly more complex systems. One interesting technique that we would like to investigate is the use of the type system to encode information such as capabilities which are used in modeling adaptive organizations [15]. This could be achieved by using encoding techniques at the type level like *phantom types*⁶.

The *language oriented* paradigm supported by F#, which allows for easier development of embedded domain specific languages, is very interesting. Two particular features seem promising for our research: code quotations and computation expressions [11]. Code quotations allow to generate and work with F# code programmatically much like Lisp's meta-programming abilities. This might be useful in a code migration scenario where code meant to migrate is somehow marked and when necessary its representation can be obtained via a quotation and migrated in a serialized form. In an adaptive organization this could prove useful.

REFERENCES

- [1] M. Scafes and C. Badica, "Preliminary design of an agent-based system for human collaboration in chemical incidents response," in *Proceedings of the 7th International Workshop on Modelling, Simulation, Verification and Validation of Enterprise Information Systems, MSVVEIS-2009*. INSTICC PRESS, 2009, pp. 53–63.
- [2] F. Leon, M. H. Zaharia, and G. M. Atanasiu, "Adaptive multiagent system for seismic emergency management," *Management & Marketing*, vol. 3, no. 4, pp. 73–80, 2008.
- [3] M. Rehak, M. Pechoucek, M. Grill, J. Stiborek, K. Bartos, and P. Celeda, "Adaptive multiagent system for network traffic monitoring," *IEEE Intelligent Systems*, vol. 24, pp. 16–25, 2009.
- [4] N. Santoro, *Design and analysis of distributed algorithms*. Wiley, 2007.
- [5] D. Syme, T. Petricek, and D. Lomov, "The F# asynchronous programming model," 2011.
- [6] G. Agha, *Actors: A Model Of Concurrent Computation In Distributed Systems*. The MIT Press, 1986.
- [7] E. A. Sultanik, R. N. Lass, and W. C. Regli, "Dynamic configuration of agent organizations," in *Proceedings of the 21st international joint conference on Artificial intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2009, pp. 305–311. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1661445.1661494>
- [8] J. Armstrong, *Programming Erlang: Software for a Concurrent World*. Pragmatic Bookshelf, July 2007. [Online]. Available: <http://www.worldcat.org/isbn/193435600X>
- [9] —, "Making reliable distributed systems in the presence of software errors," Ph.D. dissertation, Royal Institute of Technology Sweden, 2003.
- [10] P. Haller and M. Odersky, "Actors that unify threads and events," in *Proceedings of the 9th international conference on Coordination models and languages*, ser. COORDINATION'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 171–190.
- [11] D. Syme, A. Granicz, and A. Cisternino, *Expert F# 2.0*. Apress Berkley, 2010.
- [12] A. L. Lopes and L. M. Botelho, "Improving multi-agent based resource coordination in peer-to-peer networks," *Journal of Networks*, vol. 3, no. 2, February 2008.
- [13] F. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE*. Wiley, 2007.
- [14] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," *Commun. ACM*, vol. 51, pp. 107–113, January 2008.
- [15] S. A. DeLoach, W. H. Oyen, and E. T. Matson, "A capabilities-based model for adaptive organizations," *Journal of Autonomous Agents and Multiagent Systems*, vol. 16, no. 1, pp. 13–56, February 2008.

⁵<http://www.myri.com/myrinet/overview/>

⁶Phantom types – http://www.haskell.org/haskellwiki/Phantom_type

Modeling and simulation of the heating elements into an electrical oven

Andreea Maria Neacă, Mitică Iustinian Neacă

Abstract— In this paper we present the mathematical model of the thermal transfer within an electrical oven, first without anything inside and after that, with some parts to be heated on order of thermal treatment. We also present the simulation subsystem for the resistive elements and the simulation scheme of the entire oven.

The results of the simulation are very eloquent; they are done for two different powers, High Power and Low Power. Following the results we found that it is a need to control the temperature of the part to be heated and a permanent monitoring of the resistors' temperature.

I. INTRODUCTION

THIS paper presents a part of the results obtained within the mathematical modeling and during the simulation of the operation period of the heating elements of an electrical oven with resistances.

Unlike those presented in work papers [2], [4], in which we were analyzing the preheating regime of the oven (without anything inside), in this article we take into account the fact that after a period of preheating, we introduce the parts into the oven, in order to obtain thermal treatment.

During the simulation we have used the same oven with the construction presented in [2], having a parallelepiped form, with the interior cavity of: $L \times B \times H = 1 \times 0.8 \times 0.7\text{m}$ [4]. As in the study of the preheating regime we take into consideration two equipment cases with resistors which ensure a power of 150kW (HP-High Power) and respectively 10kW (LP- Low Power).

For the heating parts, we have considered the projecting data regarding the maximum load of the oven.

The paper is organized as follows: in Section II are presented mathematical calculations and equations based on which we have made this study and simulation, both for the parts to be thermally treated and for the heating elements. In section III is the simulation and modeling of a heating resistor, and of the entire electrical oven. In section IV are

the results obtained for two different power supplies, High Power and Low Power, and in the same time for two periods of time, long time and short time.

II. MATHEMATICAL MODELING

A. Projecting calculations regarding the parts to be heated

We consider that there will be heated simultaneously 80 stainless steel bars having the dimensions of $30 \times 50 \times 800\text{ mm}$. The bars are placed on special racks, 10 bars one over the other, obtaining so 8 plates with the dimensions of $30 \times 500 \times 800\text{ mm}$. The rack is made of stainless steel bars with the diameter of 10 mm and total length of approximately 23 m. By the construction of the rack we ensure a space of 50 mm between the 8 plates formed of 10 bars located vertically.

It will result so a charge with the following parameters:

- height = 0.5 m
- width $\approx 0.6\text{ m}$
- length = 0.8 m

The mass of a bar will be:

$$m_b = V \cdot \gamma = 50 \cdot 10^{-3} \times 30 \cdot 10^{-3} \times 800 \cdot 10^{-3} \times 7850 = 9.42 \text{ Kg} \quad (1)$$

The mass of the carrying charge (of the bars) is:

$$M_b = 80 \cdot m_b = 80 \times 9.42 = 753.6 \text{ Kg} \quad (2)$$

The mass of the metallic rack will be:

$$m_{\text{rack}} = \frac{\pi d^2}{4} \cdot l \cdot \gamma = 0.01^2 \cdot \pi / 4 \times 23 \times 7850 = 14.2 \text{ Kg} \quad (3)$$

The mass of the whole charge will be:

$$m_c = m_{\text{rack}} + M_b = 14.2 + 753.6 = 767.8 \text{ Kg} \quad (4)$$

By placing the bars on the rack the area through which the heat is transmitted by conduction is increased. So we obtain:

- $S_{\text{prad}} = 1.424 \text{ m}^2$ - the surface through which the heat is transmitted by radiation;

Manuscript received June 18, 2011. This work was partially supported by the strategic grant POSDRU/88/1.5/S/50783, Project ID50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resources Development 2007-2013.

Andreea Maria Neacă is with the Department of Automatics, Faculty of Automatics, Computers and Electronics, Bvd. Decebal, Nr. 107, 200440, Craiova, Dolj, ROMANIA, e-mail: neaca_andreea@yahoo.com.

Mitică Iustinian Neacă, is with the University of Craiova, Faculty of Electrotechnics, Bdul Decebal, nr.107, 200440 Craiova, Romania, e-mail: ineaca@elth.ucv.ro.

- $S_{p_{conv}} = 7.024 m^2$ - the surface through which the heat is transmitted by convection (which includes also the surfaces through which the heat is transmitted by convection and by radiation);

The volume of the parts is:

$$V_p = 80 \times 50 \cdot 10^{-3} \times 30 \cdot 10^{-3} \times 800 \cdot 10^{-3} = 0.096 m^3 \quad (5)$$

The volume of the rack is:

$$V_{rack} = \frac{\pi d^2}{4} \cdot l = 0.002 m^3 \quad (6)$$

$$V_{total} = V_p + V_{rack} = 0.098 m^3 \quad (7)$$

Placing into the oven the charge to be heated, it generates a series of modifications of the equations that govern electrical and thermal processes from within.

B. Mathematical model for the heating resistors

Within the frame of the modeling, there have been considered three different types of resistances placed in the following way:

- first type – on the cover and hearth of the furnace;
- second type – on the lateral walls;
- third type – on the backside and front-side walls (excepting the access door).

So, at the level of the heating resistances, the initial equations (before inserting the parts into the oven) were:

- current drawn from the supply network;

$$i(t) = u(t) / R(T) \text{ and } R(T) = R_0 \cdot C_r(T) \quad (8)$$

- power drawn from the supply network;

$$P_{el} = u(t) \cdot i(t) \quad (9)$$

$$P_{el} = \dot{Q}_{store} + \dot{Q}_{rad} + \dot{Q}_{conv} \quad (10)$$

$$\dot{Q}_{store} = \frac{d}{dt} (Q_{store}) = \frac{d}{dt} (m \cdot c_r(T) \cdot (T_r - T_0)) \quad (11)$$

Where: m - the mass of the resistive conductor;

$c_r(T)$ - the specific heat of the resistor;

T_r - the temperature of the resistor;

$T_0 = 300 K$ - initial temperature;

$$\dot{Q}_{rad} = \varepsilon_r c_n S_r (T_r^4 - T_1^4) = k_1 (T_r^4 - T_1^4) = \dot{Q}_{rad_{r-w}} \quad (12)$$

Where: ε_r - darkness degree of the resistor; $\varepsilon_r = 0.7$;

$c_n = 5.674 \cdot 10^{-8} \text{ W/m}^2 \text{K}^4$ - Stefan-Boltzmann constant;

S_r - the outer surface of the resistive element;

T_1 - the temperature of the inner surface of the oven's wall;

$$\dot{Q}_{conv} = \alpha_r S_r (T_r - T_a) \quad (13)$$

Where: α_r - convection coefficient of the heat transfer between the resistor and the air;

T_a - temperature of the air from inside the oven;

After introducing the parts, (8), (9), (10), (11) and (13) remain the same. As regards the heat exchange by radiation, this phenomenon will not appear only between the resistors and the wall of the oven (as in the case of the oven without charge). A part of the radiations emitted by the heating element will not reach the wall [3]; they will be absorbed by the parts to be heated.

So, (12) becomes:

$$\begin{aligned} \dot{Q}_{rad} &= \dot{Q}_{rad_{r-w}} + \dot{Q}_{rad_{r-c}} = \\ &= \varepsilon_r c_n S_r (1 - k_3) (T_r^4 - T_1^4) + \varepsilon_r c_n S_r k_3 (T_r^4 - T_c^4) \end{aligned} \quad (12')$$

Where: k_3 - the environmental factor targeting between the surface of the heating resistor and the surface of the part to be heated;

T_c - temperature of the surface of the part to be heated;

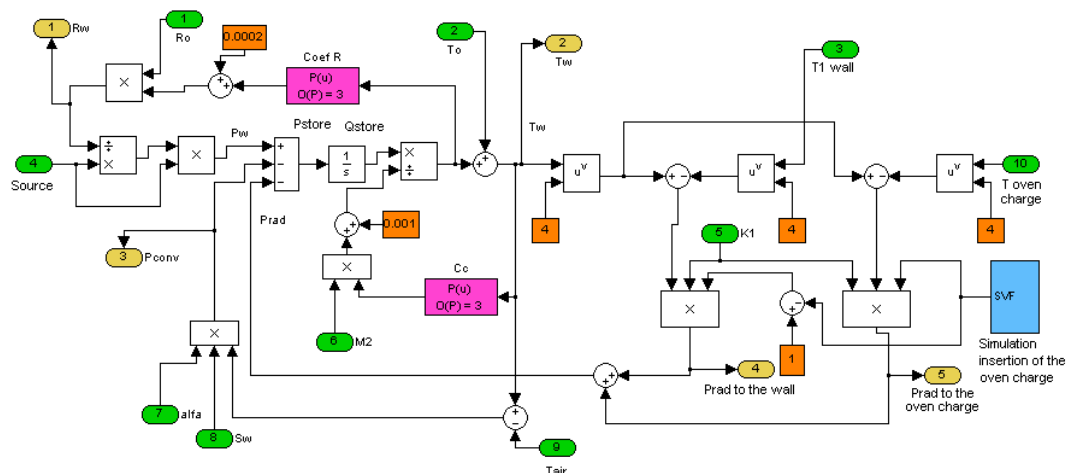
For an oven charge with the dimensions of $0.5 \times 0.6 \times 0.8$ m (for which it had been dimensioned the oven presented in work papers [2], [4]), we estimate an environmental factor targeting $k_3 \approx 0.35$.

Taking into account this modification of (12), in which it appears the term $\dot{Q}_{rad_{r-c}} = \varepsilon_r c_n S_r k_3 (T_r^4 - T_c^4)$ and [5], in the same time, the term $\dot{Q}_{rad_{r-w}}$ modifies its value, the simulation scheme of the heating resistive element has to have a flexible structure. This structure needs to permit the transformation between equations (12) and (12') at a certain moment of time, randomly chosen.

III. SIMULATION AND MODELING OF A HEATING RESISTOR

Based on the previous mathematical equations we have realized the Matlab/ Simulink modeling of the heating element, as shown in figure 1 [1],[6].

It can be observed that this simulation scheme is more complex than the one presented in [2]. Also, the insertion of the parts into the oven, at a random time, brings some modifications in different other structures which are involved in the simulation of the whole oven. It modifies the modeling subsystem of the air from inside the oven [1]. Also, the equations which describe the thermal flux change, they are modifying at the level of the oven's wall.



Additionally, it appears a subsystem which models the parts to be heated. Finally, the simulation scheme of the whole oven is presented in figure 2. (in APPENDIX)

IV. RESULTS AND CONCLUSIONS

A. High Power

In figures 3, 4 and 5 it is presented the way of variation of the temperature of the resistors which equip the 3 zones of the oven, having the power of 150kW (High Power).

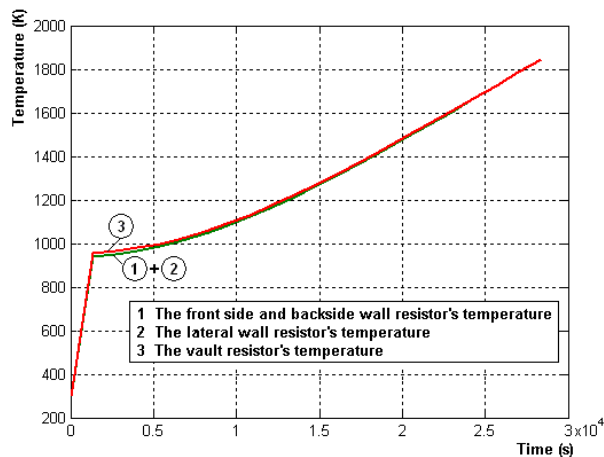


Fig. 3. Temperature of the resistors high power without oven charge

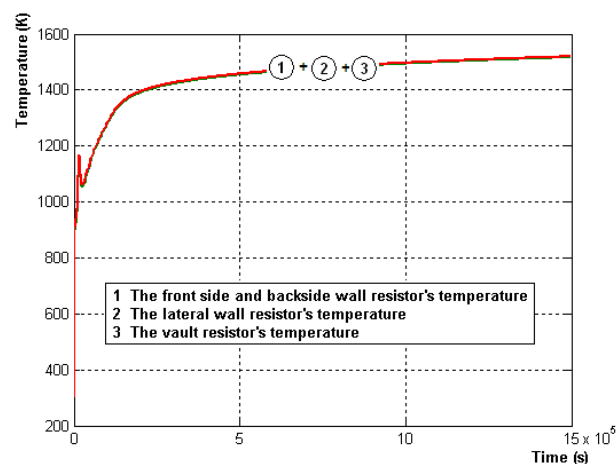


Fig. 4. Temperature of the resistors_high power_with oven charge_long time simulation

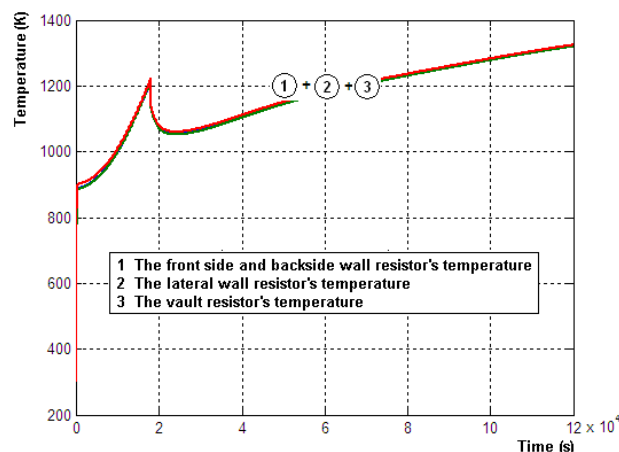


Fig. 5. Temperature of the resistors_high power_with oven charge_short time simulation

Figure 5 is, in fact, a zoom of the figure 4 to show the fact that at insertion time of the parts in the oven (approximately after 5 hours from the beginning of the

preheating period), the temperature of the resistors tends to decrease due to the appearance of a high value thermal flux of radiation between resistors and the part to be heated.

As the parts are heated, the temperatures of the resistors resume their upward course, and as in figure 4, they overpass 1500K.

It is obviously necessary to monitor the temperature of the resistors to not overpass 1250K (work prescribed temperature).

It has been chosen a preheating time of 5 hours to avoid overheating and damage of the resistors, phenomena which appear in case of the oven without charge (Figure 3).

B. Low Power

The second simulation was made for the oven equipped with heating resistors with a power of 10kW (Low Power).

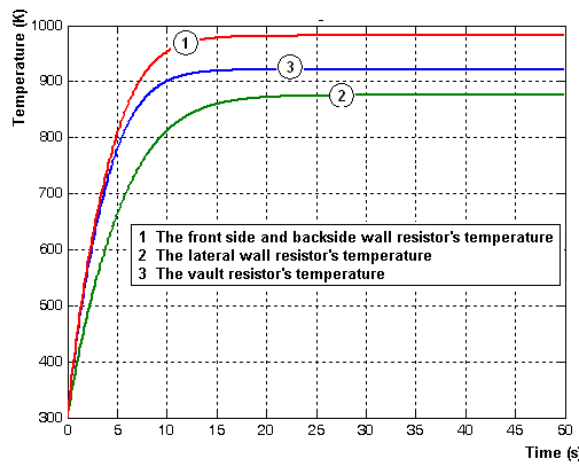


Fig. 6. Temperature of the resistors_low power_without oven charge_short time simulation

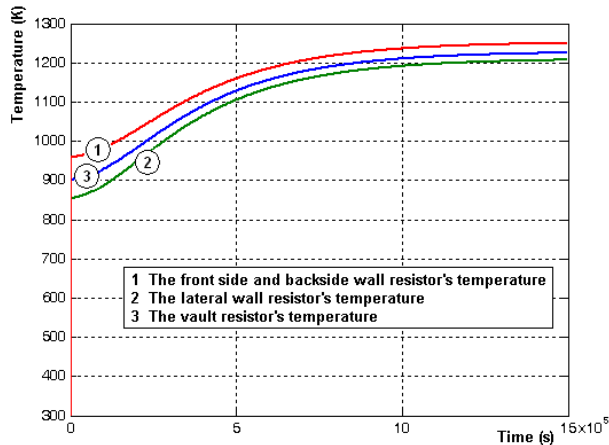


Fig. 7. Temperature of the resistors_low power_without oven charge_long time simulation

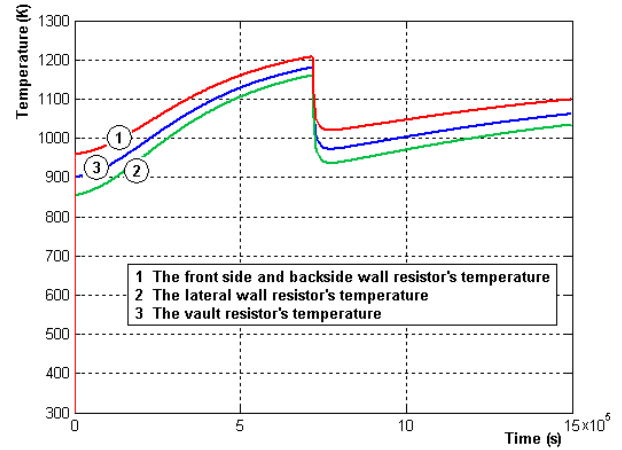


Fig. 8. Temperature of the resistors_low power_with oven charge_long time simulation

In figures 6 and 7 is presented the evolution of the resistors' temperature for the oven without parts inside. We introduce the parts to be heated at $t = 200h$, as shown in figure 8. As for High Power, it appears a sudden drop of the resistors' temperature due to the modification of the radiated thermal fluxes.

In figure 9, 10 and 11 is presented the variation of the temperature of the air from inside the oven and the temperature of the charge to be heated for HP and LP respectively.

Figure 11 shows that the power of 10kW is too small and, the time necessary to heat the charge at approximately 1000K is too long.

From the whole behavior of the oven's simulation, is highlighted the need of a control of the temperature of the part to be heated and a permanent monitory of the resistors' temperature.

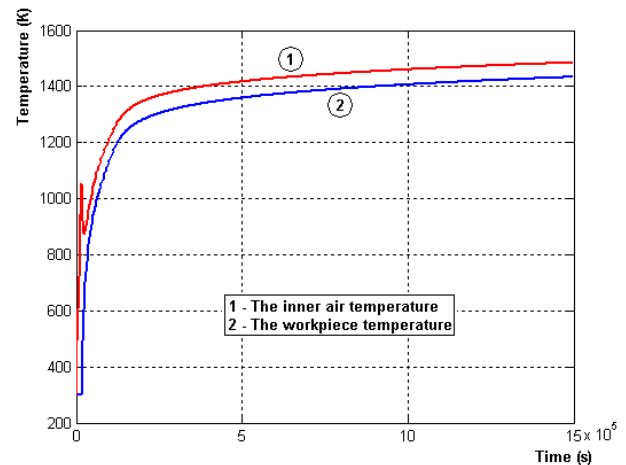


Fig. 9. Temperature of the air inside the oven and the temperature of the charge to be heated_high power_long time simulation

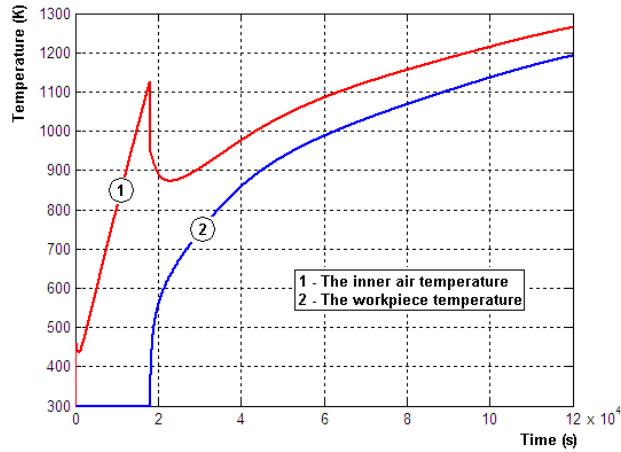


Fig. 10. Temperature of the air inside the oven and the temperature of the charge to be heated_high power_short time simulation

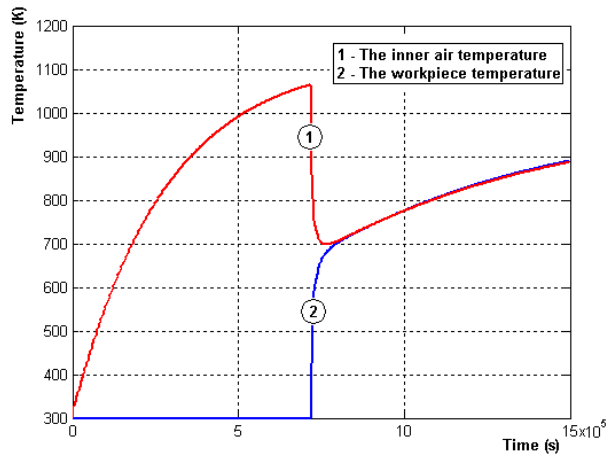


Fig. 11. Temperature of the air inside the oven and the temperature of the charge to be heated_low power_long time simulation

REFERENCES

- [1] B.D.Hahn, D.T.Valentine, *Essential MATLAB for Engineers and Scientists*, ELSEVIER, 2007.
- [2] M.I. Neacă, A.M. Neacă, "The modeling of the heating resistors in transient regime," International Conference on Development and Application Systems DAS 2010, May 27-29, 2010 - Suceava, Romania, ISSN 1844 – 5039, pag.201 – 206.
- [3] M.C.Potter, C.W.Somerton, *Thermodynamics for Engineers*, McGRAW-HILL, 1993.
- [4] A.M. Neacă, M.I. Neacă, *Modeling and simulation of the transient heating regime in an electrical oven*. 2010 IEEE International Conference on Automation, Quality and Testing, Robotics AQTR 2010- THETA 17th edition - May 28-30 2010, Cluj-Napoca, Romania, IEEE Catalog Number: CFP10AQT-CDR, ISBN: 978 – 1 – 4244 – 6722 – 8, pag. 76 - 81.
- [5] F.P.Incropera, D.P.DeWitt, *Fundamentals of Heat and Mass Transfer*, WILEY, ISBN: 978-0-470-05554-0, 2006.
- [6] S.J.Karris, *Introduction to Simulink with Engineering Applications*, Orchard Publications, 2008.

APPENDIX

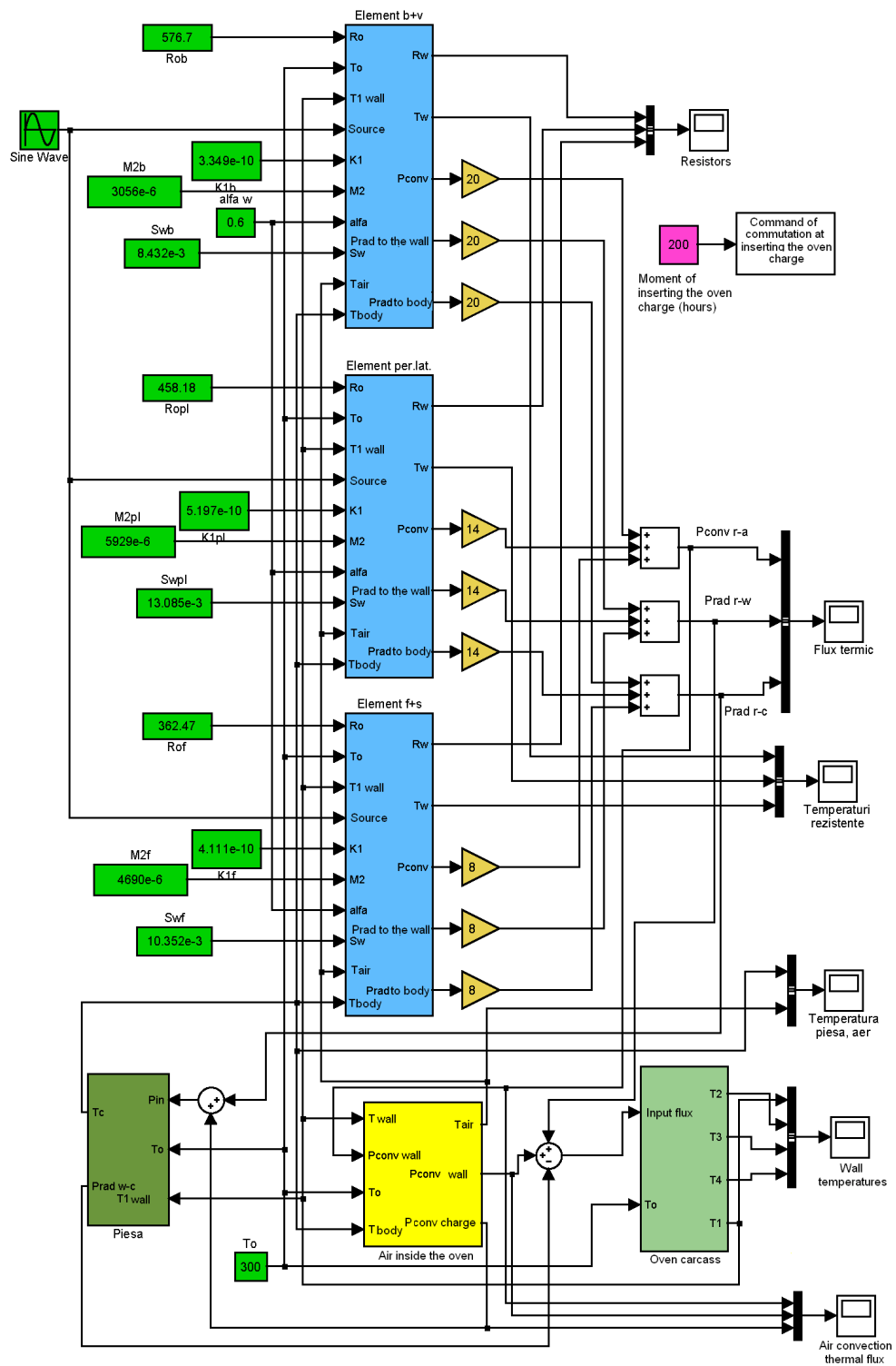


Fig. 2. The oven for thermal treatments

MIMO Conditional Integrator Control for Unmanned Airlaunch

Van Cuong Nguyen and Gilney Damm

Abstract—A Multiple Input Multiple Output (MIMO) controller based on the conditional integrator technique is designed for the robust stabilization of a new satellite launching strategy called (unmanned) airlaunch. This strategy consists in using a two-stages launching system. The first stage is composed of an airplane (manned or unmanned) that carries a rocket launcher which constitute the subsequent stages. The control objective is to stabilize the aircraft in the launch phase. It is developed separately for two nonlinear motion modes of the model, the longitudinal and lateral modes, and is applied to the full multi-input multi-output model of the aircraft. The considered model is highly nonlinear, mostly as a consequence of possible large angle of attack, sideslip and roll angle. Finally, the present work illustrates through simulations the good performance of the proposed control algorithm.

I. INTRODUCTION

This work presents the design of a Multiple Input Multiple Output (MIMO) Conditional Integrator (CI) based controller for the robust stabilization of a new satellite launching strategy called (unmanned) airlaunch. This strategy consists in using a two-stages launching system. The first stage is composed of an airplane that carries a rocket launcher which constitutes the subsequent stage. The aircraft brings the rocket to a desired drop area, consequently avoiding many costs and risks related to land rocket launching. On the other hand, this procedure brings up many other difficulties connected to the instant of releasing the rocket.

Currently, several airlaunch systems are under development (see [14], [4]). Most current airlaunch projects use standard or lightly modified airplanes as first stage. For example, there has been tests using F15, C17, B52, L-1011 in Rascal, QuickReach, Proteus and Pegasus projects. Unlike those, other projects aim to develop an airlaunch system that uses an unmanned aerial vehicle(UAV) instead of a standard aircraft with a human pilot inboard. The objective is then to use an UAV to fly the launcher to the desired drop point. There are many advantages in doing so, in first place safety since no human lives are involved during the delicate launching phase. In addition, since there is no need for life supporting devices, weight is restricted to the strict minimum. Finally, mission may take as long as necessary without human restrictions as tiredness.

The present paper addresses the stabilization of the drop phase. It intends to introduce a robust control scheme for this complex procedure. In fact, airlaunch may be very delicate for many reasons. For example, since the rocket may be

almost as heavy as the UAV, this means that the aircraft will instantaneously lose almost half of its mass. Current airlaunch systems present a much smaller ratio launcher/aircraft and rely on human pilot to stabilize the aircraft during and immediately after the launching instant. The proposed system must replace the human pilot in this stabilization task, with a much more adverse mass ratio. In the same way, the two-stages system is strongly nonlinear and can even for small perturbations be brought quite far from the initial equilibrium point. Furthermore, available models are based on experiments for different flight conditions, with lookup tables and polynomial interpolation between these points. For this reason, parameters and even models are not very well known, and need very robust control schemes like those found in [6], [11] and those used on [2].

These reasons have suggested the possibility of applying a MIMO conditional integrator based sliding mode controller for this problem. Many works have suggested sliding mode for aircraft control when large state excursion is expected (see [20], [15]). More recently, conditional integrator has been applied for the control of part of the dynamics of F-16 aircrafts. The standard procedure found in the literature is from [19].

These results have the drawback of being based on the linearized model of the aircraft. For this reason they may have poor performance for large initial conditions or changes in the operation points. The present work make use of a new control strategy based on the nonlinear model, and that behave better in these extreme situations, that are nevertheless expected in the unmanned airlaunch.

The paper is organized as follows : in section II, we describe the nonlinear mathematical system model based on [13],[21] and [10]. The control design literature is discussed in section III, and its application to the full nonlinear system model in section IV. The paper is completed by some computer simulations and conclusions.

II. SYSTEM MODEL

Drop phase is delicate to model, and requires a large amount of data and previous knowledge about the real system. It can also be represented as a hybrid system, with two discrete models, one before and one after the drop phase (which is considered as instantaneous). In the present work, we have considered the system starting at the instant of the drop, and as consequence the system may be seen as the second discrete model, with initial conditions caused by the drop phase. This set of initial conditions is the result of an imperfect drop that may result in impulses on the forces and moments applied to the UAVs. These impulses may bring it

Van Cuong Nguyen is with IBISC - Université d'Evry Val d'Essonne, Evry, France vancuong.nguyen@ibisc.fr

Gilney Damm is with IBISC - Université d'Evry Val d'Essonne, Evry, France gilney.damm@ibisc.fr

to initial conditions far from the equilibrium. This may result in instability of the aerial vehicle, and to avoid that is the main objective of the controller.

The model used in this case is that of a F-16 fighter with initial forward speed much larger than the necessary equilibrium value, and allowing large excursions of the angular states. In fact, these are the most important disturbances of the drop phase.

Following this procedure, the aircraft in the instant following the dynamic airlaunch is described in the body fixed axes as (see [3], [21], [22]) :

$$\begin{aligned}
\dot{\alpha} &= -\cos \alpha \tan \beta p + q - \sin \alpha \tan \beta r \\
&\quad - \frac{\sin \alpha}{mV \cos \beta} (T + F_x) + \frac{\cos \alpha}{mV \cos \beta} F_z \\
&\quad + \frac{g}{V \cos \beta} [\sin \alpha \cos \theta + \cos \alpha \cos \phi \cos \theta] \\
\dot{\beta} &= \sin \alpha p - \cos \alpha r - \frac{\cos \alpha \sin \beta}{mV} [T + F_x] \\
&\quad + \frac{\cos \beta}{mV} F_y - \frac{\sin \alpha \sin \beta}{mV} F_z + \frac{g}{V} [\cos \alpha \sin \beta \sin \theta \\
&\quad + \cos \beta \cos \theta \sin \phi - \sin \alpha \sin \beta \cos \phi \cos \theta] \\
\dot{V} &= \frac{\cos \alpha \cos \beta}{m} [T + F_x] + \frac{\sin \beta}{m} F_y \\
&\quad + \frac{\sin \alpha \cos \beta}{m} F_z + g [\cos \alpha \cos \beta \sin \theta \\
&\quad + \sin \beta \sin \phi \cos \theta + \sin \alpha \cos \beta \cos \phi \cos \theta] \\
\dot{p} &= \frac{1}{I_{xx} I_{zz} - I_{xz}^2} [(I_{yy} I_{zz} - I_{zz}^2 - I_{xz}^2) r q \\
&\quad - I_{xz} (I_{xx} + I_{zz} - I_{yy}) p q + I_{zz} L - I_{xz} N] \\
\dot{q} &= \frac{1}{I_{yy}} [(I_{zz} - I_{xx}) p r + I_{xz} (p^2 - r^2) + M] \\
\dot{r} &= \frac{1}{I_{xx} I_{zz} - I_{xz}^2} [(-I_{xx} I_{yy} + I_{zz}^2 + I_{xz}^2) p q \\
&\quad + I_{xz} (I_{xx} + I_{zz} - I_{yy}) r q + I_{xx} N - I_{xz} L] \\
\dot{\phi} &= p + \tan \theta (q \sin \phi + r \cos \phi) \\
\dot{\theta} &= q \cos \phi - r \sin \phi \\
\dot{\psi} &= \frac{q \sin \phi + r \cos \phi}{\cos \theta}
\end{aligned} \tag{1}$$

Where $I_{xx}, I_{yy}, I_{zz}, I_{xz}$ are the moments of inertia in kgm^2 , m is the mass of the system (kg) and g the gravity constant. $\alpha, \beta, \phi, \theta, \psi$ in rad , p, q, r in rad/s and V in m/s are the state variables of the airlaunch aircraft model, they represent the angle of attack, sideslip, roll angle, pitch angle, yaw angle, roll rate, pitch rate, yaw rate and airspeed respectively. T is the thrust force, F_x, F_y, F_z and M, L, N are aerodynamic forces and moments respectively. All forces and moments are expressed in N and Nm.

These aerodynamic forces and moments are function of all the considered states. In this model, these aerodynamic forces and moments are under look-up table from wind tunnel data measurements. Finally, the control inputs are respectively aileron (δ_a), rudders (δ_r) and elevator (δ_e) values.

This model is based on wind tunnel data from NASA found in [10], considering the following conditions :

- angle of attack is in the rang of $[-10^\circ, 45^\circ]$ and sideslip of $[-30^\circ, 30^\circ]$
- flag deflection is ignored
- physical constraints for aileron ($|\delta_a| \leq 21.5^\circ$), rudders ($|\delta_r| \leq 25^\circ$) and elevator ($|\delta_e| \leq 30^\circ$)
- all actuators are modeled as a first order model with limit rates $60^\circ/\text{s}$ for aileron and elevator, and $120^\circ/\text{s}$ for rudder.

In particular we use the low quality mode of the F-16 model, and in simulations the aerodynamic data is interpolated and extrapolated linearly from the tables.

III. CONTROL DESIGN

For several reasons, the considered system is hard to control. First because aerodynamic forces and moments are under tabular form. In the second, system presents several hard to deal with characteristics as actuators saturation, plant limitations, large initial conditions (like angle of attack) caused by the launch phase, in a very nonlinear system. Finally, by the nature of the problem, it is required that all states be stabilized, but also that the considered plant (the aerial vehicle) does not collide with the dropped rocket.

However, there exist some approaches for similar classes of systems (see [12] and [7]). In the present work, based on the conditional integrator sliding mode theory, we design a controller addressing the longitudinal and lateral motions separately. This is a standard procedure since the coupling term between both motions is rather small.

Since the objective of this paper is to study the effects of the drop phase, we will assume that all state variables are measurable for feedback control.

A. MIMO conditional integrator control design

The MIMO conditional integrator controller design for the output regulation of a class of minimum-phase nonlinear systems in case of asymptotically constant references is studied in [5], [17] and [18]. These papers concern a servo-compensator performing as a sliding mode controller outside the boundary layer, and performing as a conditional one that provides servo-compensation only inside the boundary layer. First results have studied the SISO case with a scalar sliding surface and the asymptotic stability of the system inside a boundary layer. These results were extended in [16] and [8] for linearized MIMO systems under some additional assumptions. Our present work is dedicated to use a nonlinear extension of these results developed in [1], for stabilizing an unmanned aircraft after the airlaunch phase.

Consider the system :

$$\begin{cases} \dot{e}_1 &= e_2 \\ \dot{e}_2 &= f(e_1, e_2) + g(e_1, e_2)u \end{cases} \tag{2}$$

where $e_1(t) \in \mathbb{R}^n$ is an output error vector, $e_2 = \dot{e}_1$, $u \in \mathbb{R}^n$ control input and $f(e_1, e_2) \in \mathbb{R}^n$, $g(e_1, e_2) \in \mathbb{R}^{n \times n}$ are continuous functions.

Let us define the sliding surface as :

$$s = k_0 \sigma + K_1 e_1 + e_2 \tag{3}$$

where $\sigma \in \mathbb{R}^n$ is the output of the conditional servo-compensator

$$\dot{\sigma} = -k_0 \sigma + \mu \text{sat}(s/\mu) \tag{4}$$

in which μ is the boundary layer, k_0 is a positive parameter, $K_1 \in \mathbb{R}^{n \times n}$ is chosen such a way that $K_1 + sI_n$ is Hurwitz.

The saturation function is determined as :

$$\text{sat}(s/\mu) = \begin{cases} s/\|s\| & \text{if } \|s\| \geq \mu \\ s/\mu & \text{if } \|s\| < \mu \end{cases} \tag{5}$$

We denote $\mathcal{O}_\mu$ as the region neighborhood of $(e_1, e_2) = (0, 0)$ with a radius R_μ for $\|s\| < \mu$

$$\mathcal{O}_\mu = \{e = (e_1, e_2) \in \mathbb{R}^n \times \mathbb{R}^n \mid \|e\| \leq R_\mu\} \tag{6}$$

We state the following assumptions on the forcing terms $f(e_1, e_2)$ and $g(e_1, e_2)$ to design the control algorithm.

Assumption 3.1: $f(e_1, e_2)$ is bounded by a function $\gamma(e_1, e_2)$ of class $\mathcal{K}$ function and a positive constant F_0

$$\|f(e_1, e_2)\| \leq \gamma(e_1, e_2) + F_0 \quad (7)$$

for $(e_1, e_2) \in \mathbb{R}^n \times \mathbb{R}^n$ and while the sliding surface does not enter the boundary layer, i.e. $\|s\| \geq \mu$.

Inside the boundary layer, the function $f(e_1, e_2)$ is required to be Lipschitz for $(e_1, e_2) \in \mathcal{O}_\mu$, as a consequence

$$\|f(e_1, e_2) - f(0, 0)\| \leq L_1\|e_1\| + L_2\|e_2\| \quad (8)$$

where L_1 and $L_2 \in \mathbb{R}^+$.

Assumption 3.2: $g(e_1, e_2)$ satisfies two hypothesis :

Hypothesis 1 : for $(e_1, e_2) \in \mathbb{R}^n \times \mathbb{R}^n$

$$g(e_1, e_2) + g^T(e_1, e_2) \geq 2\lambda I_n \text{ with } \lambda > 0 \quad (9)$$

Hypothesis 2 : for $(e_1, e_2) \in \mathcal{O}_\mu$, function $g(e_1, e_2)$ is Lipschitz and as a consequence

$$\|g(e_1, e_2) - g(0, 0)\| \leq \|g(0, 0)\|v(e_1, e_2) \quad (10)$$

optionally, (10) can be written as

$$\|g(e_1, e_2)g^{-1}(0) - I_n\| \|f(0)\| \leq v(e_1, e_2) \quad (11)$$

in which, $v(e_1, e_2)$ is a suitable function satisfying

$$v(e_1, e_2) = v_1\|e_1\| + v_2\|e_2\| \leq K_v \quad (12)$$

where v_1, v_2 and K_v are suitable positive parameters.

Following these conditions, the controller u defined below in (13) can be applied to (2) to stabilize the system.

$$u = -\Pi(e_1, e_2) \text{sat}(s/\mu) \quad (13)$$

in which,

$$\Pi(\cdot) = \Pi_0 + (\gamma(\cdot) + k_0\mu + F_0)/\lambda \quad (14)$$

Π_0, k_0 are positive, μ is the boundary layer as defined above.

When designing this controller, it is important to pay attention that inside the boundary layer, i.e. $\|s\| \leq \mu$, $\Pi(\cdot)$ must satisfy the condition :

$$\Pi(\cdot) - \Pi(0, 0) \leq \chi(\cdot) = \lambda\chi_1\|e_1\| + \lambda\chi_2\|e_2\| \quad (15)$$

where χ_1, χ_2 are suitable positive constants and λ was defined earlier in (9).

The stability of the control law (13) for system (2) can be demonstrated following the results of [1].

B. Longitudinal control design

In the longitudinal control design, we assume that all lateral state variables are null or constant, only longitudinal states are time varying. Moreover it is assumed that the airspeed's response is much slower than other states, and that the control surface deflection has no effects on the aerodynamic force components (lift and drag) but only on moments.

Aerodynamic forces F_x, F_z and moment M can be calculated by its aerodynamic coefficients (see more in [7]).

$F_x = (C_x(\alpha) + \bar{c}C_{x_q}(\alpha)q/(2V))\bar{q}S$, $F_z = (C_z(\alpha, \beta) + \bar{c}C_{z_q}(\alpha)q/(2V))\bar{q}S$, $M = (C_m(\alpha) + C_{m_q}(\alpha)q\bar{c}/(2V) + C_{m_{\delta_e}}(\alpha)\delta_e)\bar{q}S\bar{c}$. By replacing F_x, F_z , moment M and $\beta = 0, \phi = 0, p = 0, r = 0$ in (1). The model for longitudinal dynamic can be written as :

$$\begin{cases} \dot{\alpha} = \frac{1}{mV} [-\sin \alpha (T + C_x(\alpha)\bar{q}S) + \cos \alpha C_z(\alpha)\bar{q}S] \\ \quad + q + \frac{\rho S}{4m} (-\sin \alpha C_{x_q}(\alpha)\bar{c} + \cos \alpha C_{z_q}(\alpha)\bar{c})q \\ \quad + \frac{g}{V} \cos(\theta_0 - \alpha) \\ \dot{q} = I_7 \bar{q}S (C_m(\alpha)\bar{c} + C_{m_q}(\alpha)\bar{c}q + C_{m_{\delta_e}}(\alpha)\bar{c}\delta_e) \\ \dot{\theta} = q \end{cases} \quad (16)$$

in which S is wing area, $\bar{q}$ air pressure, $\bar{c}$ equivalent width, $I_7 = 1/I_{yy}$, $C_x(\alpha)$, $C_{x_q}(\alpha)$, $C_z(\alpha)$, $C_{z_q}(\alpha)$, $C_m(\alpha)$, $C_{m_q}(\alpha)$, $C_{m_{\delta_e}}(\alpha)$ are aerodynamic coefficients taken from [9].

Equation (16) can be rearranged as :

$$\begin{cases} \dot{\theta} = q \\ \dot{\alpha} = f_{11}^\alpha(\alpha) + (1 + f_{12}^\alpha(\alpha))q + f_{13}^\alpha(\alpha, \theta) \\ \dot{q} = f_{21}^\alpha(\alpha) + f_{22}^\alpha(\alpha)q + g_2^\alpha(\alpha)\delta_e \end{cases} \quad (17)$$

where $f_{11}^\alpha(\alpha)$, $f_{12}^\alpha(\alpha)$, $f_{13}^\alpha(\alpha, \theta)$, $f_{21}^\alpha(\alpha)$, $f_{22}^\alpha(\alpha)$ and $g_2^\alpha(\alpha)$ represent the terms of (16) respectively.

Let us define $x_1^\alpha = \alpha$, $x_2^\alpha = \dot{\alpha} = \dot{x}_1^\alpha$ and $u^\alpha = \delta_e$, which allow us to rewrite (17) into :

$$\dot{\theta} = \eta^\alpha(x_1^\alpha, x_2^\alpha, \theta) \quad (18a)$$

$$\dot{x}_1^\alpha = x_2^\alpha$$

$$\dot{x}_2^\alpha = F^\alpha(x_1^\alpha, x_2^\alpha, \theta) + G^\alpha(x_1^\alpha, x_2^\alpha)u^\alpha \quad (18b)$$

where

$$\begin{cases} \eta^\alpha(\cdot) = (x_2^\alpha - f_{11}^\alpha(x_1^\alpha) - f_{13}^\alpha(x_1^\alpha, \theta))/(1 + f_{12}^\alpha(x_1^\alpha)) \\ F^\alpha(\cdot) = \frac{\partial f_{11}^\alpha(x_1^\alpha)}{\partial x_1^\alpha} x_2^\alpha + \frac{\partial f_{13}^\alpha(x_1^\alpha)}{\partial x_1^\alpha} x_2^\alpha \\ \quad + (1 + f_{12}^\alpha(x_1^\alpha))f_{21}^\alpha(x_1^\alpha) + \left(\frac{\partial(1 + f_{12}^\alpha(x_1^\alpha))}{\partial x_1^\alpha}\right) x_2^\alpha \\ \quad + (1 + f_{12}^\alpha(x_1^\alpha))f_{22}^\alpha(x_1^\alpha) \frac{(x_2^\alpha - f_{11}^\alpha(x_1^\alpha) - f_{13}^\alpha(x_1^\alpha, \theta))}{(1 + f_{12}^\alpha(x_1^\alpha))} \\ G^\alpha(\cdot) = (1 + f_{12}^\alpha(x_1^\alpha))g_2^\alpha(x_1^\alpha) \end{cases} \quad (19)$$

We define now the reference for the angle of attack α_{ref} considered as constant in this study, and the error vector of angle of attack $e_1^\alpha = x_1^\alpha - x_{1ref}^\alpha = \alpha - \alpha_{ref}$ and the variable $e_2^\alpha = \dot{e}_1^\alpha$. Equation (18b) can be transformed into (20) :

$$\begin{cases} \dot{e}_1^\alpha = e_2^\alpha \\ \dot{e}_2^\alpha = F^\alpha(e_1^\alpha, e_2^\alpha, \theta) + G^\alpha(e_1^\alpha, e_2^\alpha)u^\alpha \end{cases} \quad (20)$$

Here we remark that $G^\alpha(x_1^\alpha, x_2^\alpha)$ is invertible, and that $F^\alpha(x_1^\alpha, x_2^\alpha, \theta)$ and $G^\alpha(x_1^\alpha, x_2^\alpha)$ are Lipschitz for the entire domain of actuation of the system.

Applying the control algorithm presented in (13) for system (20) (in this case a nonlinear single input single output system) gives the controller :

$$\begin{cases} u^\alpha = -\Pi^\alpha(e_1^\alpha, e_2^\alpha) \text{sat}(s^\alpha/\mu^\alpha) \\ \Pi^\alpha(\cdot) = \Pi_0^\alpha + (\gamma^\alpha(\cdot) + k_0^\alpha\mu^\alpha + F_0^\alpha)/\lambda^\alpha \end{cases} \quad (21)$$

with

$$\begin{cases} s^\alpha = k_0^\alpha\sigma^\alpha + K_1^\alpha e_1^\alpha + e_2^\alpha \\ \dot{\sigma}^\alpha = -k_0^\alpha\sigma^\alpha + \mu^\alpha \text{sat}(s^\alpha/\mu^\alpha) \end{cases} \quad (22)$$

where $\lambda^\alpha = \min(\|G^\alpha(\cdot)\|)$ for $\alpha \in (-10^\circ, 45^\circ)$. $\Pi_0^\alpha, \mu^\alpha, K_1^\alpha$ and k_0^α are positive constants.

This controller can be shown to assure the stability of angle of attack and its derivative. Furthermore, the errors will be ultimately bounded, where the remaining signals stands for the disturbance on the aircraft speed. Finally, it is interesting to remark that variable θ is left free, in order to allow situations as a looping, where θ is continuously varying. Its derivative on the other hand is bounded, and also goes to a residual set.

Since the airspeed control is only a secondary objective, we design a simple PI controller for the thrust to regulate the airspeed of the system. Its form is :

$$T = -k_P(V - V_{ref}) - k_I(\dot{V} - \dot{V}_{ref})$$

where V_{ref} is airspeed reference, $k_P = 1242$ and $k_I = 955$.

C. Lateral control design

As in the case of longitudinal control design, in the lateral control design it is considered that only lateral state variables are time varying.

Aerodynamic force F_y and moments L , N is calculated by its aerodynamic coefficients (see more in [7]). $F_y = (C_y(\beta) + (C_{y_p}(\alpha)p + C_{y_r}(\alpha)r)\bar{b}/(2V))\bar{q}S$, $L = (C_l(\beta) + C_{l_p}(\alpha,\beta)p\bar{b}/(2V) + C_{l_r}(\alpha,\beta)r\bar{b}/(2V) + C_{l_{\delta_a}}(\alpha)\delta_a + C_{l_{\delta_r}}(\alpha)\delta_r)\bar{q}S\bar{b}$, $N = (C_n(\beta) + C_{n_p}(\alpha,\beta)p\bar{b}/(2V) + C_{n_r}(\alpha,\beta)r\bar{b}/(2V) + C_{n_{\delta_a}}(\alpha)\delta_a + C_{n_{\delta_r}}(\alpha)\delta_r)\bar{q}S\bar{b}$. By replacing F_y , moments L , N and $\alpha = 0$, $\theta = 0$ in (1). The lateral nonlinear dynamic model used for the control design procedure is consequently reduced as below :

$$\begin{cases} \dot{\beta} = \frac{1}{mV}(-\cos(\alpha_0)\sin(\beta)(T + C_x(\alpha_0)\bar{q}S) \\ + \cos(\beta)C_y(\beta)\bar{q}S - \sin(\alpha_0)\sin(\beta)C_z(\alpha_0,\beta)\bar{q}S \\ + \sin(\alpha_0)p - \cos(\alpha_0)r + \frac{\rho S}{4m}(\cos(\beta)C_{y_p}(\alpha_0)\bar{b}p \\ + \cos(\beta)C_{y_r}(\alpha_0)\bar{b}r) + \frac{g}{V}(\cos(\alpha_0)\sin(\beta)\sin(\theta_0) \\ + \cos(\beta)\cos(\theta_0)\sin(\phi) - \sin(\alpha_0)\sin(\beta)\cos(\phi)) \\ \dot{\phi} = p + \cos(\phi)\tan(\theta_0)r \\ \dot{p} = I_3C_l(\alpha_0,\beta)\bar{q}S\bar{b} + I_4C_n(\alpha_0,\beta)\bar{q}S\bar{b} + \frac{\rho V S \bar{b}}{4}[(I_3C_{l_p}(\alpha_0) \\ + I_4C_{n_p}(\alpha_0))p + (I_3C_{l_r}(\alpha_0) + I_4C_{n_r}(\alpha_0))r] \\ + \bar{q}S[(I_3C_{l_{\delta_a}}(\alpha_0) + I_4C_{n_{\delta_a}}(\alpha_0))\delta_a + (I_3C_{l_{\delta_r}}(\alpha_0) \\ + I_4C_{n_{\delta_r}}(\alpha_0))\delta_r] \\ \dot{r} = I_4C_l(\alpha_0,\beta)\bar{q}S\bar{b} + I_9C_n(\alpha_0,\beta)\bar{q}S\bar{b} + \frac{\rho V S \bar{b}}{4}[(I_4C_{l_p}(\alpha_0) \\ + I_9C_{n_p}(\alpha_0))p + (I_4C_{l_r}(\alpha_0) + I_9C_{n_r}(\alpha_0))r] \\ + \bar{q}S[(I_4C_{l_{\delta_a}}(\alpha_0) + I_9C_{n_{\delta_a}}(\alpha_0))\delta_a + (I_4C_{l_{\delta_r}}(\alpha_0) \\ + I_9C_{n_{\delta_r}}(\alpha_0))\delta_r] \end{cases} \quad (23)$$

In which $\bar{b}$ is equivalent length, $I_3 = \frac{I_{zz}}{(I_{xx}I_{zz} - I_{xz}^2)}$, $I_4 = \frac{I_{xz}}{(I_{xx}I_{zz} - I_{xz}^2)}$, $I_9 = \frac{I_{xx}}{(I_{xx}I_{zz} - I_{xz}^2)}$. $C_y(\alpha, \delta_e)$, $C_{y_p}(\alpha_0)$, $C_{y_r}(\alpha_0)$, $C_l(\alpha_0, \beta)$, $C_n(\alpha_0, \beta)$, $C_{l_p}(\alpha_0)$, $C_{n_p}(\alpha_0)$, $C_{l_r}(\alpha_0)$, $C_{n_r}(\alpha_0)$, $C_{l_{\delta_a}}(\alpha_0)$, $C_{n_{\delta_a}}(\alpha_0)$, $C_{l_{\delta_r}}(\alpha_0)$, $C_{n_{\delta_r}}(\alpha_0)$ are lateral aerodynamic coefficients taken from [9].

Equation (23) can be rearranged as :

$$\begin{cases} \begin{bmatrix} \dot{\beta} \\ \dot{\phi} \end{bmatrix} = f_{11}^\beta(\beta, \phi) + f_{12}^\beta(\beta, \phi) \begin{bmatrix} p \\ r \end{bmatrix} \\ \begin{bmatrix} \dot{p} \\ \dot{r} \end{bmatrix} = f_{21}^\beta(\beta, \phi) + f_{22}^\beta(\beta, \phi) \begin{bmatrix} p \\ r \end{bmatrix} + g_2^\beta(\beta, \phi) \begin{bmatrix} \delta_a \\ \delta_r \end{bmatrix} \end{cases} \quad (24)$$

where $f_{11}^\beta(\cdot)$, $f_{12}^\beta(\cdot)$, $f_{13}^\beta(\cdot)$, $f_{21}^\beta(\cdot)$, $f_{22}^\beta(\cdot)$, and $g_2^\beta(\cdot)$ represent the terms of (23) respectively. Equation (24) is mainly used for controller design and stability analysis.

Let us define $x_1^\beta = [\beta, \phi]^T$, $x_2^\beta = \dot{x}_1^\beta = [\dot{\beta}, \dot{\phi}]^T$ and $u^\beta = (\delta_a, \delta_r)^T$, that allow us to rewrite equation (24) to :

$$\begin{cases} \dot{x}_1^\beta &= x_2^\beta \\ \dot{x}_2^\beta &= F^{\beta'}(x_1^\beta, x_2^\beta) + G^{\beta'}(x_1^\beta, x_2^\beta)u^\beta \end{cases} \quad (25)$$

where

$$\begin{cases} F^{\beta'}(\cdot) = (\frac{\partial f_{11}^\beta(\cdot)}{\partial x_1^\beta} + (f_{12}^\beta(\cdot) + f_{12}^\beta(\cdot)f_{22}^\beta(\cdot))(f_{12}^\beta(\cdot))^{-1})x_2^\beta \\ - (f_{12}^\beta(\cdot) + f_{12}^\beta(\cdot)f_{22}^\beta(\cdot))(f_{12}^\beta(\cdot))^{-1}f_{11}^\beta(\cdot) + f_{12}^\beta(\cdot)f_{21}^\beta(\cdot) \\ G^{\beta'}(\cdot) = f_{12}^\beta(\cdot)g_2^\beta(\cdot) \\ f_{12}^\beta(\cdot)[p, r]^T = \frac{\partial(f_{12}^\beta(\cdot)[p, r]^T)}{\partial x_1^\beta} \end{cases} \quad (26)$$

We define an output error vector $e_1^\beta = x_1^\beta - x_{1ref}^\beta$ and $e_2^\beta = \dot{e}_1^\beta$ where $x_{1ref}^\beta = (\beta_{ref}, \phi_{ref})^T$ is the output reference considered as constant. Equation (26) can be transformed into (20) with two new state variables e_1^β and e_2^β .

$$\begin{cases} \dot{e}_1^\beta &= e_2^\beta \\ \dot{e}_2^\beta &= F^\beta(e_1^\beta, e_2^\beta) + G^\beta(e_1^\beta, e_2^\beta)u^\beta \end{cases} \quad (27)$$

$G^\beta(x_1^\beta, x_2^\beta)$ is invertible, and $F^\beta(x_1^\beta, x_2^\beta)$ and $G^\beta(x_1^\beta, x_2^\beta)$ are Lipschitz in the considered domain of $x_1^\beta = [\beta, \phi]^T$, $x_1^\beta = x_2^\beta$ with $\beta \in (-30^\circ, 30^\circ)$ and $\phi \in (-90^\circ, 90^\circ)$.

Application of control law (13) for system (27) leads to the controller :

$$\begin{cases} u^\beta &= -\Pi^\beta(e_1^\beta, e_2^\beta) \text{sat}(s^\beta/\mu^\beta) \\ \Pi^\beta(\cdot) &= \Pi_0^\beta + (\gamma^\beta(\cdot) + k_0^\beta\mu^\beta + F_0^\beta)/\lambda^\beta \end{cases} \quad (28)$$

with

$$\begin{cases} s^\beta &= k_0^\beta\sigma^\beta + K_1^\beta e_1^\beta + e_2^\beta \\ \dot{\sigma}^\beta &= -k_0^\beta\sigma^\beta + \mu^\beta \text{sat}(s^\beta/\mu^\beta) \end{cases} \quad (29)$$

where $\lambda^\beta = \min(\|G^\beta(\cdot)\|)$ for $\beta \in (-30^\circ, 30^\circ)$, $\phi \in (-90^\circ, 90^\circ)$. Π_0^β is a constant large enough, k_0^β is a positive parameter, μ^β is the boundary layer and K_1^β is a positive definite matrix chosen such a way that $K_1 + sI_2$ is Hurwitz.

IV. SIMULATION RESULTS

In section III, the design methodology of the conditional integrator controller to stabilize the angle of attack, sideslip and roll angle is proposed when full knowledge of the aerodynamic characteristics is available. This section presents numerical simulation results for the controller to demonstrate the performance of the proposed conditional integrator control laws in the drop phase.

As mentioned in section II, we assume that the system is in the time instant immediately following the launch. For this reason, its initial conditions are very far from the trimmed operating point of the aircraft that is a complete F-16 nonlinear model in this paper. This model is used since it has already been applied for (manned) air launch, and because its nonlinear model, wind tunnel informations and data are widely known and used for control design. It is important to remark that the model used in the following simulations is even more complete than that used in the control design, for example it includes actuator dynamics and their limitations.

As a consequence, simulations also illustrate some properties of robustness to unmodeled dynamics.

In the following simulations, we have simultaneously applied the SISO longitudinal controller for angle of attack, and the MIMO lateral one for the sideslip and roll angles in the full nonlinear F-16 aircraft model. In order to illustrate the robustness of the CI controller, initial conditions are defined different from trimmed conditions, ranging from small to large values. We may note that the control inputs are limited by their physical limitations told in section II.

The system is studied at the operating point $(V, h) = (154m/s, 3000m)$ corresponding to the trimmed angle of attack $\alpha_0 = 3.6^\circ$, pitch angle $\theta_0 = 3.6^\circ$, sideslip and roll angle $\beta_0 = \phi_0 = 0^\circ$ and to trimmed control surface states : aileron and rudder $\delta_a = \delta_r = 0^\circ$, elevator $\delta_e = -2.3^\circ$.

The control law in (14) whose $\Pi(\cdot)$ can be written more simply as below :

$$\Pi(\cdot) = \Pi_0 + \gamma(\cdot) \quad (30)$$

in which, $\gamma(\cdot) = \gamma_1 \|e_1\| + \gamma_2 \|e_2\|$, γ_1 and γ_2 are positive constant.

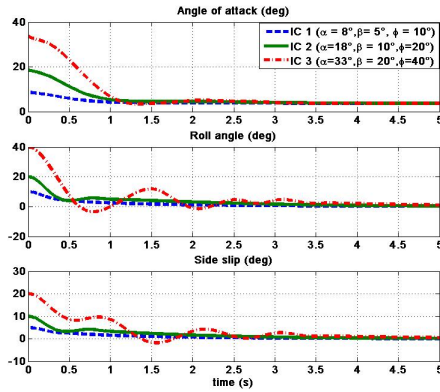


Fig. 1. Output : angle of attack, sideslip Angle and roll angle

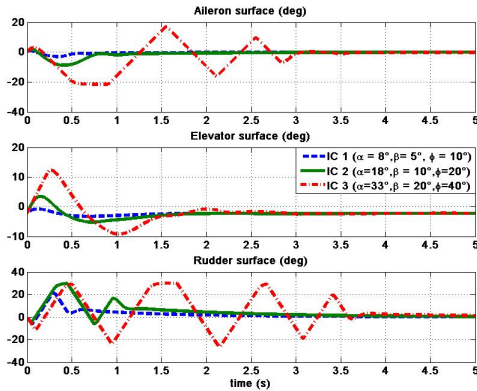


Fig. 2. Control surface : aileron, elevator and Rudder

Application of this control law to two motion modes presented in subsections (III-B) and (III-C) is done by determining the set of parameters Π_0^i , γ_1^i , γ_2^i , μ^i , K_1^i and k_0^i with $i = \alpha, \beta$ corresponding to longitudinal mode and lateral mode respectively. We use the design parameters in the Table I for the longitudinal controller, and the design parameters in Table II for the lateral controller. In Table II, Π_0^β and k_0^β are defined in matrix form because of the difference in

the dynamic property of two state variables *sideslip* and *roll angle*. This definition does not have any consequence on the stability of the system.

Π_0^α	μ^α	γ_1^α and γ_2^α	k_0^α	K_1^α
25.0	1.0	1.1 and 1.1	1.3	1.5

TABLE I
PARAMETERS FOR THE LONGITUDINAL MODE

Π_0^β	μ^β	γ_1^β and γ_2^β	k_0^β	K_1^β
$\begin{bmatrix} 20 & 0.0 \\ 0.0 & 20 \end{bmatrix}$	1.0	0.10 and 0.10	$\begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 2.0 & 0.0 \\ 0.0 & 2.0 \end{bmatrix}$

TABLE II
PARAMETERS FOR THE LATERAL MODE

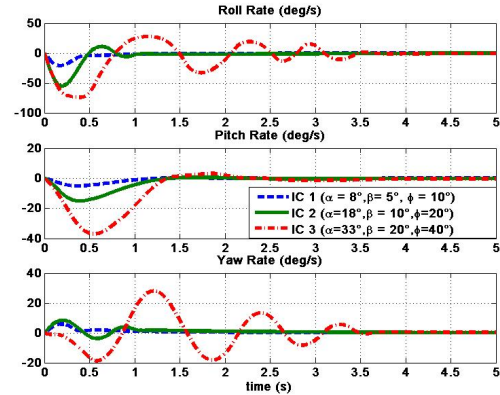


Fig. 3. State variables : Angular rates of system

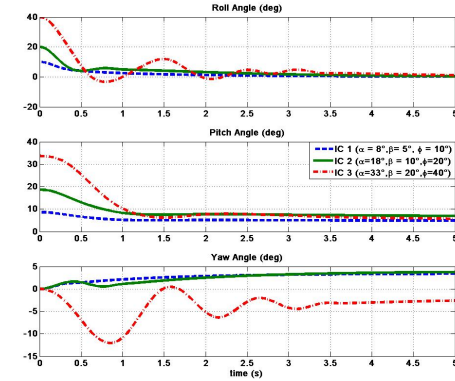


Fig. 4. State variables : Euler's angle of system

The plots in Fig. 1 to Fig. 4 are obtained with the controller structure described in section III used to regulate the angle of attack, sideslip angle and roll angle. The simulation is done with three cases of Initial Conditions (IC) different from trimmed conditions. Dashed plots correspond to the first case where initial conditions are $(\alpha = 8^\circ, \beta = 5^\circ \text{ and } \phi = 10^\circ)$, solid plots are the second case under initial conditions $(\alpha = 18^\circ, \beta = 10^\circ \text{ and } \phi = 20^\circ)$, and dash dotted plots correspond to the third case where initial conditions are $(\alpha = 33^\circ, \beta = 20^\circ \text{ and } \phi = 40^\circ)$.

Fig. 1 represents the convergence of the output variables to the operating point of the aircraft at the end of 5s without static steady error on the output of the system. Figs. 3 and 4 show that angular rates converge to zero, and Euler's angles converge to constant values under all three cases of initial

conditions of the state variables. Fig. 2 demonstrates the control variables limited by their rate limit and their angle limit. The performance requirement for the system is then illustrated by this simulation.

Collision Avoidance

Airlaunch problem does not only require stability of system's states, but also to avoid the possibility of collision between the aircraft and the rocket after the drop phase. Fig. 5 shows the altitude of the aircraft from 0 to 1s in the three previous cases of study and an additional case in which $\alpha = 33^\circ$, $\beta = 25^\circ$ and $\phi = 40^\circ$. They are compared with the trajectory of the rocket that drop freely with the initial airspeed of the aircraft (the thin solid plot). It is important to remark that there is a small distance from the initial height of the rocket and the aircraft, representing the distance from the respective centers of mass.

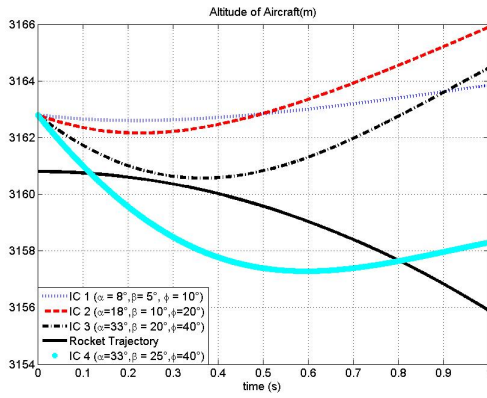


Fig. 5. Altitude of the aircraft

In the three first cases (dot plot, dash plot and dash dot plot), the altitude of the aircraft satisfies the specification that requires there is no collision between the aircraft and the rocket in the airlaunch phase. However, the aircraft approaches the rocket in the third case. In the last case (solid plot), there is an intersection of aircraft and the rocket before the ascent of the aircraft. This can be explained by the fact that the altitude of aircraft is function of angle of attack, roll angle, pitch angle and sideslip angle by (31). For some initial conditions of angles, the intersection between the aircraft and the rocket trajectories will happen after the airlaunch instant.

$$\dot{h} = -V(\cos \alpha \cos \beta \sin \theta - \sin \beta \cos \theta \sin \phi - \sin \alpha \cos \beta \cos \phi \cos \theta) \quad (31)$$

V. CONCLUSION

We have presented the design of a controller aiming to stabilize an unmanned aircraft performing the airlaunch of a satellite launcher (rocket). The proposed control strategy was based on the conditional integrator theory. This controller is composed of a longitudinal part for angle of attack control and a lateral one for the sideslip and roll angle control. These two controllers are designed using an F-16 model representing an aircraft performing airlaunch, just after dropping the second stage. The effects of the proposed controller are illustrated in computer simulations with several initial conditions distinct from the equilibrium. In three

of the studied cases, the stability of the system after the drop phase is assured and there is no collision between the aircraft and the rocket, even if when the initial error becomes large, the states are poorly stabilized causing bad transients. Moreover, for very large initial error conditions, there may be a collision between the two systems after the airlaunch instant. This can either illustrate the limitations of the proposed controller or the limitations of the aircraft itself. In future works both possibilities will be explored, as well as to study other control strategies for this particularly interesting and difficult problem.

REFERENCES

- [1] Gilney Damm and Van Cuong Nguyen. Mimo conditional integrator control for a class of nonlinear systems. 2011.
- [2] G.R. Damm, R. Marino, and F. Lamnabhi-Lagarigue. Adaptive nonlinear output feedback for transient stabilization and voltage regulation of power generators with unknown parameters. *Int. J. of Robust and Nonlinear Control*, 14 :833–855, 2004.
- [3] Thor I. Fossen. *Guidance and Control of Ocean Vehicles*. Wiley, 1994.
- [4] Gary C. Hudson. Quickreach responsive launch system. *American Institute of Aeronautics and Astronautics*, 2006.
- [5] Hassan K. Khalil. On the design of robust servomechanisms for minimum phase nonlinear systems. *Robust Nonlinear Control 2000*, John Wiley & Sons, 10 :339–361, 2000.
- [6] Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, 2001.
- [7] Taeyoung Lee and Youdan Kim. Nonlinear adaptive flight control using backstepping and neural networks controller. *Journal of guidance, control and dynamics*, 24 :675–682, 2001.
- [8] Attaullah Y. Memon and Hassan K. Khalil. Output regulation of nonlinear systems using conditional servocompensators. *Automatica*, 2010.
- [9] Eugene Morelli. Global nonlinear parametric modeling with application to f-16 aerodynamics. *American Institute of Aeronautics and Astronautics*, 1998.
- [10] L.T. Nguyen, M.E. Ogburn, and P. Deal. Simulator study of fighter airplane with relaxed longitudinal static stability. *Technical report NASA*, page 1538, 1979.
- [11] P. Tomei R. Marino. *Nonlinear Control Design - Geometric, Adaptive and Robust*. Prentice Hall, Hemel Hempstead, London, 1995.
- [12] John W.C. Robinson. Block backstepping for nonlinear flight control law design. *Springer*, page 231–257, 2007.
- [13] Jan Roskam. *Airplane Flight Dynamic and Automatic Flight Controls, Part 1. Design, Analysis and Research Corporation*, 2001.
- [14] Marti Sarigul-Klijn and Nesrin Sarigul-Klijn. Selection of a carrier aircraft and a launch method for air launching space vehicles. *American Institute of Aeronautics and Astronautics*, 2008.
- [15] S. Seshagiri and H.K. Khalil. Robust output feedback regulation of minimum-phase nonlinear systems using conditional integrators. *Automatica*, 41 :43–54, 2005.
- [16] Sridhar Seshagiri and Hassan K. Khalil. Robust output feedback regulation of minimum-phase nonlinear systems using conditional integrators. *Automatica*, 2004.
- [17] Sridhar Seshagiri and Hassan K. Khalil. Robust output feedback regulation of minimum-phase nonlinear systems using conditional integrators. *Automatica*, 2005.
- [18] Sridhar Seshagiri and Hassan K. Khalil. Robust output regulation of minimum phase nonlinear systems using conditional servocompensators. *Wiley InterScience*, 2005.
- [19] Sridhar Seshagiri and Ekpraisit Promtun. Sliding mode control of f-16 longitudinal dynamics. *American Control Conference*, pages 1770–1775, 2008.
- [20] Jean-Jacques Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice Hall, 1991.
- [21] Lars Sonneveldt. *Nonlinear F-16 Model Description*. Delft University of Technology, Netherlands, 2006.
- [22] Peter H. Zipfel. *Modeling and Simulation of Aerospace Vehicle Dynamics, 2nd edition*. American Institute of Aeronautics and Astronautics, 2000.

Robust Control Synthesis of Bilinear Systems with Nonlinear Weights by Sum of Squares Optimization

Eitaku Nobuyama, Takahiko Aoyagi and Yasushi Kami

Abstract—Robust control problems for nonlinear systems are usually formulated as L_2 -induced norm minimization problems and it is known that those problems are reduced to the solvability of Hamilton-Jacobi equations. However, in the case of bilinear systems the usual L_2 -induced norm minimization problem leads to an obvious solution (the zero input). To avoid the obvious solution Simizu et al. (1997) introduced nonlinear weights on the evaluated signal and proposed a design method using linearization of the state-dependent matrix Riccati inequality derived from the Hamilton-Jacobi equation. In contrast to this, the purpose of the present paper is to propose a new design method using SOS (Sum-of-Squares) optimization technique without linearization.

I. INTRODUCTION

Robust control problems for nonlinear systems are usually formulated as L_2 -induced norm minimization problems and those problems are reduced to the solvability of the so-called “Hamilton-Jacobi equation” (see, for example, [1], [8] and references therein). However, in the case of bilinear systems the usual L_2 -induced norm minimization problem leads to an obvious solution (the zero input is optimal!). To avoid the obvious solution [7] introduced nonlinear weights on the evaluated signal and proposed a design method using linearization of the state-dependent matrix Riccati inequality derived from the Hamilton-Jacobi equation. In contrast to this, the purpose of this paper is to propose a new design method using SOS (Sum-of-Squares) optimization without linearization.

It is known that the Hamilton-Jacobi equality coming from the L_2 -induced norm minimization problem is reduced to the solvability of an inequality condition of quadratic form, i.e.,

$$h^T(x)M(x)h(x) \geq 0, \quad \forall x \quad (1)$$

and this inequality is moreover reduced to the following matrix positive semi-definiteness condition:

$$M(x) \succeq 0, \quad \forall x \quad (2)$$

where $M(x)$ is a Riccati-type matrix including the state variables. This matrix inequality is usually called “a state-dependent matrix Riccati inequality” derived from the L_2 -induced norm optimization problem. Most papers have tried to find a solution to the matrix inequality (2) so far. See,

This work is partially supported by Regional Innovation Cluster Program (Global Type [2nd Stage]). The open access version is found in [9].

E. Nobuyama and T. Aoyagi are with Department of Systems Design and Informatics, Kyushu Institute of Technology, Iizuka-city, Fukuoka 820-8502, Japan nobuyama@ces.kyutech.ac.jp

Y. Kami is with Department of Electrical and Computer Engineering, Akashi National College of Technology, Akashi-city, Hyogo 674-8501, Japan kami@akashi.ac.jp

for example, [5], [2] and the references therein. However, it should be noted that the condition (2) is just a sufficient condition for (1) unless $h(x)$ is independent of x , because $M(x)$ includes x . In most L_2 -induced norm optimization cases, $h(x)$ includes x (in our case $h(x) = P^{-1}x$ as shown in the later section) and hence the methods of [5], [2] and other papers based upon the condition (2) can have significant conservativeness. Note that [2] proposed a redesign method for reducing the conservativeness; however, it has to find a solution to (2) before applying the redesign method. Hence, the redesign method cannot be applied if the matrix inequality (2) does not have a solution.

In the present paper, to avoid the conservativeness we propose a new method for finding a solution to (1) directly without finding a solution to (2). A key idea of our method is to treat the dependency of $h(x)$ with x as an equality condition and formulate the problem to be concerned as an SOS (Sum of Squares) optimization problem with an equality constraint. After that we apply SOS optimization technique to the problem to propose an iterative algorithm for finding a robust feedback controller.

In this paper, the following notations are used:

$\mathbf{R}$	the set of real numbers.
$\mathbf{Z}$	the set of integers.
$\mathbf{Z}_+$	the set of non-negative integers.
$\mathbf{R}[x]$	the set of polynomials in x .
M^T	the transpose of the matrix M .
$\otimes$	the Kronecker product.
Σ	the set of SOS polynomials. In particular, Σ_x denotes the set of SOS polynomials in x .
I	an identity matrix of appropriate size. (I_r denotes the $r \times r$ identity matrix.)

For a square matrix M , $M \succ 0$ and $M \succeq 0$ imply that M is positive definite and positive semi-definite, respectively.

II. PROBLEM STATEMENT

A. Plant and nonlinear weights

Consider the following bilinear systems:

$$\begin{aligned} \dot{x}_p(t) &= A_p x_p(t) + B_{p1} w(t) + \sum_{i=1}^{n_q} B_{p2i} x_p(t) u_i(t) \\ &= A_p x_p(t) + B_{p1} w(t) + B_{p2}(x) u(t) \end{aligned} \quad (3)$$

$$z_p(t) = C_p x_p(t) \quad (4)$$

where $x_p \in \mathbf{R}^{n_p}$ is the state variable, $u := [u_1 \cdots u_{n_q}]^T \in \mathbf{R}^{n_q}$ is the input, $w \in \mathbf{R}^{n_w}$ is the exogenous input, $z_p \in \mathbf{R}^r$

is the output to be evaluated, and the matrices A_p, B_{p1}, B_{p2i} ($i = 1, \dots, n_q$) are real matrices of appropriate sizes with

$$B_{p2}(x) := \sum_{i=1}^{n_q} B_{p2i} x_i.$$

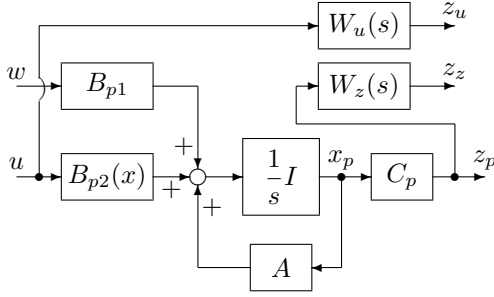


Fig. 1. Plant with frequency weights

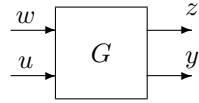


Fig. 2. Generalized plant

In this paper, we consider to evaluate the L_2 -induced norm from w to z_p and u with the following frequency weights $W_z(s)$ and $W_u(s)$, respectively (see Fig. 1):

$$W_z(s) : \begin{cases} \dot{x}_z(t) = A_z x_z(t) + B_z z_p(t), \\ z_z(t) = C_z x_z(t) + D_z z_p(t), \end{cases} \quad (5)$$

$$W_u(s) : \begin{cases} \dot{x}_u(t) = A_u x_u(t) + B_u u(t), \\ z_u(t) = C_u x_u(t) + D_u u(t) \end{cases} \quad (6)$$

where $x_z \in \mathbb{R}^{n_z}$, $x_u \in \mathbb{R}^{n_u}$ and the matrices in (5) and (6) are real matrices of appropriate size.

Here, we assume that the state variable is available for feedback. Then the plant with the frequency weights in Fig. 1 can be represented as a generalized plant G in Fig. 2 which is given by

$$\dot{x}(t) = Ax(t) + B_1 w(t) + B_2(x)u(t) \quad (7)$$

$$z(t) = \begin{bmatrix} C_{11} \\ C_{12} \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ D_{12} \end{bmatrix} u(t) \quad (8)$$

$$y(t) = x(t) \quad (9)$$

where $y(t)$ is the output for feedback and

$$x(t) = \begin{bmatrix} x_p(t) \\ x_z(t) \\ x_u(t) \end{bmatrix}, \quad z(t) = \begin{bmatrix} z_z(t) \\ z_u(t) \end{bmatrix}, \quad (9)$$

$$A = \begin{bmatrix} A_p & 0 & 0 \\ B_z C_p & A_z & 0 \\ 0 & 0 & A_u \end{bmatrix}, \quad B_1 = \begin{bmatrix} B_{p1} \\ 0 \\ 0 \end{bmatrix}, \quad (10)$$

$$B_2(x) = \begin{bmatrix} B_{p2}(x) \\ 0 \\ B_u \end{bmatrix}, \quad C_{11} = \begin{bmatrix} D_z C_p & C_z & 0 \end{bmatrix}, \quad C_{12} = \begin{bmatrix} 0 & 0 & C_u \end{bmatrix}, \quad D_{12} = D_u. \quad (11)$$

Let n ($:= n_p + n_z + n_u$) denote the dimension of x .

The purpose of this paper is to find a feedback controller which reduces the effect of w on z . For this purpose, the problem to minimize the L_2 -induced norm from w to z defined by

$$\sup_{w \neq 0} \frac{\|z\|_2}{\|w\|_2} \quad (12)$$

is usually considered where $\|\cdot\|_2$ denotes the L_2 norm. However, the bilinear system (3) is uncontrollable for $x = 0$ because of $B_2(0) = 0$, so that the effect of w cannot be reduced around $x = 0$. Moreover, it is known that the zero input $u = 0$ is optimal for the problem of minimizing the L_2 -induced norm (12) when the evaluated variable z is affine in x and u . Hence, the minimization problem with respect to the L_2 -induced norm (12) is no use for our purpose.

Although the system is uncontrollable at $x = 0$, the system behavior can be improved by some proper controllers except at $x = 0$. To formulate the problem of finding such controllers [7] introduce nonlinear weights on z . It is shown by them that the obvious solution (the zero input) can be avoided by introducing the nonlinear weights.

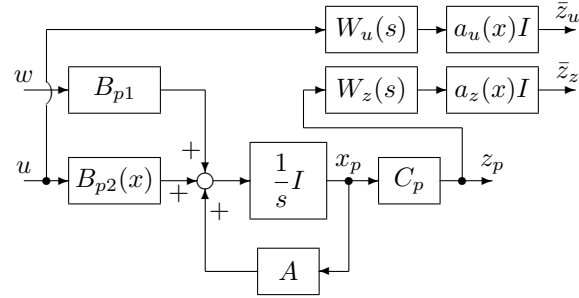


Fig. 3. Plant with nonlinear weights

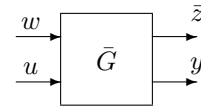


Fig. 4. Generalized plant including nonlinear weights

As in the paper of [7] we will also introduce nonlinear weights on z as shown in Fig. 3 where $a_z(x)$ and $a_u(x)$ are the nonlinear weights which are functions of x .

With the introduction of the nonlinear weights the new generalized plant $\bar{G}$ shown in Fig. 4 is represented as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 w(t) + B_2(x)u(t), \\ \bar{z}(t) &= \begin{bmatrix} a_z(x)C_{11} \\ a_u(x)C_{12} \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ a_u(x)D_{12} \end{bmatrix} u(t), \\ y(t) &= x(t) \end{aligned} \quad (13)$$

where

$$\bar{z}(t) := \begin{bmatrix} \bar{z}_z(t) \\ \bar{z}_u(t) \end{bmatrix} = \begin{bmatrix} a_z(x)z_z(t) \\ a_u(x)z_u(t) \end{bmatrix}.$$

Then the problem to be considered in this paper is formulated as the one of finding the feedback controller which minimizes the L_2 -induced norm from w to $\bar{z}$ defined by

$$\sup_{w \neq 0} \frac{\|\bar{z}\|_2}{\|w\|_2}. \quad (14)$$

The next theorem is shown by [3], [7] using linealization.

Theorem 1: Consider the bilinear system (13) with $C_{12} = 0$. For given $\gamma > 0$ suppose that there exists a positive definite symmetric matrix P which satisfies

$$PA + A^T P + \frac{1}{\gamma^2} P B_1 B_1^T P + C_{11}^T C_{11} \prec 0 \quad (15)$$

and the nonlinear weights $a_z(x)$ and $a_u(x)$ satisfy the condition

$$\frac{1}{a_u^2(x)} x^T P B_2(x) B_2^T(x) P x + (1 - a_z^2(x)) x^T C_{11}^T C_{11} x \geq 0. \quad (16)$$

Then the L_2 -induced norm from w to $\bar{z}$ is less than or equal to γ via the feedback control

$$u(t) = -\frac{1}{a_u^2(x)} B_2^T(x) P x(t). \quad (17)$$

This theorem gives a method for choosing the nonlinear weights after P is obtained; however, this means that the nonlinear weights cannot be chosen before obtaining P and the condition (16) restricts the choice of the nonlinear weights. In contrast to this, in our method given below the nonlinear weights can be chosen a priori and the condition which they have to satisfy is just that $a_z^2(x) \in \mathbf{R}[x]$ and $1/a_u^2(x) \in \mathbf{R}[x]$.

A typical choice of the nonlinear weights is as follows:

$$a_z(x) = \sqrt{1 + x^T R_z x}, \quad a_u(x) = \frac{1}{\sqrt{1 + x^T R_u x}} \quad (18)$$

where $R_x \succeq 0$ and $R_u \succeq 0$. Fig. 5 shows an example in the case of $x \in \mathbf{R}$. The weight $a_z(x)$ shown in Fig. 5 is utilized for suppressing the effect of w on z_z and $a_u(x)$ is for allowing large input values except at $x = 0$.

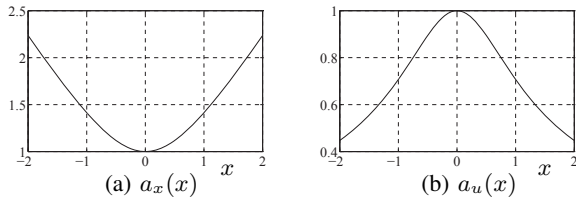


Fig. 5. Example of nonlinear weights

B. Derivation of state-dependent inequalities

In the sequel, we assume $D_{12} = I$ for simplicity. Then we have the next theorem.

Theorem 2: Suppose that for given $\gamma > 0$ there exists a positive definite symmetric matrix P which satisfies the

following state-dependent inequality:

$$\begin{aligned} \phi(x, P) := & x^T \left[-P^{-1} (A + B_2(x) C_{12}) - (A + B_2(x) C_{12})^T P^{-1} \right. \\ & - P^{-1} \left(\frac{1}{\gamma^2} B_1 B_1^T - \frac{1}{a_u^2(x)} B_2(x) B_2^T(x) \right) P^{-1} \\ & \left. - a_z^2(x) C_{11}^T C_{11} \right] x > 0, \quad \forall x (\neq 0) \in \mathbf{R}^n \end{aligned} \quad (19)$$

Then by the feedback

$$u(t) = - \left(\frac{1}{a_u^2(x)} B_2^T(x) P^{-1} + C_{12} \right) x(t) \quad (20)$$

the closed-loop system is asymptotically stable and the L_2 -induced norm (14) is less than or equal to γ , i.e.,

$$\sup_{w \neq 0} \frac{\|\bar{z}\|_2}{\|w\|_2} \leq \gamma. \quad (21)$$

Proof: First, we will show that the closed-loop system via the feedback (20) with $w(t) = 0$ is stable when (19) holds. To show this we adopt $V(t) = x^T(t) P^{-1} x(t)$ as a Lyapunov function candidate. Then we have from (20) and (19)

$$\begin{aligned} & \frac{d}{dt} V(t) \\ &= \dot{x}^T P^{-1} x + x^T P^{-1} \dot{x} \\ &= (Ax + B_2(x)u)^T P^{-1} x + x^T P^{-1} (Ax + B_2(x)u) \\ &= \left[Ax - B_2(x) \left(\frac{1}{a_u^2(x)} B_2^T(x) P^{-1} + C_{12} \right) x \right]^T P^{-1} x \\ & \quad + x^T P^{-1} \left[Ax - B_2(x) \left(\frac{1}{a_u^2(x)} B_2^T(x) P^{-1} + C_{12} \right) x \right] \\ &< -x^T \left[\frac{1}{\gamma^2} P^{-1} B_1 B_1^T P^{-1} + a_z^2 C_{11}^T C_{11} \right. \\ & \quad \left. + \frac{1}{a_u^2} P^{-1} B_2(x) B_2^T(x) P^{-1} \right] x \leq 0, \quad \text{for } x \neq 0. \end{aligned} \quad (22)$$

This shows the closed-loop system is asymptotically stable.

Next, from (13) we have

$$\begin{aligned} \gamma^2 |w|^2 - |z|^2 &= \gamma^2 |w|^2 - a_z^2 x^T C_{11}^T C_{11} x - a_u^2 x^T C_{12}^T C_{12} x \\ & \quad - a_u^2 u^T u - 2a_u^2 x^T C_{12}^T u \end{aligned} \quad (23)$$

and the following identity holds:

$$\begin{aligned} 0 &= 2x^T P^{-1} \dot{x} - 2x^T P^{-1} \dot{x} \\ &= 2x^T P^{-1} \dot{x} - 2x^T P^{-1} A x - 2x^T P^{-1} B_1 w \\ & \quad - 2x^T P^{-1} B_2(x) u. \end{aligned} \quad (24)$$

By adding the both-sides of (24) to (23) and completing the square we have

$$\begin{aligned}
& \gamma^2 |w|^2 - |z|^2 \\
&= 2x^T P^{-1} \dot{x} + \gamma^2 w^T w - 2x^T P^{-1} B_1 w \\
&\quad - a_u^2 \left[u^T u + 2 \left(\frac{1}{a_u^2} x^T P^{-1} B_2(x) + x^T C_{12}^T \right) u \right] \\
&\quad - x^T [2P^{-1} A + a_z^2 C_{11}^T C_{11} + a_u^2 C_{12}^T C_{12}] x \\
&= 2x^T P^{-1} \dot{x} + \gamma^2 \tilde{w}^T \tilde{w} - \frac{1}{\gamma^2} x^T P^{-1} B_1 B_1^T P^{-1} x \\
&\quad + a_u^2 x^T \left(\frac{1}{a_u^2} P^{-1} B_2(x) + C_{12}^T \right) \\
&\quad \quad \times \left(\frac{1}{a_u^2} B_2^T(x) P^{-1} + C_{12} \right) x \\
&\quad - x^T [2P^{-1} A + a_z^2 C_{11}^T C_{11} + a_u^2 C_{12}^T C_{12}] x - a_u^2 \tilde{u}^T \tilde{u} \\
&= 2x^T P^{-1} \dot{x} + \gamma^2 \tilde{w}^T \tilde{w} - a_u^2 \tilde{u}^T \tilde{u} + \phi(x, P) \quad (25)
\end{aligned}$$

where

$$\begin{aligned}
\tilde{w} &= w - \frac{1}{\gamma^2} B_1^T P^{-1} x, \\
\tilde{u} &= u + \left(\frac{1}{a_u^2} B_2^T(x) P^{-1} + C_{12} \right) x.
\end{aligned}$$

Then from (19) and (20)

$$\gamma^2 |w|^2 - |z|^2 \geq 2x^T P^{-1} \dot{x} + \gamma^2 \tilde{w}^T \tilde{w},$$

and hence

$$\begin{aligned}
& \int_0^\tau (\gamma^2 |w|^2 - |z|^2) dt \geq \int_0^\tau (2x^T P^{-1} \dot{x} + \gamma^2 \tilde{w}^T \tilde{w}) dt \\
&= x^T(\tau) P^{-1} x(\tau) - x^T(0) P^{-1} x(0) + \int_0^\tau (\gamma^2 \tilde{w}^T \tilde{w}) dt.
\end{aligned}$$

Here, let $x(0) = 0$ and $\tau \rightarrow \infty$ then

$$\gamma^2 \|w\|^2 - \|z\|^2 = \int_0^\infty (\gamma^2 \tilde{w}^T \tilde{w}) dt \geq 0, \quad (26)$$

which leads to (21). Note that to derive (26) we use $\lim_{\tau \rightarrow \infty} x(\tau) = 0$ which holds because the closed-loop is asymptotically stable. Q.E.D.

Note that $\phi(x, p)$ in (19) can be represented as

$$\phi(x, P) = (P^{-1} x)^T M(x, P) (P^{-1} x) \quad (27)$$

where

$$\begin{aligned}
M(x, P) &:= -(A + B_2(x) C_{12}) P - P(A + B_2(x) C_{12})^T \\
&\quad - \left(\frac{1}{\gamma^2} B_1 B_1^T - \frac{1}{a_u^2(x)} B_2(x) B_2^T(x) \right) \\
&\quad - a_z^2(x) P C_{11}^T C_{11} P. \quad (28)
\end{aligned}$$

From this we have the next corollary.

Corollary 1: Suppose that for given $\gamma > 0$ there exists a positive definite symmetric matrix P which satisfies the following state-dependent inequality:

$$M(x, P) \succ 0, \quad \forall x \in \mathbf{R}^n \quad (29)$$

Then by the feedback (20) the closed-loop system is asymptotically stable and the L_2 -induced norm (14) is less than or equal to γ .

Proof: It is obvious from (27) that (19) is satisfied if (29) holds. Hence we obtain this corollary from Theorem 2. Q.E.D.

The inequality (29) is called “a state-dependent matrix Riccati inequality” and equivalent to

$$\begin{aligned}
& h^T [-(A + B_2(x) C_{12}) P - P(A + B_2(x) C_{12}) \\
&\quad - \left(\frac{1}{\gamma^2} B_1 B_1^T - \frac{1}{a_u^2(x)} B_2(x) B_2^T(x) \right) \\
&\quad - a_z^2(x) P C_{11}^T C_{11} P] h > 0, \\
& \forall x \in \mathbf{R}^n, \quad h(\neq 0) \in \mathbf{R}^n. \quad (30)
\end{aligned}$$

Note that h is independent of x in (30), whereas there is a relationship of $h = P^{-1}x$ between h and x in (19). This means that the condition (30) can be very conservative compared with the condition (19). As mentioned in Introduction, most papers have tried to find P which satisfies a matrix state-dependent inequality like (29) (or (30)). In contrary to this, we will try to find P which satisfies (19) (instead of (30)) to reduce the conservativeness.

III. SUM OF SQUARES

In this section, we briefly survey the so-called “SOS (Sum of Squares) optimization.”

A. Definitions and basic properties

A monomial in $x = [x_1 \ \cdots \ x_n]^T$ is represented as $x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ with $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbf{Z}_+^n$. This is also written as $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$. The degree of a monomial x^α , denoted by $\deg(x^\alpha)$, is defined by $\deg(x^\alpha) := \sum_{i=1}^n \alpha_i$ and the degree of a polynomial $f(x) \in \mathbf{R}[x]$, denoted by $\deg(f)$, is defined by the degree of the monomial which has the highest degree among all the monomials included in $f(x)$. For a polynomial matrix $F(x) \in \mathbf{R}^{q \times r}$ the degree of $F(x)$, denoted by $\deg(F)$, is defined by $\deg(F) := \max_{ij} \deg(F_{ij})$ where F_{ij} denotes the (i, j) element of $F(x)$.

A real polynomial $f(x) \in \mathbf{R}[x]$ is said to be an *SOS (Sum of Squares) polynomial* if it can be represented as a sum of squares of some polynomials, i.e., there exist some polynomials $g_i(x) \in \mathbf{R}[x]$ ($i = 1, \dots, p$) such that

$$f(x) = \sum_{i=1}^p g_i^2(x). \quad (31)$$

Moreover, a polynomial symmetric matrix $F(x) \in \mathbf{R}^{r \times r}[x]$ is said to be an *SOS matrix* if it can be represented as

$$F(x) = L^T(x) L(x)$$

for some polynomial matrix $L(x)$ of appropriate size. In this paper, we denote the set of SOS polynomials by Σ , and the set of $r \times r$ SOS matrices by $\Sigma^{r \times r}$.

From the definitions it is obvious that

$$f(x) \in \Sigma \Rightarrow f(x) \geq 0 \quad (\forall x \in \mathbf{R}^n), \quad (32)$$

$$F(x) \in \Sigma^{r \times r} \Rightarrow F(x) \succeq 0 \quad (\forall x \in \mathbf{R}^n). \quad (33)$$

Here, for a positive integer d let $v_d(x)$ be a polynomial vector in x of size $_{n+d}C_d$ defined by

$$v_d(x) := \begin{bmatrix} 1 & x_1 & \cdots & x_n & x_1^2 & x_1x_2 & \cdots & x_n^2 \\ \cdots & x_1^d & \cdots & x_n^d \end{bmatrix}^T, \quad (34)$$

which contains all monomials whose degrees are less than or equal to d where $_{\alpha}C_{\beta} = \alpha! / (\beta!(\alpha - \beta)!)$ for positive integers α and β with $\alpha \geq \beta$. Then the next lemmas are known.

Lemma 1: ([4]) Let $f(x) \in \mathbf{R}[x]$ with $\deg(f) = 2d$. Then the following (i) and (ii) are equivalent:

- (i) $f(x) \in \Sigma$.
- (ii) There exists a positive semi-definite symmetric matrix of appropriate size such that $f(x) = v_d^T(x)Qv_d(x)$.

Lemma 2: ([6]) Let $F(x) \in \mathbf{R}[x]^{r \times r}$ with $\deg(F) = 2d$. Then the following (i) and (ii) are equivalent:

- (i) $F(x) \in \Sigma^{r \times r}$.
- (ii) There exists a positive semi-definite symmetric matrix of appropriate size Q such that $F(x) = (v_d(x) \otimes I_r)^T Q (v_d(x) \otimes I_r)$.

Using these lemmas, the problem of determining whether a polynomial $f(x)$ (or a polynomial matrix $F(x)$) is an SOS polynomial (or an SOS matrix) or not is reduced to an SDP (Semi-Definite Programming) problem, which can be solved numerically, of checking the positive semi-definiteness of the corresponding matrix Q .

B. SOS polynomials with equality constraints

Let us consider the following equality constraints:

$$f_j(x) = 0, \quad j = 1, \dots, p \quad (35)$$

where $f_j(x) \in \mathbf{R}[x]$ and their feasible set is defined by

$$S := \{x \in \mathbf{R}^n \mid f_j(x) = 0, \quad j = 1, \dots, p\}. \quad (36)$$

Here we consider the problem of determining whether a given polynomial $f_0(x) \in \mathbf{R}$ is non-negative or not for all $x \in \mathbf{R}$ in the feasible set, i.e., the following condition holds or not:

$$f_0(x) \geq 0, \quad \forall x \in S. \quad (37)$$

For this problem, define a generalized Lagrange function $L(x, \lambda)$ by

$$L(x, \lambda) := f_0(x) - \sum_{j=1}^p \lambda_j(x) f_j(x) \quad (38)$$

where $\lambda_j(x) \in \mathbf{R}[x]$ and let

$$\lambda(x) := \begin{bmatrix} \lambda_1(x) & \cdots & \lambda_p(x) \end{bmatrix}^T \in \mathbf{R}[x]^p.$$

Then if for given $\lambda[x] \in \mathbf{R}[x]^p$

$$L(x, \lambda) \geq 0, \quad \forall x \in \mathbf{R}^n \quad (39)$$

holds, the condition (37) is satisfied. In fact, (39) implies

$$f_0(x) \geq \sum_{j=1}^p \lambda_j(x) f_j(x) = 0, \quad \forall x \in S. \quad (40)$$

Moreover, if $L(x, \lambda) \in \Sigma$ then (39) holds from Lemma 1 and hence (37) holds. These facts are summarized in the following lemma.

Lemma 3: If the following (i) or (ii) holds, the condition (37) holds.

- (i) There exists $\lambda(x) \in \mathbf{R}[x]^p$ such that $L(x, \lambda) \geq 0$ ($\forall x \in \mathbf{R}^n$).
- (ii) There exists $\lambda(x) \in \mathbf{R}[x]^p$ such that $L(x, \lambda) \in \Sigma$.

IV. PROPOSED METHOD

Theorem 2 implies that the state feedback (20) will stabilize the closed-loop system and (25) is satisfied if we can obtain a positive definite symmetric matrix P satisfying (19). In this section, we propose an SOS optimization method to find such P .

To this end, let us introduce sufficiently small $\epsilon > 0$ and define

$$\tilde{M}(x, P) := M(x, P) - \epsilon I, \quad (41)$$

$$\tilde{\phi}(x, P) := (P^{-1}x)^T \tilde{M}(x, P) (P^{-1}x). \quad (42)$$

Then it is easy to see

$$\tilde{M}(x, P) \succeq 0, \quad \forall x \in \mathbf{R}^n \Rightarrow M(x, P) \succ 0, \quad \forall x \in \mathbf{R}^n,$$

$$\tilde{\phi}(x, P) \geq 0, \quad \forall x \in \mathbf{R}^n \Rightarrow \phi(x, P) > 0, \quad \forall x(\neq 0) \in \mathbf{R}^n.$$

Hence, for obtaining the feedback (20) it suffices to find $P \succ 0$ such that

$$\tilde{\phi}(x, P) \geq 0, \quad \forall x \in \mathbf{R}^n. \quad (43)$$

From (42), the condition (43) can be written as

$$h^T \tilde{M}(x, P) h \geq 0, \quad \forall (x, h) \in \mathbf{R}^{2n} \text{ such that } h = P^{-1}x$$

and moreover this can be written as

$$h^T \tilde{M}(x, P) h \geq 0, \quad \forall (x, h) \in \tilde{S} \quad (44)$$

where

$$\tilde{S} := \{(x, h) \in \mathbf{R}^{2n} \mid x - Ph = 0\}. \quad (45)$$

By this, the condition (43) is represented as the condition (44) including the equality constraint $x - Ph = 0$. For the condition (44) we define a generalized Lagrange function as in Section III-B as follows:

$$L(x, h, \lambda; P) := h^T \tilde{M}(x, P) h - \lambda^T(x, h)(x - Ph) \quad (46)$$

where $\lambda(x, h) \in \mathbf{R}[x, h]^n$. Then, from Lemma 3, (44) is satisfied if there exist λ and $P(\succ 0)$ which satisfies

$$L(x, h, \lambda; P) \geq 0, \quad \forall (x, h) \in \mathbf{R}^{2n} \quad (47)$$

Here, suppose the degree of λ is given, say m , then λ can be written as

$$\lambda(x, h) = H v_m(x, h)$$

where $v_m(x, h)$ is a vector of size $2n+mC_m$ which contains all monomials in x and h whose degrees are less than or equal to m , and H is an $n \times (2n+mC_m)$ real matrix. From this, (47) is reduced to

$$L_m(x, h; H, P) := h^T \tilde{M}(x, P)h - v_m^T(x, h)H^T(x - Ph) \geq 0, \quad \forall (x, h) \in \mathbf{R}^{2n}, \quad (48)$$

and the problem to be concerned becomes the one for finding matrices P and H which satisfies (48).

Note that $L_m(x, h; H, P)$ includes the product of H and P in the last term. Hence, we consider an iterative algorithm which repeats a step of finding H for fixed P and a step of finding P for fixed H .

First, suppose P is fixed. In this case, $L_m(x, h; H, P)$ can be written as

$$L_m(x, h; H, P) = v_{d_1}^T(x, h)Q_1(H)v_{d_1}(x, h) \quad (49)$$

where $d_1 = \deg(L_m)$, the degree of L_m as a polynomial in x and h , $v_{d_1}(x, h)$ is a vector of size $2n+d_1C_{d_1}$ which contains all monomials in x and h whose degrees are less than or equal to d_1 , and $Q_1(H)$ is a $(2n+d_1C_{d_1}) \times (2n+d_1C_{d_1})$ symmetric matrix. Then $Q_1(H)$ is affine in H because so is L_m . Hence, in the case of fixed P , the problem to be concerned is reduced to an SDP problem to find H such that $Q_1(H) \succeq 0$, because (48) is satisfied if $L_m \in \Sigma_{(x, h)}$ which is equivalent to the existence of $Q_1(H) \succeq 0$ by Lemma 1.

Next, suppose H is fixed. In this case, L_m is not affine in P , but by Schur complement (48) is equivalent to

$$G(x, h; H, P) := \begin{bmatrix} G_{11}(x, h; H, P) & a_z^2(x)h^T PC_{11}^T \\ a_z^2(x)C_{11}Ph & a_z^2(x)I_{n_z} \end{bmatrix} \succeq 0, \quad \forall (x, h) \in \mathbf{R}^{2n}, \quad (50)$$

which is affine in P , where

$$\begin{aligned} G_{11}(x, h; H, P) &:= h^T [-(A + B_2(x)C_{12})P - P(A + B_2(x)C_{12})^T \\ &\quad - \bar{\gamma}B_1B_1^T + \bar{a}_u^2(x)B_2(x)B_2^T(x)]h \\ &\quad - v_m^T(x, h)H^T(x - Ph) \end{aligned} \quad (51)$$

and

$$\bar{\gamma} := \frac{1}{\gamma^2}, \quad \bar{a}_u(x) := \frac{1}{a_u(x)}. \quad (52)$$

Since G is affine in P , it can be written as

$$G(x, h; H, P) = (v_{d_2}(x, h) \otimes I)^T Q_2(P)(v_{d_2}(x, h) \otimes I) \quad (53)$$

where $d_2 = \deg(G)$, the degree of G as a polynomial matrix in x and h , $v_{d_2}(x, h)$ is a vector of size $2n+d_2C_{d_2}$ which contains all monomials in x and h whose degrees are less than or equal to d_2 , and $Q_2(P)$ is a real symmetric matrix of appropriate size. Hence, in the case of fixed H , the problem to be concerned is reduced to an SDP problem to find P such that $Q_2(P) \succeq 0$, because (48) is satisfied if $G \in \Sigma_{(x, h)}^{(n_z+1) \times (n_z+1)}$ which is equivalent to the existence of $Q_2(H) \succeq 0$ by Lemma 2.

Note that L_m and G are also affine in $\bar{\gamma}$ and hence we can consider to maximize $\bar{\gamma} = 1/\gamma^2$ (i.e., minimize γ).

Now, let us summarize our method as an algorithm.

Algorithm 1:

Step0 Choose an initial value $P_0 \succ 0$ and small ϵ . Let $k := 0$ and $\bar{\gamma}_0 = 0$.

Step1 Let $P = P_k$ and get the optimal value $\bar{\gamma}^*$ and its optimizer H^* by solving numerically the next SDP problem

$$\max_{\bar{\gamma}, H} \bar{\gamma} \quad \text{s.t.} \quad Q_1(H) \succeq 0, \quad (54)$$

and let $H_k := H^*$.

Step2 Let $H = H_k$ and get the optimal value $\bar{\gamma}^*$ and its optimizer $P^* \succ 0$ by solving numerically the next SDP problem

$$\max_{\bar{\gamma}, P \succ 0} \bar{\gamma} \quad \text{s.t.} \quad Q_2(P) \succeq 0, \quad (55)$$

and let $\bar{\gamma}_{k+1} := \bar{\gamma}^*$ and $P_{k+1} = P^*$.

Step3 If $|\bar{\gamma}_{k+1} - \bar{\gamma}_k|$ is sufficiently small (i.e., $\bar{\gamma}_k$ is convergent), then return P_{k+1} as a solution and exit; otherwise, let $k := k + 1$ and go to Step 1.

Note that a feasible solution to $\tilde{M}(0, P) \succeq 0$ for large γ can be used as an initial value P_0 , because $\tilde{M}(0, P) \succeq 0$ is a usual Riccati inequality and has a feasible solution for large γ .

V. CONCLUSIONS

In this paper, we have derived an inequality condition of quadratic form for the robust control problem of bilinear systems with nonlinear weights, and moreover proposed an iterative method for finding a solution to the inequality condition using SOS optimization technique.

REFERENCES

- [1] J. Huang and C.-F. Lin, "Numerical approach to computing nonlinear H_∞ control laws," *Journal of Guidance, Control and Dynamics*, vol. 18, 1995, pp. 989-994.
- [2] H. Ichihara, "Optimal control for polynomial systems using matrix sum of squares relaxations," *IEEE Transactions on Automatic Control*, vol. 54, No. 5, 2009, pp. 1048-1053.
- [3] S. Ohsaku, T. Nakayama, I. Kamimura and Y. Motozono, "Nonlinear H_∞ state feedback controller for semi-active controlled suspension," *Proceedings of AVEC '98*, 1998, pp. 63-68.
- [4] P. A. Parrilo, "Semidefinite programming relaxations for semialgebraic problems," *Mathematical Programming*, vol. 96, 2003, pp. 293-320.
- [5] S. Prajna, S., A. Papachristodoulou, and F. Wu, (2004). "Nonlinear control synthesis by sum of squares optimization," *Proceedings of the Asian Control Conference 2004*, 2004, pp. 157-165.
- [6] C. W. Scherer and C. W. Hol, "Matrix sum-of-squares relaxations for robust semi-definite programs," *Mathematical Programming*, vol. 107, 2006, pp. 189-211.
- [7] E. Shimizu, M. Sampei and M. Koga, "Design of a nonlinear H_∞ state feedback controller for bilinear systems with Nonlinear Weight," *Proceedings of the 36th IEEE Conference on Decision and Control*, 1997, pp. 2323-2324.
- [8] A. J. van der Schaft, *L_2 -gain and passivity techniques in nonlinear control*, Springer-Verlag; 1996.
- [9] E. Nobuyama, T. Aoyagi and Y. Kami, "A sum-of-squares optimization approach to robust control of bilinear systems," *Robust Control - Volume 1*, InTech, 2011.

Some Aspects Concerning the Oscillations Damping in a Class of Affine Systems – a Driveline Simulation Example

Virginia Ecaterina Oltean, Radu Dobrescu, Dan Popescu, Maximilian Nicolae and Andreea Udrea

Abstract—For a class of affine systems with stable system matrix and oscillating behavior, the control objective consists of oscillations damping while tracking a piecewise constant reference signal. Under some controllability assumptions, a control solution consists of a pole assignment procedure combined with the internal model principle. The paper discusses, within a simulation experiment, a possible extension of this control approach to a constrained affine model of a vehicle driveline, from the literature. Two choices of the dominant poles are considered and the closed loop simulated behaviors are compared to the results reported for a model predictive control approach. The proposed control solution can be adapted for piecewise affine systems, modeled as affine hybrid automata.

I. INTRODUCTION

PIECEWISE affine systems represent a class of hybrid systems frequently encountered as modeling approximations of a large class of nonlinearities [1] arising both in control engineering [2], [3] and biological modeling [4], [5]. The control of oscillations is an important objective for piecewise affine systems, generally associated with driving quality and comfort in the evolution of automotive systems [6].

In order to analyze and control complex piecewise affine nonlinear dynamics, a prior study of affine systems is relevant for the description of the local behavior of the hybrid system. This paper firstly reviews a control approach for an affine system, ensuring oscillations damping and the tracking of a piecewise constant reference input. The two objectives are reached, under certain assumptions, using a classic pole assignment procedure combined with the internal model principle [7]. Secondly, this control approach is applied to a constrained affine model of a driveline presented in [8], with the parameter values specified in [9].

The paper is structured as follows. Section 2 reviews the control approach for affine unconstrained systems, illustrated by a second order simulation example. In Section 3, the control scheme for oscillations damping and step shaped reference tracking is adapted to the constrained affine driveline model in [8], an analysis of the open loop system behavior is presented, a qualitative poles assignment

philosophy is proposed, and, for two choices of the closed loop poles locations, the simulation results are comparatively analyzed.

II. AN OSCILLATIONS DAMPING CONTROL SOLUTION FOR AFFINE SYSTEMS – A REVIEW

A. The control problem

Consider the continuous-time affine system

$$\dot{x} = Ax + bu + f, \quad y = c^T x, \quad (1)$$

with $x \in \mathbf{R}^n$ the continuous state, u a scalar-valued input signal, y the scalar output and A , $b(\neq 0)$ and $f(\neq 0)$ real matrices of appropriate dimensions, respectively. Denote $\Lambda(A)$ the set of eigenvalues of A .

Assumption 1. (a) *The matrix A has at least one pair of complex eigenvalues, i.e. $\exists(\alpha \pm j\beta) \in \Lambda(A)$.* (b) *The pair (A, b) in (1) is controllable.*

Consider also a class of piecewise-constant exogenous signals $r: [0, T] \rightarrow \mathbf{R}$

$$r(t) = \begin{cases} r_1, & 0 \leq t < t_1 \\ \vdots & \\ r_i, & t_{i-1} \leq t < t_i \\ \vdots & \\ r_m, & t_{m-1} \leq t \leq T \end{cases}, \quad (2)$$

where $r_i \in \mathbf{R}$, $i = 1:m$, and the time intervals between any two consecutive switching moments t_i , $i = 1:m-1$, may not be equal, respectively.

Problem 1. *Given the affine system (1) satisfying the Assumption 1 and an exogenous signal r (2), find a control law $u(x, r)$ s.t. in closed loop: (i) the system is stable and the oscillations of the output y are damped and (ii) the output y tracks the reference signal (2).*

B. The control system: pole assignment, internal model and extended system

The first part of the control objective can be realized by a pole assignment strategy, implemented as a static feedback control law, while the second part of the control objective requires - for zero asymptotic tracking error - the presence of an internal model of the exogenous signal. Since the

This work was supported in part by the National Center for Programs Management from Romania under research grant SICONA-12100/2008.

The authors are with the Automatic Control and Industrial Informatics Department, Polytechnica University of Bucharest, Spl. Independenței 313, 77206 Bucharest, ROMANIA (e-mails: ecaterina.oltean@aii.pub.ro, radud@isis.pub.ro, dan_popescu_2002@yahoo.com)

reference signal (2) is a combination of delayed step signals, the tracking is ensured by an integrator as internal model. This solution is implemented in the control structure depicted in Fig.1, with the state equations

$$\begin{aligned}\dot{x} &= Ax + bu + f, \\ \dot{z} &= -c^T x + r,\end{aligned}\quad (3)$$

output signal $y = c^T x$ and tracking error $\varepsilon = r - y$.

Define the extended affine system of order $(n+1)$

$$\dot{x}_e = A_e x_e + b_e u + f_e + g_e r, \quad (4)$$

where $x_e = [x^T \ z]^T$ is the extended state, u is the control input, r is the reference signal to be tracked and the system matrices are

$$\begin{aligned}A_e &= \begin{bmatrix} A & 0_{n \times 1} \\ -c^T & 0 \end{bmatrix}, \quad b_e = \begin{bmatrix} b \\ 0 \end{bmatrix}, \\ f_e &= \begin{bmatrix} f \\ 0 \end{bmatrix}, \quad g_e = \begin{bmatrix} 0_{n \times 1} \\ 1 \end{bmatrix},\end{aligned}\quad (5)$$

and consider the following problem associated to Problem 1.

Problem 2 Given (i) the system (4)-(5), with (A, b) satisfying Assumption 1b and

$$\det \begin{bmatrix} A & b \\ -c^T & 0 \end{bmatrix} \neq 0 \quad (6)$$

and (ii) a set Λ_e of $(n+1)$ strictly negative real numbers, find a control law

$$u = k_e^T x_e = [k_o^T \ \varphi] \cdot \begin{bmatrix} x \\ z \end{bmatrix} \quad (7)$$

so that $\Lambda(A_e + b_e k_e^T) = \Lambda_e$.

The solution is based on the following two results.

Lemma 1 If the pair (A, b) , with $A \in \mathbf{R}^{n \times n}$ and $b \in \mathbf{R}^n$, is controllable and (A, b, c^T) satisfies (8), then the pair (A_e, b_e) , defined in (5), is also controllable.

The proof is given in [7]. The controllability of the extended pair (A_e, b_e) is necessary and sufficient for pole assignment, but, obviously, if (A, b) in (1) is not controllable, but the uncontrollable eigenvalues of A are real and stable then the uncontrollable “part” produces neither oscillations nor instability and, after a structural decomposition, the pole assignment procedure can be applied for the controllable “part”. In other words, a solution to Problem 2 is also a solution to Problem 1; however, a solution to Problem 1 can still be obtained in more relaxed conditions, as specified above.

The second part of the control objective concerns the asymptotic error behavior, for a reference signal from the

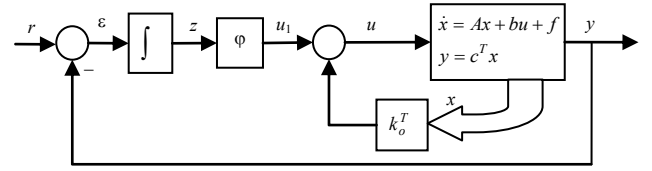


Fig.1. The control structure for the affine system in Problem 2.

class (2). After some computation [7], it results that, for $r(t) = r_i \cdot 1(t)$, with the Laplace transformed $r(s) = r_i/s$, $r_i > 0$, the stationary output value is

$$y(\infty) = \lim_{s \rightarrow 0} s y(s) = r_i, \quad (8)$$

where $y(s) = L[y(t)]$ denotes the Laplace transformed of the output signal, and the resulting the steady state error is $\varepsilon(\infty) = 0$.

The control procedure solving Problem 2 can be summarized as follows.

Control design algorithm

Input: the n -th order model (1) with (A, b, c^T) in observable canonical form, the reference signal (2), and desired closed loop spectrum $\Lambda_e = \{\lambda_i \mid \lambda_i < 0\}_{i=1:n+1}$.

Output: the feedback matrix $k_e^T = [k_o^T \ \varphi]$ in (7).

1. If the pair (A, b) is controllable, then go to step 2. Else, stop.
2. If the condition (6) is satisfied, then go to step 3. Else, stop.
3. Build the matrices (5) of the extended system (4).
4. For the pair (A_e, b_e) , build the feedback matrix $k_e^T = [k_o^T \ \varphi]$.
5. Given $\Lambda_{est} = \{\lambda_{est j} \mid \lambda_{est j} < 0\}_{j=1:n-1}$, the control law (7) is implemented using a minimal observer for the process state x in (3).
6. Stop.

C.A simulation example reviewed

Consider the second order affine system (1) in observable canonical form, with the matrices

$$\begin{aligned}A &= \begin{bmatrix} -2\zeta & -1 \\ 1 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad f = \begin{bmatrix} 10 \\ 0 \end{bmatrix} \\ c^T &= [0 \ 1], \quad 0 < \zeta < 1.\end{aligned}\quad (9)$$

The output of the system defined by (1) and (9) has to become stable and aperiodic and to track the signal $r: [0, T] \rightarrow \mathbf{R}$,

$$r(t) = \begin{cases} r_1, & 0 \leq t < t_1 \\ r_2, & t_1 \leq t < t_2 \\ r_3, & t_2 \leq t \leq T \end{cases} \quad (10)$$

The matrices defined in (9) satisfy condition (6). The

matrices of the extended system (4) are

$$A_e = \begin{bmatrix} -2\zeta & -1 & 0 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}, b_e = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, f_e = \begin{bmatrix} 10 \\ 0 \\ 0 \end{bmatrix}, g_e = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad (11)$$

and the desired closed loop spectrum is

$$\Lambda_e = \{-1, -2, -10\}. \quad (12)$$

The resulting extended feedback matrix is, for $\zeta = 0.1$,

$$k_e^T = [k_o^T \mid \varphi] = [-12.8 \quad -31 \mid 20]. \quad (13)$$

In order to avoid possible Zeno behavior, in the presence of switching, a discrete time simulation model is defined by

$$x_d(+1) = A_d x_d + b_d u_d + f_d, \quad y_d = c_d^T x_d, \quad (14)$$

where, denoting $h > 0$ the sampling step and introducing the matrices in (9),

$$A_d = \exp(Ah), \quad b_d = \int_0^h \exp(A\theta) d\theta \cdot b, \quad (15)$$

$$f_d = f \cdot h, \quad c_d^T = c^T,$$

and the discrete variables are $x_d(k) = x(kh)$, $u_d(k) = u(kh)$, with $k = 0, 1, \dots$. Denote $A_{oe} = A_e + b_e k_e^T$. The simulation model of the closed loop system without observer is

$$\begin{aligned} x_{ed}(+1) &= A_{oe} x_{ed} + f_{ed} + g_{ed} r_d, \\ y_d &= c_{ed}^T x_{ed}, \end{aligned} \quad (16)$$

where,

$$\begin{aligned} A_{oed} &= \exp(A_{oe}h), \quad f_{ed} = f_e \cdot h, \\ g_{ed} &= g_e \cdot h, \quad c_{ed}^T = c_e^T, \end{aligned} \quad (17)$$

and, for any $k > 0$, the sampled signals are $x_{ed}(k) = x_e(kh)$, $r_d(k) = r(kh)$. With the simulation values

$$\begin{aligned} T = 30, \quad h = 0.001, \quad r_1 = 0, \quad r_2 = 2, \quad r_3 = 1, \quad t_1 = 5, \\ t_2 = 15, \quad \zeta = 0.1, \end{aligned} \quad (18)$$

the open loop evolution of the affine system (14) with the discrete input signal associated to (10) is depicted in Fig.2. The evolution of the closed loop system (16) with the same discrete input signal associated to (10) is depicted in Fig.3.

In order to build a minimal observer for the system (1)-(9), with eigenvalue $\lambda_{est} < 0$, rewrite the matrices (9) in the partitioned form

$$\begin{aligned} A &= \begin{bmatrix} A_1 & A_3 \\ A_2 & A_4 \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, \quad f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \\ c^T &= [0 \mid 1], \end{aligned} \quad (19)$$

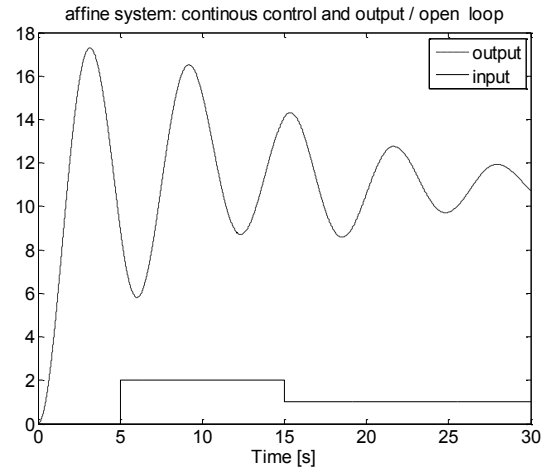


Fig.2. Simulation of the discrete affine system (14) with sampled piecewise constant input (10) and parameters (18).

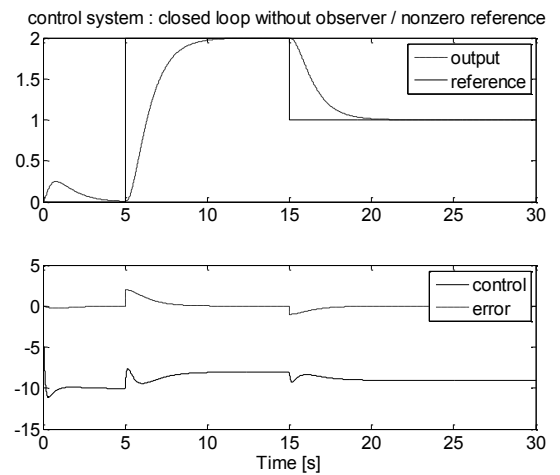


Fig.3. Simulation of the discrete control system (16) with sampled piecewise constant input (10) and parameters (18); the state is assumed to be known.

where $A_1 = -2\zeta = -0.2$, $b_1 = 1$, $f_1 = 10$ and the rest results by identification. The continuous time minimal observer is

$$\dot{w} = Jw + Hy + Mu + f_L, \quad \hat{x}_1 = w - Ly, \quad (20)$$

with

$$\begin{aligned} J &= A_1 + LA_2, \quad M = b_1 + Lb_2, \quad f_L = f_1 + Lf_2, \\ H &= -A_1L + A_3 - LA_2L + LA_4, \end{aligned} \quad (21)$$

and L chosen s.t. $\sigma(J) = \{\lambda_{est}\}$. The feedback law (7)-(13) is implemented, in continuous time, in the form

$$u = -12.8\hat{x}_1 - 31y + 20z, \quad (22)$$

and the simulation model comprises the discrete dynamics of the extended system (4)-(11) and of the observer (20), respectively. In the simulation experiment, the initial condition of the observer is disturbed $w(0) = 4 \neq x_1(0) = 0$.

The simulated evolutions for two distinct eigenvalues of the observer are comparatively presented in Fig.4, showing a deterioration of the performances, compared to Fig.3, for

slower observer dynamics (with time constant $T_{est} = 1/|\lambda_{est}|$).

III. AN APPROACH FOR A DRIVELINE OSCILLATIONS DAMPING PROBLEM – A SIMULATION EXAMPLE

The control approach presented in the previous section can be extended, with certain limitations, to affine systems with constrained control inputs, such as some automotive systems models.

A. The affine driveline model and the control problem

Consider the 5th order affine state-space model of the three inertias driveline model introduced in [8], defined as in (1). The state variables are

$$\begin{aligned} x_1 &= \theta_e - \theta_t i_t, \quad x_2 = \theta_t / i_f - \theta_w, \\ x_3 &= \omega_e, \quad x_4 = \omega_t, \quad x_5 = \omega_w, \end{aligned} \quad (23)$$

where ω_w , ω_t , and ω_e stand for the angular speeds of the wheel, of the transmission and of the engine, respectively, while θ_w , θ_t , and θ_e denote the corresponding angles, respectively; i_f is the final reduction gear ratio and i_t is the transmission ratio. The matrices of the affine model are

$$A = \begin{bmatrix} 0 & 0 & 1 & -i_t & 0 \\ 0 & 0 & 0 & \frac{1}{i_f} & -1 \\ \frac{-k_c}{J_1} & 0 & \frac{-d_c - d_e}{J_1} & \frac{d_c i_t}{J_1} & 0 \\ \frac{k_c i_t}{J_2} & \frac{-k_d}{i_f J_2} & \frac{d_c i_t}{J_2} & \frac{-d_{sum}}{J_2} & \frac{d_d}{i_f J_2} \\ 0 & \frac{k_d}{J_3} & 0 & \frac{d_d}{i_f J_3} & \frac{-d_w - d_d - c_{r2}}{J_3} \end{bmatrix}, \quad (24)$$

$$b = \begin{bmatrix} 0 & 0 & \frac{1}{J_1} & 0 & 0 \end{bmatrix}^T, \quad f = \begin{bmatrix} 0 & 0 & 0 & 0 & \frac{-T_{car}}{J_3} \end{bmatrix}^T,$$

where the parameters are defined in [8] and [9], with numerical values specified in Table 1 in [9]. In this paper, it is assumed that the control input of the system, represented by the engine torque $u = T_e$, is constrained by lower and upper limits, according to

$$0 \leq u(t) \leq T_{em}, \quad \forall t \in \mathbf{R}, \quad (25)$$

where $T_{em} = 200 \text{ Nm}$ is the maximum torque that can be generated. Also, it is assumed that the measured output is the wheel velocity

$$y = \omega_w r_w, \quad (26)$$

with r_w [m] the wheel radius. In a first approach, in order to analyze only the influence of the closed loop poles placement, it is assumed that no observer is used and hence all the state variables are available.

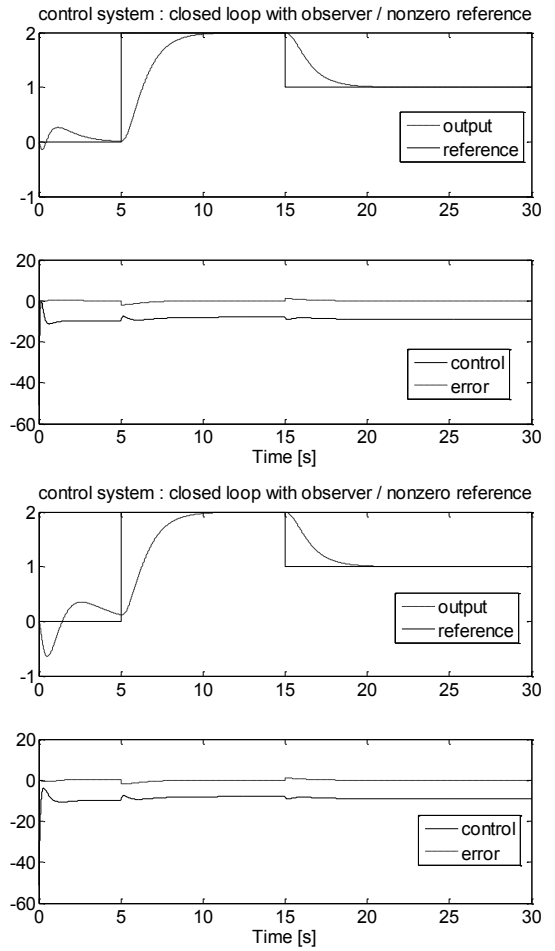


Fig.4. Simulation of the discrete dynamics associated to the extended system (11) with observer (20), input (10) and parameters (18); the pole of the minimal observer is $\lambda_{est} = -10$ (up) and $\lambda_{est} = -1$ (down).

Denote $\Lambda(A) = \{\lambda_i\}_{i=1}^5$ the spectrum of the system matrix A in (24), with the eigenvalues

$$\begin{aligned} \lambda_1 &= -2300, \quad \lambda_2 = -273,5, \\ \lambda_{3,4} &= -1,7 \pm 14,1j, \quad \lambda_5 = -0,2. \end{aligned} \quad (27)$$

In order to define the control objective, consider firstly a simulation of the system behavior to a step engine torque control signal, computed for a desired vehicle velocity $v_d = 30 \text{ km/h}$. The results are depicted in Fig.5 and the simulation model implements the discrete time version of the affine continuous time model (1) with matrices (24), as specified in previous section, with a sampling step $h = 0.005 \text{ s}$. The quality of the system behavior is defined by two main parameters:

1. the *vehicle velocity* response, $y = \omega_w r_w$ depicted in Fig.5 top right, and
2. the *driveline oscillations* resulted from the evaluation of the difference between the engine speed, divided by the total transmission ratio ω_e / i_{tot} , with $i_{tot} = i_f i_t$ - and the wheel speed ω_w , depicted in Fig.5 bottom left.

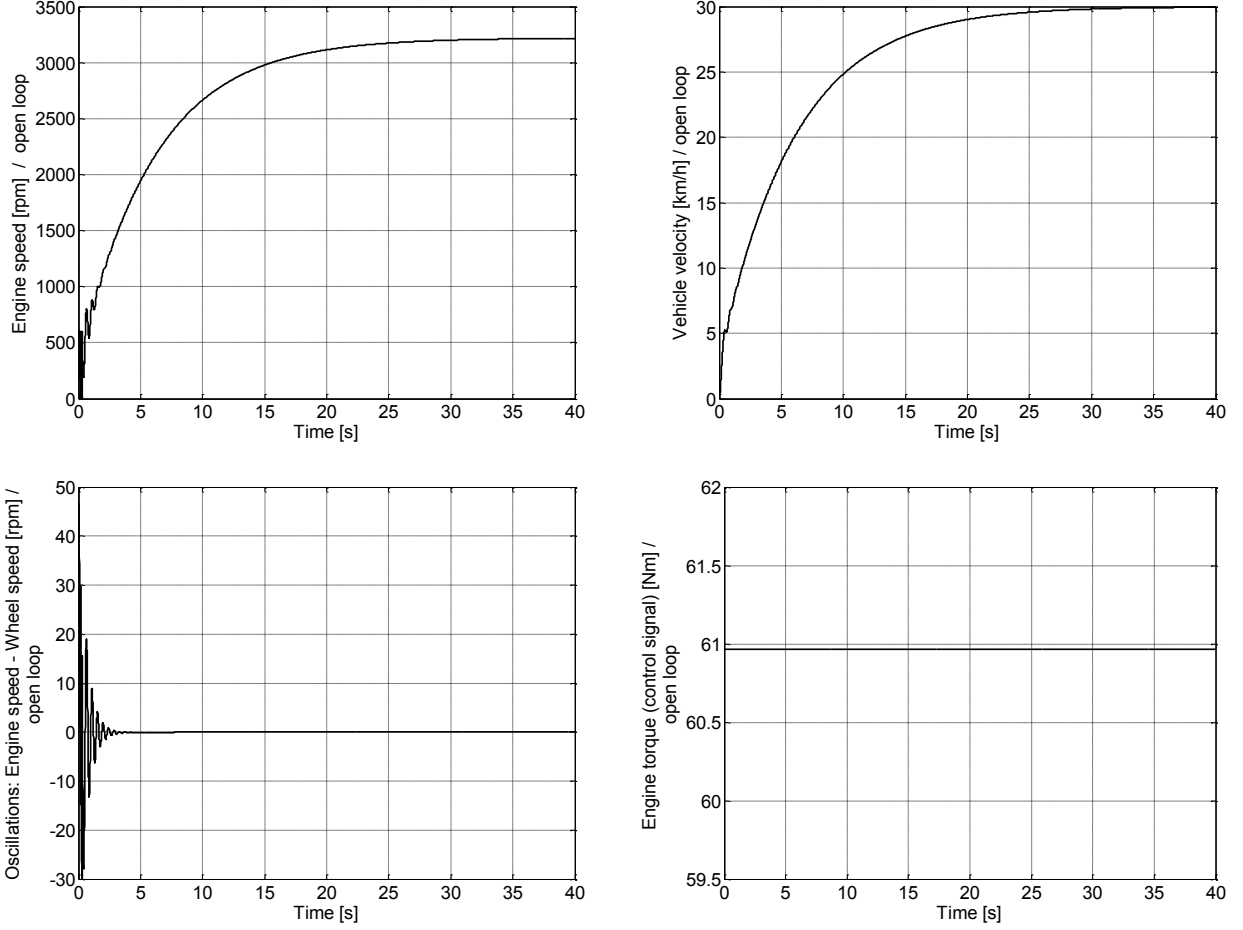


Fig. 5. Simulation results in open loop for the driveline affine model (24): engine speed (top left), wheel speed (top right), driveline oscillations (bottom left) and torque control.

The vehicle velocity response has an adequate shape, without overshoot, but with a too long time response, and the driveline oscillations, generated during the transitory time, have too high amplitude and frequency. Consequently, the *control objective* is to ensure a better response time, with minimum overshoot, while damping the driveline oscillations [8].

B. The control scheme and simulation cases

Consider the control scheme depicted in Fig.6, which extends the scheme in Fig.1 by incorporating a saturation function $Sat(\cdot)$, implementing the torque constraint (25).

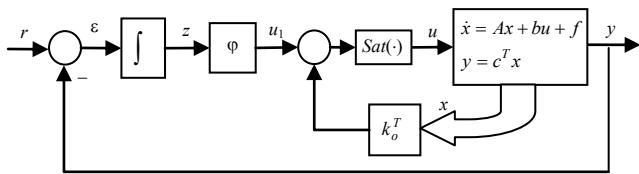


Fig.6. The control structure for the affine model of the driveline in [8]: u is the saturated engine torque, as control input, and r is the vehicle velocity reference.

Consider the closed loop extended system matrix $A_{oe} = A_e + b_e k_e^T$, where A_e and b_e are defined as in (5) with the matrices specified in (24), while the feedback matrix k_e has to be computed.

After examining the open loop simulation results in Fig.5 and the poles locations (27) of the affine driveline model specified in (1) and (24) and (26), the proposed *pole assignment philosophy* for the matrix A_{oe} , illustrated in Fig.7, is characterized as follows:

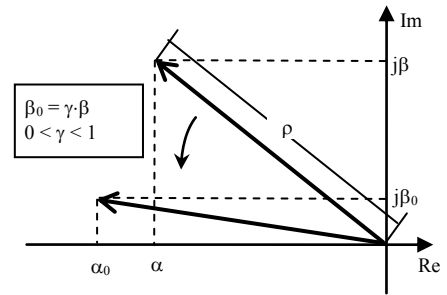


Fig.7. The pole assignment strategy for the pair of complex eigenvalues of the affine process matrix.

1. The eigenvalues $\lambda_1 = -2300, \lambda_2 = -273.5$ will be leaved unchanged, as they are real and far from the imaginary axis.
2. The complex eigenvalues $\lambda_{3,4} = -1.7 \pm 14.1j$ will be replaced by a rotated pair of complex poles, $\lambda_{30,40} = \alpha_0 \pm \beta_0 j$, where $\beta_0 = \gamma \cdot \beta$, $0 < \gamma < 1$, $\beta = \text{Im}(\lambda_3) = |\text{Im}(\lambda_4)|$ and with module preservation (Fig.7). This reduces the contribution to the oscillating behavior and drives to faster response time, without forcing the torque.
3. The eigenvalue $\lambda_5 = -0.2$, residing at the minimal distance from the imaginary axis, will be replaced by a real value $\lambda_{50} = \delta \lambda_5$, with $\delta \geq 10$, for a faster response time.
4. The last eigenvalue λ_{06} , associated to the controller state variable z in the extended system (5)-(6) is chosen real and stable, satisfying $0.5\lambda_{50} \leq \lambda_{06} < \lambda_{50}$.

With this general strategy, two pole assignment cases for the extended system (4)-(5) are discussed, which differ only in the choice of the last two eigenvalues, as specified at the points 3 and 4 above.

Case 1. The desired closed loop spectrum for the extended system matrix $A_{oe} = A_e + b_e k_{e1}^T$ is $\Lambda_1(A_{oe}) = \{\lambda_{i0}\}_{i=1}^6$, with

$$\begin{aligned} \lambda_{10} &= \lambda_1, \lambda_{20} = \lambda_2, \lambda_{30,40} = -14.1 \pm (0.1 \cdot 14)j, \\ \lambda_{05} &= -4, \lambda_{06} = -2, \end{aligned} \quad (28)$$

and the resulting feedback matrix k_{e1}^T , computed with the MATLAB routine `place` - which implements a robust pole assignment algorithm [10] -, has the components:

$$\begin{aligned} k_{e1}^T(1) &= -19.673, k_{e1}^T(2) = -356.6111, \\ k_{e1}^T(3) &= -5.2193, k_{e1}^T(4) = -0.1378, \\ k_{e1}^T(5) &= -23.4329, k_{e1}^T(6) = 106.8595 \end{aligned} \quad (29)$$

Case 2. The desired closed loop spectrum for the extended system matrix $A_{oe} = A_e + b_e k_{e2}^T$ is $\Lambda_2(A_{oe}) = \{\tilde{\lambda}_{i0}\}_{i=1}^6$, with $\tilde{\lambda}_{i0} = \lambda_{i0}$, $i = 1:4$, and

$$\tilde{\lambda}_{05} = -2, \tilde{\lambda}_{06} = -1. \quad (30)$$

The resulting feedback matrix k_{e2}^T has the components:

$$\begin{aligned} k_{e2}^T(1) &= -5.5883, k_{e2}^T(2) = -176.9881, \\ k_{e2}^T(3) &= -4.7093, k_{e2}^T(4) = -0.1238, \\ k_{e2}^T(5) &= 20.1909, k_{e2}^T(6) = 26.7149. \end{aligned} \quad (31)$$

The pole assignments in Case 1 (28), and in Case 2 (30) are illustrated in Fig.8.

In [8], there is also an engine torque rate constraint

$$-T_e^m \leq \dot{u}(t) \leq T_e^M, \quad \forall t \in \mathbf{R}, \quad (32)$$

which generates, for the discrete time affine simulation model, the constraint

$$-\Delta T_e \leq u_d(k) - u_d(k-1) \leq T_e^M, \quad \forall k \in \mathbf{Z}, \quad (33)$$

with $u_d(k) = u(kh)$ and $\Delta T_e = 6 \text{ Nm}$ [9].

The closed loop simulation experiment with torque constraints (25) and (32)-(33) are illustrated in Fig.9. In Case 1, the last two eigenvalues $\lambda_{05} = -4, \lambda_{06} = -2$ are placed on the negative real axis *left from* the corresponding eigenvalues (30) in Case 2 - compare Fig.8 top right with Fig.8 bottom right. Consequently:

- the vehicle velocity reaches the desired reference $v_d = 30 \text{ km/h}$ in Case 1 faster than in Case 2 (Fig.9 top right), but there is an overshoot in Case 1, and no overshoot in Case 2;
- the gain introduced by the feedback matrix in Case 1 is higher, on the entry 6, than in Case 2, which drives to a higher engine torque value, thus violating the constraint (25) (Fig.9 bottom right); consequently, the saturation of the engine torque is reflected in a worse behavior of the driveline oscillations in Case 1 compared to Case 2 (Fig.9 bottom left).

Summing up, a choice of the last two eigenvalues of the extended control system “far” from the origin, on the negative real axis, drives to a better time response of the vehicle velocity, but with the price of (a) an overshoot in the response shape and of (b) more intense driveline oscillations, as an effect of engine torque limitation (22).

IV. CONCLUSION

The oscillations damping is a control objective frequently imposed to automotive systems. For affine unconstrained systems with open loop oscillating behavior and controllable states, a control approach based on a pole assignment procedure combined with the internal model principle, implemented via a state observer, ensures oscillations damping, while tracking, with zero steady state error, a corresponding exogenous signal.

More accurate affine models of real systems, like the three inertias driveline model introduced in [8], [9], imply a set of limitations imposed to the system variables. This paper has illustrated, through a simulation example, the possibility to adapt the control approach for unconstrained affine systems to the specific engine torque constraints (25) and (32)-(33) in the driveline model specified in [8]. In order to detail the influence of the closed loop pole placement, in a first approach, the observer is ignored in the implementation of the feedback control, which is directly applied to the process state vector.

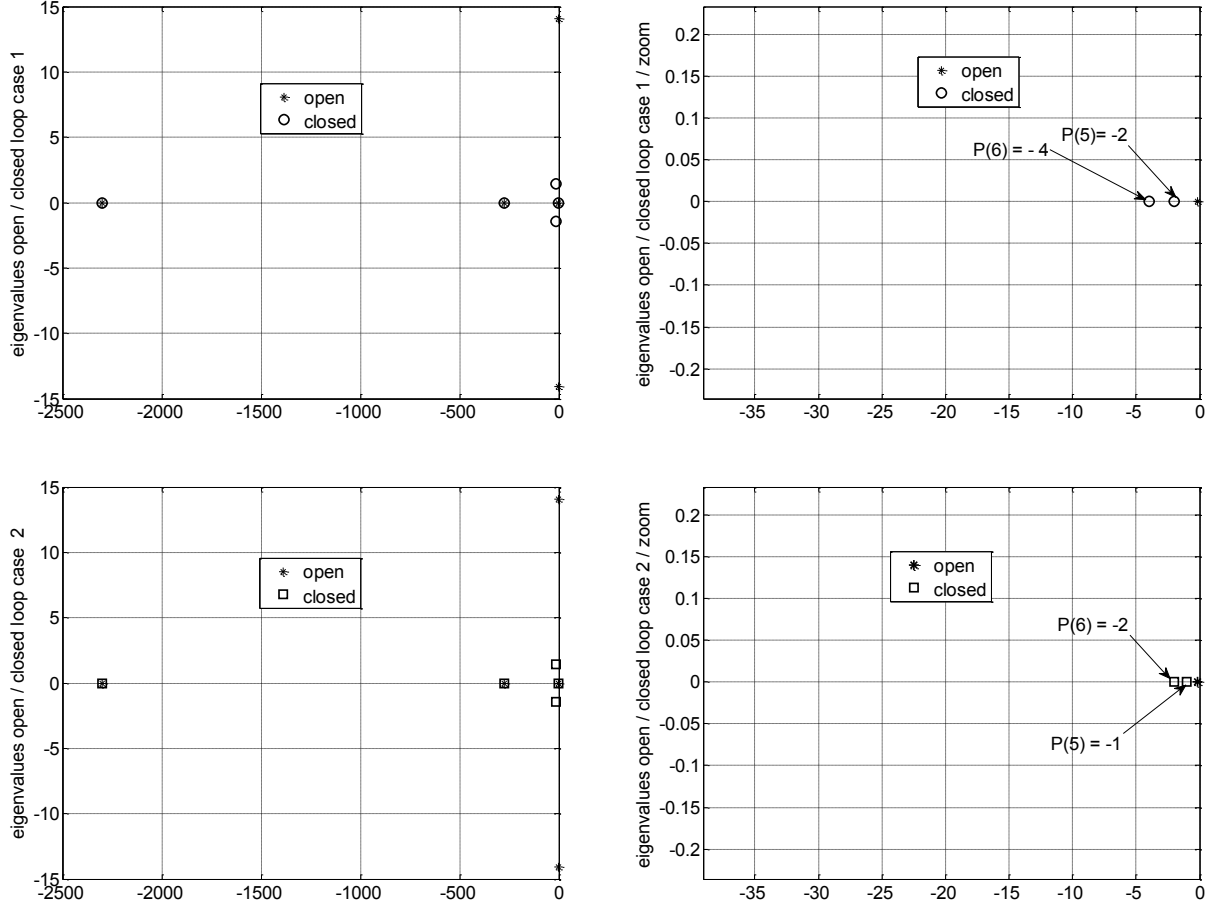


Fig. 8. Locations of the poles of the driveline affine model in open loop (24) and of the extended affine control system for the two choices of the poles assignment (28) and (30).

A general qualitative pole assignment philosophy is proposed and the simulation experiments show that a choice of the closed loop eigenvalues “too far” from the origin may drive to unacceptable overshoot in the vehicle velocity response, together with a worse driveline oscillating behavior. An acceptable closed loop behavior, with a response time faster than in the open loop case, without overshoot of the vehicle velocity and with driveline oscillations damping is obtained when placing the last two eigenvalues of the extended closed loop system at a distance from 5 up to 10 times larger than the length from the origin to the closest open loop eigenvalue; this ensures a balance between performance and engine torque magnitude. In this study, this situation occurs in Case 2 and the open loop eigenvalue close to the origin is $\lambda_5 = -0.2$ (see (27)).

Also, it is interesting to note that the shape of the response curves for the simulation experiments in Case 1 are similar to the ones obtained in [8], for the MPC scheme applied to the affine driveline model.

A first future research direction concerns the study of the influence of the observer dynamics on the controlled

driveline behavior, as illustrated in the unconstrained case in Section 2. A second relevant research topics is the extension of the discussed control scheme to piecewise affine models with constraints, within a hybrid control approach.

ACKNOWLEDGMENT

The authors are grateful to Prof. C. Lazăr and to PhD F. Căruntu, for helpful discussions and suggestions concerning the driveline affine and piecewise affine models as well as to the reviewers, for their remarks.

REFERENCES

- [1] W.P.M.H. Heemels, B. De Schutter, and A. Bemporad, “Equivalence of hybrid dynamical systems”, *Automatica*, Vol. 37, 2001, pp. 1085-1091.
- [2] P. Falcone, F. Borrelli, J. Pekar, and E. Greg Stewart, “Reference governor for constrained piecewise affine systems. A vehicle dynamics control application”, *Proc. of the European Control Conference (ECC)*, 23-26 August 2009, Budapest, Hungary, Chalmers Publication Library.
- [3] A. Azuma, and J. Imura, “Optimal control of sampled-data piecewise affine systems and its application to CPU processing control”, *Proc.*

of 42nd IEEE Conference on Decision and Control (CDC03), Maui (Hawaii), Dec. 9-12, 2003, pp. 161-166.

- [4] R. Ghosh, and Claire Tomlin, "Symbolic reachable set computation of piecewise affine hybrid automata and its application to biological modeling: Delta-notch protein signaling", *Systems Biology*, Vol.1, No.1, 2004, pp 170-183.
- [5] H. de Jong, J. Geiselman, C. Hernandez, and M. Page, "Hybrid modeling and simulation of genetic regulatory networks: a qualitative approach" in *Hybrid Systems Computation, HSCC2003, LNCS 2623*, 2003, pp. 267-282.
- [6] A. Balluchi, L. Benvenuti, M.D. di Benedetto, C. Pinello, and A. Sangiovanni-Vincentelli, "Automotive engine control and hybrid systems: Challenges and opportunities", in *Proceedings of the IEEE*, Vol. 88, 2000, pp 888-912.
- [7] Virginia Ecaterina Oltean, R. Dobrescu, and D. Popescu, "On oscillations damping and control for a class of affine systems"

(submitted for publication in the Proceedings of the 11th WSEAS International Conference on Systems Theory and Scientific Computation, Florence, Italy, 23-25 August, 2011).

- [8] Andreea E. Bălău, C.F. Căruntu, and C. Lazăr, "Driveline oscillations modeling and control", in *2011 Proc. of 18th International Conference on Control Systems and Computer Science*, Bucharest, Romania, pp. 332-338.
- [9] C. F. Căruntu, Andreea. E. Bălău, M. Lazăr, P. P. J. van den Bosch, and S. Di Caraiano, "A predictive control solution for driveline oscillations damping," in *2011 Proc. of Hybrid Systems: Computation and Control (HSCC 2011)*, Chicago, USA, pp. 181-190.
- [10] J. Kautsky, and N.K. Nichols, "Robust pole assignment in linear state feedback," *Int. J. Control*, 41, 1985, pp. 1129-1155.

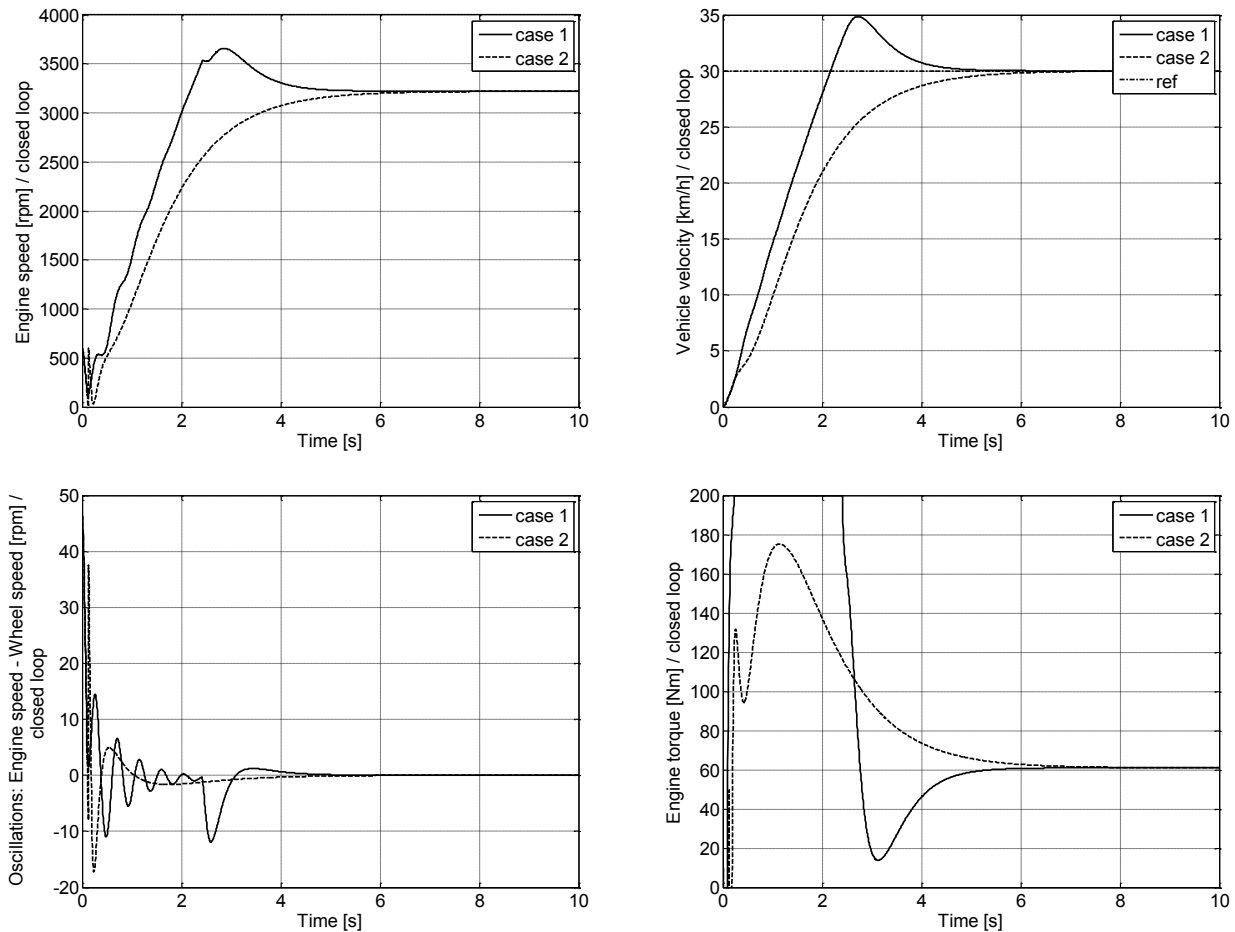


Fig. 9. Simulated closed loop evolution of the affine driveline model with engine torque constraints (25) and (32)-(33): engine speed (top left), vehicle velocity (top right), driveline oscillations (bottom left) and engine torque as control signal.

On Robotic Emotional Behavior

Andreea – Raluca Olteanu, Ionel - Cristinel Simion , Nicu – George Bizdoaca, *Member, IEEE*

Abstract— Research in the area of computational approaches to emotions has received increased attention over the last years. The purpose of this paper is to investigate emotion from a computational perspective, in order to improve human – robot interaction. The paper presents an experimental approach regarding animatronic design in conjunction with realistic animated motions.

I. INTRODUCTION

In recent years, emotion has increasingly been used in interface and robot design, primarily because of the collective acknowledgement, people tend to treat computers as they treat other people. Moreover, many studies have been performed to integrate emotions into products including toys, electronic games and software agents [22]. However, for robotic field research, emotions are meaningless unless they result in some outward change in the robot, including facial, vocal, or behavioral modifications [1].

Artificial emotions are used in social robots because they facilitate believable human–robot interaction, they can also provide feedback to the user, such as indicating the robot’s internal state, goals and intentions. Emotions can act as a control mechanism, driving behavior and reflecting how the robot is affected by, and adapts to, different factors over time [22].

Emotion can be used to coordinate planning, to determine control precedence between different behavior modes and to trigger learning and adaptation, particularly when the environment is unknown or difficult to predict. One approach is to use computational models of emotions that mimic animal survival instincts, such as look for food, escape from danger, etc. [15]. Several researchers have investigated the use of emotion in human–robot interaction. Suzuki [24] describe an architecture in which interaction leads to changes in the robot’s emotional state and modifications in its actions. Breazeal [15] presents how emotions influence the operation of Kismet’s motivational system and how this affects its interaction with humans.

Andreea – Raluca Olteanu PhD Candidate Automation Department, Faculty of Automation, Computers and Electronic, Craiova, Romania; (e-mail: a_raluk@yahoo.com).

Ionel - Cristinel Simion, Master Student, Mechatronics Department, Faculty of Automation, Computers and Electronic, Craiova, Romania; (e-mail: cristisimion84@gmail.com).

Nicu – George Bizdoaca is Professor, PhD Adviser at Mechatronics Department, Faculty of Automation, Computers and Electronic, Craiova, Romania; (e-mail: nicu@robotics.ucv.ro).

Nourbakhsh [25] discusses how mood changes can trigger different behavior in Sage, a museum tour robot [22].

This paper is organized as follows: Section II is dedicated to fundamental issues concerning emotions, Section III briefly presents social robots; Section IV presents our experimental study and, finally, Section V concludes the paper.

II. FUNDAMENTAL ISSUES CONCERNING EMOTIONS

A. Towards a generic definition of “emotion”

Emotions play a major role in human interaction. Quite often, emotional reactions are caused by social interactions, influenced by societal and cultural norms, or used to communicate desires to other people. Emotions carry conversational content, allowing conversational partners to form common ground and communicate more effectively [1].

The ample scope of affective phenomena opens up a definition space that can be explored through many different perspectives, and results in varied descriptions at multiple levels of abstraction. Thus it is difficult to reach consensus in the characterization of affect and emotion [2]. Kleinginna [3] considers 92 different definitions given by researchers, organized to different categories, ranging from their relation to physiological components or emotional behaviors, to definitions based upon motivational and adaptive views, together with 9 skeptical remarks, regarding that emotion is not such a useful concept.

In 1984, Scherrer defines emotions as feeling states, and measure these states by asking the subjects about the “level” of emotions they are experiencing [4], while Plutchick in 1962 define emotion as “a patterned bodily reaction of either destruction, reproduction, incorporation, orientation, protection, reintegration, rejection or exploration, or some combination of these, which is brought by a stimulus.”[5]. Scherer [6], considers the following phenomena important in developing an affective model: emotions, moods and attitudes.

B. Modeling Emotion

Recent years have seen a significant expansion in research on computational models of human emotional processes, driven both by their potential for basic research on emotion and cognition as well as their promise for an ever increasing range of applications.

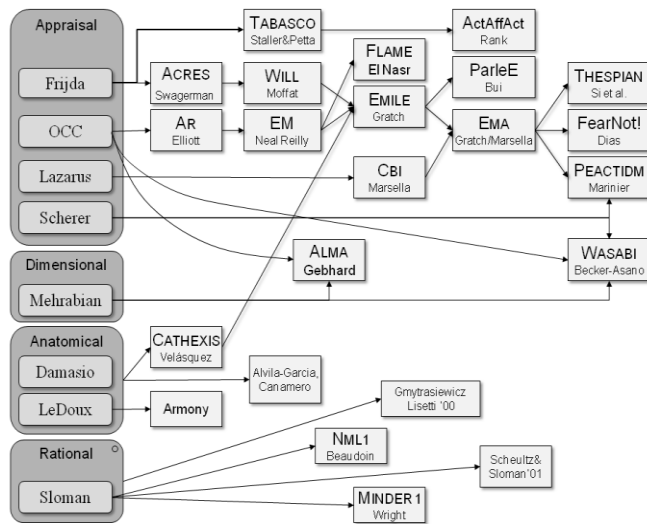


Fig. 1. Several computational models of emotion [7]

Architectures are being built both to elucidate mechanisms of emotions and to enhance believability and effectiveness of synthetic agents and robots (see Figure 1).

This has led to a truly interdisciplinary and mutually beneficial partnership between emotion research in psychology and computational science [7].

Key components of emotional processing are the mechanisms that evaluate a current internal and external situation, in terms of an affective state or emotion. This process is referred to as *appraisal*, (also *cognitive appraisal*). Appraisal theory develops a catalogue of human emotions and seeks to provide a computational account of the ‘appraisals’ that lead one emotion to be evoked over another, but without the ‘heat’ provided by emotion’s biological underpinnings in humans. Appraisal theory is therefore a good candidate for algorithms that enable a robot to simulate the appearance of human emotional behavior. However, a good theory of robot emotions must address the fact that many robots will have no human–computer interface, thus a more abstract view of emotion is required [21].

Dimensional theories of emotion argue that emotion and other affective phenomena should be conceptualized as points in a continuous dimensional space, while anatomic theories stem from an attempt to reconstruct the neural links and processes that underlie organisms’ emotional reactions. Rational approaches start from the question of what adaptive function does emotion serve and then attempt to abstract this function away from its “implementation details” in humans and incorporate it into a model of intelligence [7].

C. Basic Elements of Emotional Phenomena

Searching for irreducible elements of emotion, the most important accounts correspond to those that view the range of emotional phenomena as a set of discrete emotions, and those which take on the perspective that emotions can be further reduced into specific dimensions [2].

Many researchers argue that any given emotion is reducible to a small number of underlying dimensions.

Earlier approaches take a two-dimensional view of emotion in which the two dimensions correspond to *valence* (how positive or negative the emotion is) and *arousal* (the energy or excitation level associated with the emotion). [2]. Thus, an emotion is considered to be either *pleasant* or *unpleasant* (also labeled as *positive* or *negative*), either *activated* or *deactivated* (also labeled as *aroused* or *sleepy*, and *engaged* or *disengaged*) [7].

In Figure 2 the Circumplex Model of Affect proposed by Russell [7] was represented, which is one of the major models of emotion. The Circumplex Model of Affect is a spatial model based on dimensions of affect and this affective concepts fall in a circle in the following order: pleasure (0°), excitement (45°), arousal (90°), distress (135°), displeasure (180°), depression (225°), sleepiness (270°), and relaxation (315°) (see Figure 2). According to this model, there are two components of affect that exist: first, *pleasure-displeasure*, the horizontal dimension of the model, and second *arousal-sleep*, the vertical dimension of the model. Therefore, it seems that any affect word can be defined in terms of its pleasure and arousal components. The remaining four variables mentioned above do not act as dimensions, but rather help to define the quadrants of the affective space [27]. It is obvious that this kind of classification is based upon self-reports and introspection and such a model is deeply subjective.

Modeling the emotion computationally implies thinking on a much lower level of abstraction, because at some point it is necessary to be able to relate these emotions and their localization in this two-dimensional space in terms of some measurable feature of the body, or reveal something about the kind of behavior that these emotional experiences result in, such as approach or avoidance [2].

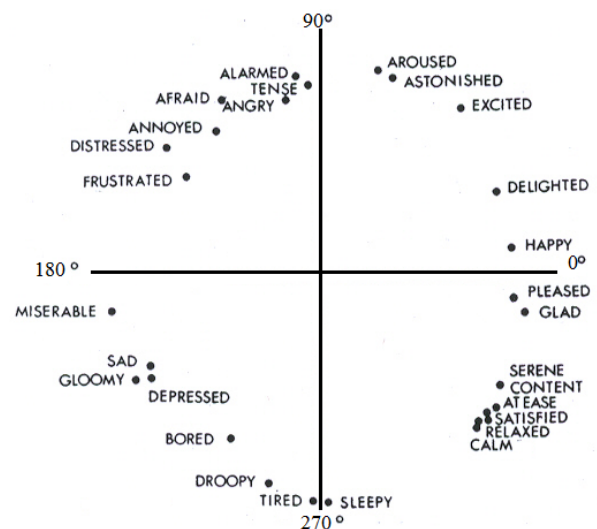


Fig. 2. A circumplex model of affect [7]

Each emotion is characterized by physiological and behavioral qualities, including movement, posture, voice, facial expression, and pulse rate fluctuation.

TABLE I
A SELECTION OF LISTS OF “BASIC” EMOTIONS [10]

Reference	Fundamental emotion	Basis for inclusion
Arnold (1960)	<i>Anger, aversion, courage, dejection, desire, despair, fear, hate, hope, love, sadness</i>	Relation to action tendencies
Ekman, Friesen and Ellsworth (1982)	<i>Anger, disgust, fear, joy, sadness, surprise</i>	Universal facial expressions
Frijda (1986)	<i>Desire, happiness, interest, surprise, wonder, sorrow</i>	Forms of action readiness
Gray (1982)	<i>Rage and terror, anxiety, joy</i>	Hardwired
Izard (1971)	<i>Anger, contempt, disgust, distress, fear, guilt, interest, joy, shame, surprise</i>	Hardwired
James (1884)	<i>Fear, grief, love, rage</i>	Bodily involvement
Mowrer (1960)	<i>Pain, pleasure</i>	Unlearned emotional states
Oatley and Johnson-Laird (1987)	<i>Anger, disgust, anxiety, happiness, sadness</i>	Do not require propositional content
Panksepp (1982)	<i>Expectancy, fear, rage, panic</i>	Hardwired
Plutchik (1994)	<i>Acceptance, anger, anticipation, disgust, joy, fear, sadness, surprise</i>	Relation to adaptive biological processes
Tomkins (1984)	<i>Anger, interest, contempt, disgust, distress, fear, joy, shame, surprise</i>	Density of neural firing
Watson (1930)	<i>Fear, love, rage</i>	Hardwired
Weiner & Graham (1984)	<i>Happiness, sadness</i>	Attribution independent

Ekman [8] and others argue for a set of “basic” emotions that are innate and universal across cultures: *anger, disgust, fear, joy, sadness, surprise*. All other emotional categories are then built up from combinations of these basic emotions [1].

There are three main ways in which the notion of *basic emotions* has been used in the literature [2]. Ekman [18] suggests that there are a number of discrete emotions that differ one from another in important ways. For instance, fear, anger, and joy differ in their eliciting conditions, as well as in their usually associated behavioral and physiological characteristics.

Plutchick [19] argues that these emotions are fundamental and sufficient elements to describe all emotional phenomena.

The term basic applies to them in the sense that they are descriptive of the most common emotional phenomena, and when combined, they can produce other more complex emotions [2].

Izard [20] suggests that these discrete emotions have a biological basis, and they are basic for the organism. They evolved due to their adaptive value in helping organisms deal with recurrent, fundamental life- and survival related tasks. Thus, the characteristics shared by these emotions are largely biologically determined [2].

Ekman’s model [8], with its six basic emotional states, has been influential in work on emotional expression in robots. The Kismet robot [15], for example, can communicate an emotive state and other social cues to a human through facial expressions, gaze, and the quality of the voice. The computations needed to communicate an ‘emotional state’ to a human might also improve the way robots function in the human environment [21].

To achieve a translation from emotions into facial expressions, emotions need to be parameterized. In the robot Kismet [15], facial expressions are generated using an interpolation-based technique over a three-dimensional, componential affect space (arousal, valence, and stance). Probo [26] uses two dimensions (valence and arousal) to construct an emotion space, based on the circumplex model of affect defined by Russell [7]. In the emotion space a Cartesian coordinate system is used, where the x coordinate represents the valence and the y -coordinate the arousal, consequently each emotion $e(v, a)$ corresponds to a point in the valence-arousal plane (Figure 3). This way, the basic emotions can be specified on a unit circle, placing the neutral emotion $e(0, 0)$ in the origin of the coordinate system. Thus, each emotion can also be represented as a vector with the origin of the coordinate system as initial point and the corresponding valence-arousal values as the terminal point.

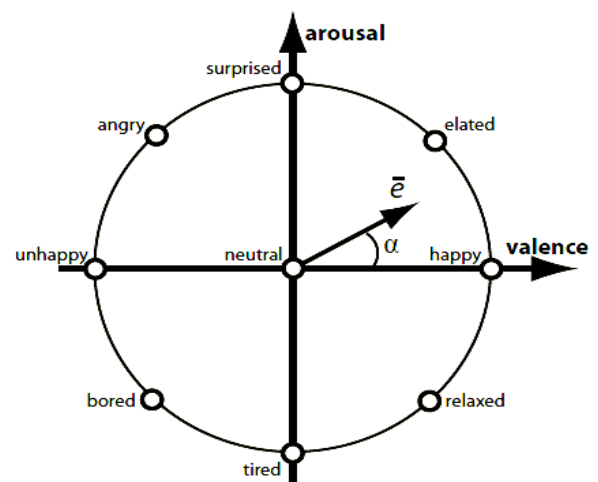


Fig. 3. Probo emotion space [26]

The direction α of each vector defines the specific emotion, whereas the magnitude defines the intensity of the emotion. The intensity i can vary from 0 to 1, interpolating the existing emotion $i = 1$ with the neutral emotion $i = 0$. Each DOF that influences the facial expression is related to the current angle α of the emotion vector.

III. SOCIAL ROBOTS

In recent years, the robotics community has seen a gradual increase in social robots, that is, robots that exist primarily to interact with people [1]. Robots are used to entertain humans with an active and affective communication expressing emotions through various communication channels. The robot AIBO [11] is used as an amusement toy, PARO [13] is a seal-like robot developed to help the healing process of infants in a hospital environment, the Aurora project [14] created a robot called Robota that deals with children who have psychological or physical learning and communication difficulties and facilitates the emotional development of children by engaging them in interactive games that encourage expressing their affective states while playing with the robot. Another robot developed by NEC called PaPeRo [12] or robots with infant-like abilities of interaction, such as Kismet [15], have been used to demonstrate the ability of people to interpret and react appropriately to a robot's displays of emotions. Museum tour-guide robots [16] and robots that interact with the elderly [17] demonstrate not only the benefits of having robots interact with people, but also the need for the interactions to be smooth and natural. Many of these robots have incorporated at least some rudimentary emotional behaviors, but such behaviors are usually ad-hoc and not extensible to other robots. [1]

Thus such social robots could be found in education, in virtual learning situations, in advisory services where it is planned to make information systems more accessible to users when embodied conversational agents to assist them in dealing with specific situations. The area of entertainment is also an important area of application for such systems, as can be seen in the example of various computer games based on avatar technology as well as the success of small robots, which are used as children's toys.

In the framework of the interdisciplinary European research project "Humaine" the key abilities of an emotional embodied conversational agent are defined as follows: first, the ability to co-ordinate different signs such as gestures, facial expression, posture and language; second, articulateness and expressiveness, and third, the generation of affectivity and attentiveness in the communication process.

All robot systems, whether socially interactive or not, must solve a number of common design problems. These include *cognition* (planning, decision making), *perception* (navigation, environment sensing), *action* (mobility, manipulation), *human-robot interaction* (user interface,

input devices, feedback display) and *architecture* (control, electromechanical, system). Socially interactive robots, however, must also address those issues imposed by social interaction [22].

IV. EXPERIMENTAL STUDY

Wowwee Alive Chimpanzee [23] is an experimental platform for human-robot interaction. It has three touch sensors (one under his chin, one on the back and one on the top of his head), two vibration sensors and two microphones (both ears).

The Chimpanzee has four moods which determine his reactions to user interaction: *curious* (default mood), *happy*, *fearful* and *angry*. The Chimpanzee's emotions change as a direct result of how he is treated.

The Chimpanzee's Infrared Vision System enables him to detect movement in front of him. His vision is on only when he is not moving or making a sound. The Chimpanzee will track objects in front of his face: up, down, left and right. When tracking an object the Chimpanzee will keep his vision locked on it until he loses sight of it. This will not affect his mood. The Chimpanzee has a Stereo Sound Detection System which can detect sharp, loud sounds (like a clap) to his left, his right and directly ahead. He will only listen when he is not moving or making sound.

When he hears a sound, the Chimpanzee will respond with an animation. His hearing and vision is pretty limited to stimuli just 10-20 cm from his face. Responding to sounds will also influence his mood.

In the head there are eight motors (6V DC motor), which are moving the head and the eyes up and down, left and right, moving the mouth and eyelids open and close and the eyebrows and nose up and down.

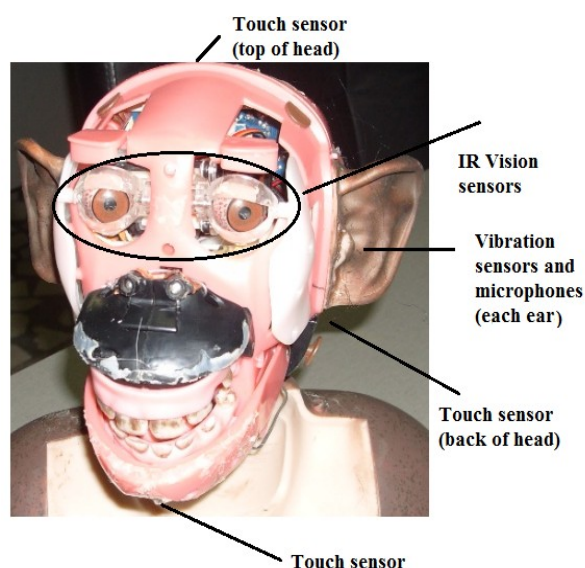


Fig. 4. The Chimpanzee Sensors

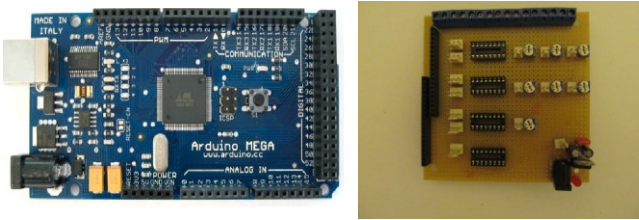


Fig. 5. The Arduino Mega 2560 Board

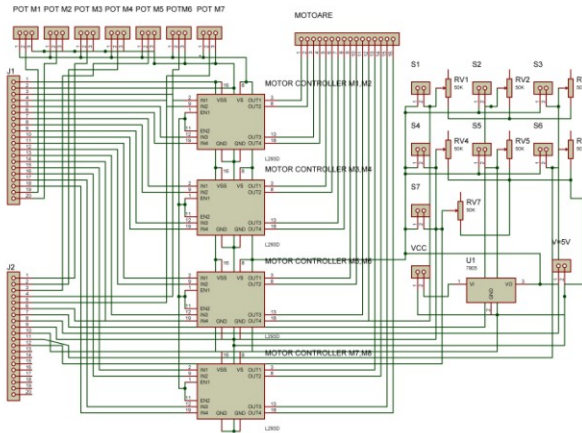


Fig. 6. Motor controls circuit and sensor interface

The original robot was controlled by the remote. For testing and performance enhancement, we have added a "parasitic" microcontroller to the Chimpanzee that allows external control of this device. Arduino Mega 2560 was chosen, a microcontroller board based on the ATmega2560 with a number of facilities for communicating with a computer or other microcontrollers.

V. CONCLUSION

In the last few years there has been an increased interest for research in the area of computational approaches to emotions.

The idea of our experimental project is the design and implementation of an emotional architecture which allows an easy integration of new drives, sensors, actors and behaviors. In the future, we intend to implement with the existing equipment a robot affective system that will produce the robot's emotional states in accordance with the internal and external stimuli such as interactions between the user and the Chimpanzee. Human – robot interaction will be enhanced by implementing a communication module which consists of a MP3 Trigger, thus giving the robot the ability to play MP3s and a Voice Recognition Module (VROBOT) that offers possibilities of recognizing 30 voice commands, customized by user.

ACKNOWLEDGMENT

This work was supported by CNCSIS 289, PNCDI-2008-2 IDEI.

REFERENCES

- [1] R. Gockley, R. Simmons, J. Forlizzi, "Modeling Affect in Socially Interactive Robots", *Proceedings of the 15th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN06)*, Hatfield, UK, September 6-8, 2006
- [2] J. D. Velasquez, "When Robots Weep: A Computational Approach to Affective Learning", *Doctoral Dissertation*, Massachusetts Institute of Technology, 2007
- [3] P. Kleinginna, A. Kleinginna, "A categorized list of emotion definitions, with suggestions for a consensual definition", *Motivation and Emotion* 5, pp 345- 379, 1981
- [4] K. R. Scherer, "On the nature and function of emotion: A component process approach", in K. R. Scherer & P. Ekman, eds, 'Approaches to emotion', Lawrence Erlbaum, Hillsdale, NJ, pp. 293-317, 1984.
- [5] R. Plutchick, "The emotions: Facts, theories, and a new model." New York: Random House, 1962.
- [6] K. R. Scherer, "Psychological models of emotion," in *The Neuropsychology of Emotion*, J. C. Borod, Ed. Oxford: Oxford University Press, 2000, pp. 137-162.
- [7] S. Marsella, J. Gratch, and P. Petta, "Computational Models of Emotion." In Scherer, K.R., Bänziger, T., & Roesch, E. (eds.) *A blueprint for an affectively competent agent: Cross-fertilization between Emotion Psychology, Affective Neuroscience, and Affective Computing*. Oxford University Press
- [8] P. Ekman, Ed., *Emotion in the Human Face*, 2nd ed. Cambridge, England: University of Cambridge Press, 1982.
- [9] J. Russell, "Core affect and the psychological construction of emotion," *Psychological Review*, vol. 110, no. 1, pp. 145-172, 2003.
- [10] A. Ortony, T.J. Turner, 'What's Basic About Basic Emotions?', *Psycho-logical Review* 97(3), 315-331, 1990.
- [11] AIBO. *Entertainment Robot*. <http://www.aibo.com/>, 2003
- [12] PaPeRo. *Child care robot*. <http://www.incx.nec.co.jp/robot/english/childcare>, 2002.
- [13] T. Shibata, T. Tashima, and T. K. "Emergence of emotional behavior through physical interaction between human and pet robot." *Proceedings of the International Workshop on Humanoid and Human Friendly Robotics*, 2003.
- [14] J. Nadel, C. Guerini, A. Peze, and C. Rivert. *The evolving nature of imitation as a format of communication*. Cambridge University Press, 1999.
- [15] C. Breazeal, "Emotion and sociable humanoid robots," *International Journal of Human Computer Studies*, vol. 59, pp. 119-155, 2003.
- [16] I. Nourbakhsh, J. Bobenage, S. Grange, R. Lutz, R. Meyer, and A. Soto, "An affective mobile robot educator with a full-time job," *Artificial Intelligence*, vol. 114, no. 1-2, pp. 95-124, 1999.
- [17] M. Montemerlo, J. Pineau, N. Roy, S. Thrun, and V. Verma, "Experiences with a mobile robotic guide for the elderly," in *Proceedings of the National Conference of Artificial Intelligence (AAAI 02)*, Edmonton, AB, July 2002, pp. 587-592.
- [18] P. Ekman, "Basic Emotions", in T. Dalgleish & M. Power, eds, 'Handbook of Cognition and Emotion', John Wiley & Sons, Inc, 1999.
- [19] R. Plutchik, "The Psychology and Biology of Emotion", HarperCollins, New York, 1994.
- [20] C. Izard, "Human Emotions", Plenum, New York, 1977
- [21] M.A. Arbib and J-M. Fellous, "Emotions: from brain to robot. Trends in cognitive sciences", 8:554-61, December 2004.
- [22] D.S. Kwon, "A Survey of Socially interactive robots", Cap 2. Methodology, GCT662, HRI, 2008
- [23] <http://www.wowwee.com/en/support/chimpanzee>
- [24] K. Suzuki, et al., Intelligent agent system for human-robot interaction through artificial emotion, in: *Proceedings of the IEEE SMC*, 1998.
- [25] I. Nourbakhsh, "An affective mobile robot educator with a full-time job", *Artificial Intelligence* 114 (1-2) (1999) 95-124.
- [26] J. Saldien, K. Goris, S. Yilmazyildiz, W. Verhelst, D. Lefeber, "On the design of the huggable robot Probo", *Journal of Physical Agents*,

- [27] J. Altarriba, D. Basnight, D. M. and T.M. Canary, "Emotion representation and perception across cultures". In W. J. Lonner, D. L. Dinnel, S. A. Hayes, & D. N. Sattler (Eds.), *Online readings in psychology and culture*. Bellingham, Washington: Western Washington University Center for Cross-Cultural Research.

On the Use of Collaborative Intelligence in an Agent-based Environmental Monitoring and Analysis System

Mihaela Oprea, Elia Dragomir, and Mădălina Cărbureanu

Abstract—The paper introduces the architecture of a multi-agent system for environmental monitoring and analysis, and focuses on the use of collaborative intelligence between the environmental agents (the agents that monitor and analyze the evolution of specific environment pollutants concentrations in different locations) and meteorological agents (that provide information about all meteorological parameters that can influence the degree of environmental pollution: temperature, speed of wind, humidity etc). The collaboration between agents is facilitated by asynchronous messaging and a common ontology (that is shared by all agents, environmental and meteorological). Some experimental results obtained for different scenarios of the multi-agent system simulation are also discussed in this paper.

I. INTRODUCTION

The application of agent-based technology to distributed problem solving from a variety of domains has the advantage of possible implementation of collaborative intelligence, which measures the collaborative ability of a group of agents that work together in order to achieve specific goals. Collaboration is based on three mechanisms of agent's interaction, communication (facilitated by the social knowledge), coordination and cooperation. A collaborative agent can cooperate and negotiate with other agents in order to achieve its goals and the system's goal.

In the last decade, several agent-based collaborative systems had been reported in literature, from different domains: business process management, e-learning, robotics, construction, planning and scheduling in manufacturing systems, computer-aided design etc. A brief review of the literature makes the evidence of the intensive research work already done in this direction. Most of the industrial collaborative systems are for the manufacturing domain (see e.g. [1], [2], [3], [4], [5]), for planning and scheduling, and for the product design. In [1] it is presented a multiagent negotiation based model for collaborative supply chain planning.

Collaborative networks are tackled in [2]. An architecture (named CAPPS) for scheduling intensive production management, based on collaborative agents is proposed in [3], for use in Japan. A new process planning method for assembly lines based on collaborative systems is described in [4]. In [5] it is presented an industrial case study in the development of a collaborative e-Engineering environment for mechanical

product design, based on intelligent software agents. An interesting real world application from the construction domain is described in [6], where it is presented a new approach based on collaborative agents to coordinate construction equipment operation in real time. A case study with two cranes working together to lift a panel on a bridge with height limits from the bridge structure is discussed in detail. Several robotic applications that use collaborative agents are presented in the literature. In [7] it is presented a macroscopic analytical model of agent collaboration in distributed robotic systems. The implementation of the collaboration ability in agent-based web applications (e.g. for B2B workflow monitoring or coalition formation in virtual enterprises) are tackled in several research papers (some examples are given in [8] and [9]).

The use of learning capability in collaborative agent-based systems is one of the techniques that increase the accuracy of systems' performance. A recent example is given in [10], where it is proposed the use of novel data mining hybrid techniques, combined with a biological method.

Apart from the learning capability, that can improve the performance of a collaborative agent-based system, the ontology development is another key issue needed for collaborative intelligence implementation. An example of using the ontology in such a system is given in [11], for collaborative design in educational environments.

From the short overview of some agent-based collaborative systems, briefly presented in this introduction, we can conclude that there are many research challenges in the development of such performing systems in different domains of expertise. In this paper, we shall focus on the use of collaborative intelligence in the environmental domain of application.

Agent-based technology can provide solutions proper to solve various environmental problems. Some research projects had reported in the literature the progress made in this direction, [12], [13], [14], [15].

The potential use of multi-agent systems in real time environmental monitoring, analysis, forecasting, and control raises new challenges for the researchers working in the area of distributed intelligent systems [16]. Among these, the implementation of distributed collaborative intelligence can improve the efficiency of agent-based environmental systems (for environmental networks) in the context of solving emergency situations in real time, such as, for example, the critical situation of a severe air pollution with an important impact on water and soil pollution in a given region, amplified by specific meteorological conditions.

Manuscript received May 30, 2011. (Write the date on which you submitted your paper for review.)

M. Oprea is with the University Petroleum-Gas of Ploiesti, Department of Informatics, Bdul Bucuresti 39, Ploiesti, 100680, Romania (corresponding author to provide phone: +40-255-575059; fax: +40-244-575847; e-mail: m_oprea@yahoo.com).

E. Dragomir is with the University Petroleum-Gas of Ploiesti, Department of Informatics, Bdul Bucuresti 39, Ploiesti, 100680, Romania (e-mail: elia.dragomir@yahoo.com).

M. Cărbureanu is with the University Petroleum-Gas of Ploiesti, Department of Informatics, Bdul Bucuresti 39, Ploiesti, 100680, Romania (e-mail: mcarbureanu@upg-ploiesti.ro).

In literature there is presented a large number of environmental ontologies, such as AIR_POLLUTION_Onto, an ontology for air pollution analysis and control [24], SWEET (Semantic Web for Earth and Environmental Terminology) ontology [25] and a large number of environmental MAS such as MAS_AirPollution [26] for air pollution analysis and control, etc.

In this paper it is presented the architecture of a multi-agent system for environmental quality monitoring and analysis, named EnvMAS, that was designed as a collaborative system, in which the agents work jointly to solve different environmental problems, such as air quality monitoring and analysis in urban regions, surface water quality monitoring and analysis, flood monitoring and forecasting in a specific hydrographic basin. The preliminary work presented here is focused on the air pollution monitoring.

The paper is organized as follows. Section II describes the problem of environmental monitoring and analysis and briefly presents the solution of using agent-based systems. The architecture of the EnvMAS system is introduced in section III. Some experimental results are discussed in section IV. The final section concludes the paper.

II. AGENT-BASED ENVIRONMENTAL MONITORING AND ANALYSIS

A. The problem of environmental monitoring and analysis

In the last years, several severe environmental problems occurred with a high direct impact on population and on the environment (such as floods, tsunamis, air pollution, water pollution, soil pollution, earthquakes, nuclear accidents, etc.).

The first solution that has to be adopted is to have a real time monitoring network, with stations in the most important locations. For example, in the case of air pollution, the network is composed by the monitoring stations placed in the sites where there are air pollution problems (e.g. sites with known air pollution sources such as chemical plants, power plants, heavy traffic, etc.).

An environmental monitoring network has to check continuously the environmental quality standards, and if the admissible levels are exceeded, to alert the environmental decision factors, and/or the population, and if possible, to reduce the impact on the environment and on the population by applying some real time control measures. Also, such networks will provide data that can be used in order to make forecasting about the evolution of different environmental quality parameters (e.g. forecasting the PM10 concentration (particulate matter), forecasting the flow of a river).

In conclusion, the problem of environmental monitoring and analysis can be formulated as follows: "Given a certain geographical area, develop a monitoring and analysis network that provides, continuously, the environment state."

This simplified formulation of the problem can be

extended with the environmental control problem (eventually, in real time). We have to note that the problem is very complex, due to incomplete and not accurate data, to the uncertainty (inherently, in environmental processes), and to other environment components, that can influence, directly or indirectly, the state of the environment.

B. Agent-based environmental systems

One of the best solutions that can be adopted for solving the problem of environmental monitoring and analysis is given by multi-agent systems. In such a case, the monitoring and analysis network is actually a multi-agent system, geographically distributed, in which each station has associated one or more agents that monitor and analyze the state of the environment in that location, and which inform the supervisor agents if some environmental problems can occur. Figure 1 shows the overview of such a system. The multi-agent system is composed by the agents of type A associated to each station, and a supervisor agent, S, that can ask the monitoring and analysis agents (A) to provide the local environment state. Each agent A can inform the supervisor agent, S, if a problem occurs.

As the environmental problems are very complex, and they need additional information from other parts of the environment (that have associated their specific monitoring

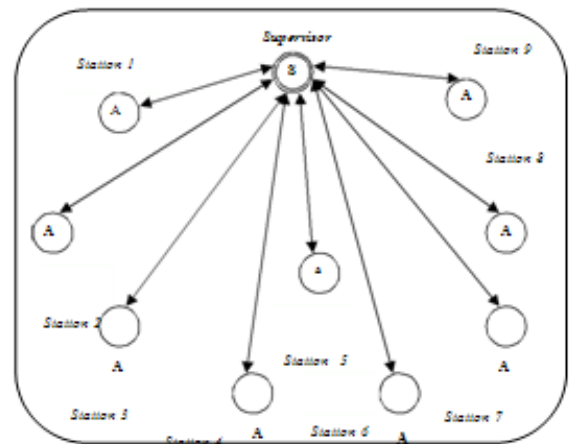


Fig.1. Overview of an agent-based environmental monitoring and analysis system.

and analysis network), the multi-agent system has to interact with the agents or their resources (e.g. databases) from other part of the environment multi-agent monitoring and analysis systems. This interaction can be solved by collaborative intelligence. Each multi-agent system has a certain global goal even if the part of the environment is the same. The goal depends on the specific problem that is tackled, which requires a specific monitoring network. For example, water can be monitored from the pollution viewpoint (requiring a water pollution monitoring network) or from the possible flood viewpoint (requiring a hydrometric monitoring network).

III. THE ARCHITECTURE OF THE EnvMAS SYSTEM

We propose the architecture of an agent-based collaborative system, EnvMAS, for environmental quality monitoring and analysis that is shown in Figure 2. The system can tackle three major environmental problems, air pollution, water pollution, and soil pollution. We have considered that each environment monitoring network has associated a multi-agent system, and there is a global environment supervisor system (e.g. at the level of the Emergency Situations Management System), that monitor and analyze the whole environment pollution state, and manage the environmental problems solving, by taking control measures.

Each multi-agent system that is a part of the EnvMAS system is composed by the following types of agents: environmental agents (located at specific monitoring stations of the environment monitoring network), and supervisor agents. An environmental agent (A) monitors the evolution of a set of environment parameters (e.g. air pollutants concentrations), and makes an analysis of the local environment state by doing environment quality standard checking, and provides information to the supervisor agent. The supervisor agent can be local (of a specific environment monitoring network) or global (of the global environment supervisor system). The local supervisor agent (LSA) requires reports from the environmental agents, and informs the global environment supervisor system, while the global supervisor agent (GSA) requires reports from the local supervisor agents of each environment monitoring system in order to analyze the global state of the environment.

The EnvMAS system has four multi-agent systems, associated to the four environment monitoring networks, the meteorological network (MAS-1), the air pollution monitoring network (MAS-2), the soil pollution monitoring network (MAS-3), and the water pollution monitoring network (MAS-4). Each multi-agent system has local databases.

All the multi-agent systems are sharing the same ontology, the EnvMAS ontology, which contains concepts from the environmental sciences, specific to the environmental problems that can be solved. Each environmental agent can have a list of tasks to perform according to the environmental problem that need to be solved at a given time. Some examples of tasks are: Monitor_Analyze_P, Forecast_P, Envir_Status_Report, Init_Collaboration, Knowledge_Extraction, and Control.

Thus, an agent can monitor one or more parameters, can analyze the evolution of a parameter by checking the environment quality standards, can do short-term or medium-term forecasts of a specific parameter, starting from time series stored in databases, by using artificial neural networks, for example, can apply some control measures (for the reduction of the impact on the population and on the environment), can initiate a collaboration with other agents

(to solve jointly some environmental problems), can do knowledge extraction by using data mining techniques, such as decision trees.

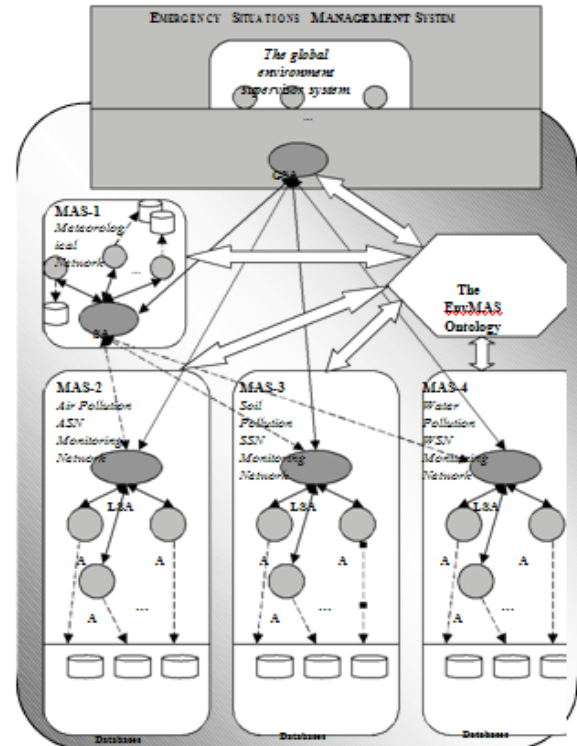


Fig.2. The architecture of the EnvMAS system.

For building decision trees, the knowledge it is extracted from a data base, through data examination and rules induction. It can be used a set of specific algorithms, such as: CHAID, CART, C5.0, ID3, J48, etc. From all these algorithms, the most used it is ID3, an inductive technique of artificial intelligence for decision trees building, that uses a special measure called information gain.

As a case study we have considered the use of collaborative intelligence between the environmental agents that monitor and analyze the evolution of some air pollutants concentrations in different locations of urban regions, and a meteorological agent, that provides information about the meteorological parameters that can influence the air pollution degree: temperature, speed of wind, wind direction, humidity, etc. The collaboration between the agents of the EnvMAS system is facilitated by asynchronous messaging and the ontology shared by all agents. A collaborative agent has some social knowledge, that includes the knowledge about the other agents (the agent's acquaintances), and is based on the shared ontology as well as on some learning capability of the agent. In the current version of the EnvMAS implementation we have not included a learning capability. Some more guidelines in developing the social knowledge of a collaborative agent are given in [17] (for social action modeling), and in [18].

We have experimented different scenarios of the EnvMAS system simulation in order to test the agents' collaboration benefits and the results are presented in the next section.

IV. EXPERIMENTS

We have implemented the first version of the EnvMAS system, named EnvMAS-Air1 [19], in Zeus [20], a toolkit for agent-based system development that allows a quick realization of a prototype multi-agent system, as it has incorporated an automated agent code generation tool. The system tackles the air pollution monitoring and analysis problem as a simulation, and makes air quality analysis in a specific urban region, based on the recorded data of concentrations for major air pollutants (carbon monoxide, sulphur dioxide, hydrogen sulfide, and particulate matter), that were taken from public available databases [21]. Four scenarios of the system run were experimented.

In the first scenario there are six environmental agents, corresponding to six air quality monitoring stations, a meteorological agent (*Meteo_agent*) which delivers the weather conditions, a local supervisor agent, (*Air_agent*), that centralizes the data from the stations agents, and a supervisor agent (*Supervisor_agent*) that receives the information needed for the experiment and analyzes the whole region air quality. The six environmental agents are scanning a database specific to each monitoring station in order to find the record from a certain day (which is an input from the user). These data are sent to the *Air_agent* to be analyzed. The *Air_agent* receives also the weather parameters for that day from the *Meteo_agent*. Based on those data, the *AirQualityStatus* is generated and sent to the *Supervisor_agent*.

Figure 3 presents the interaction diagram between the agents of the EnvMAS-Air1 system. The coordination between the agents, one of the distinguishing features of the multi-agents systems, can be regarded as the process by which the individual decisions of the agents contribute to the global goal of the system. In Zeus the agent interactions are variations on the multi-round contract-net, which involves one or more *Initiators* (e. g. *Supervisor_Agent*, *Air_Agent*) that issue a call for proposals (CFP), and one or more *Respondents* (stations agents) that reply.

In the second scenario, there are also six environmental agents for the stations, an agent that centralizes the data and the one which is the supervisor. The difference between the first two scenarios is that the *Meteo_agent* does not exist anymore and the stations agents are searching by themselves in the weather conditions databases for the weather conditions (precipitations, temperature, wind speed, wind direction, etc). The third scenario is similar to the first one, except the fact that instead of six stations agents there are twelve stations agents involved in the system. The fourth scenario is the result of the combination between the second and third scenarios: there are twelve stations agents but there is no meteo agent. The environmental agents are doing the task Monitor_

Analyze_P, by reading the data from databases and making multiple comparisons between the data supplied and the upper limit of the air pollutants' concentration, according to the air quality standard. The output is given as a general characterization of the environment (*no_problem*, if all the recorded values are normal, *weak_air_pollution*, if there are some possible exceedances, *severe_air_pollution*, if there are recorded values that may have an impact on the human health). For this, the *Supervisor_Agent* uses ID3 algorithm to generate a decision tree in order to establish the air quality status. One of the extracted rules is: If CO=*severe_air_pollution* and SO2=*weak_air_pollution* and PM10=*weak_air_pollution* and H2S=*no_problem* Then air_quality_status=*severe_air_pollution*. If the upper limit is exceeded for a certain pollutant, an alert windows pops up.

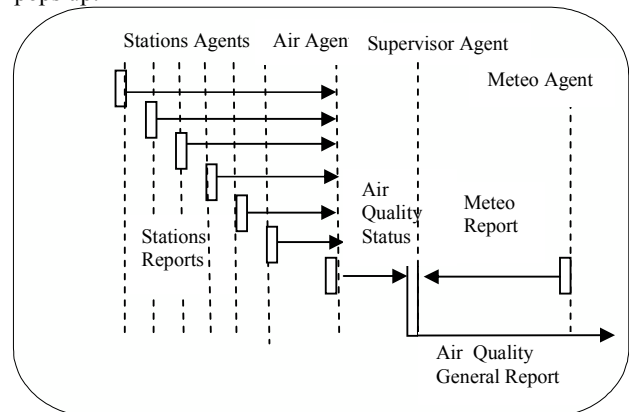


Fig.3. The interaction diagram between the agents of EnvMAS-Air1 system.

The data set used in the experiment made represents the recorded data for major air pollutants at some monitoring station that are part of the National Air Quality Monitoring Network of Romania [21]. This network consists of 117 stations placed in 37 counties for monitoring the influence of the traffic, industrial activity and urban settlements over the air quality. The measurements are made hourly, at each station, and after a primary validation they are stored in dedicated databases. The data are also published online for the population information regarding a possible pollution episode. The records of carbon monoxide (CO), sulphur dioxide (SO2), hydrogen sulfide (H2S), and particulate matter (PM10) concentrations from January 2009 until December 2010 were used for this application.

Figure 4 shows a screenshot from the ontology used in the first and third scenarios (with collaborative agents) of the EnvMAS-Air1 run. The terms included in the ontology were used in the inter-agent communication.

In Figure 5 it is given a screenshot of the EnvMAS-Air1 system messages exchange between agents during system run.

When the EnvMAS-Air1 system runs, the user has to select from the interface a certain day for the analysis. After

that, the Supervisor agent initializes the agents' communication asking the other agents to provide the recorded data for that day. A timer starts to count the seconds elapsed between the day selection and the system' goal achievement (i.e. air quality status report). The environmental agents extract the information from MySQL databases, one for each monitoring station, where all the needed data are stored.

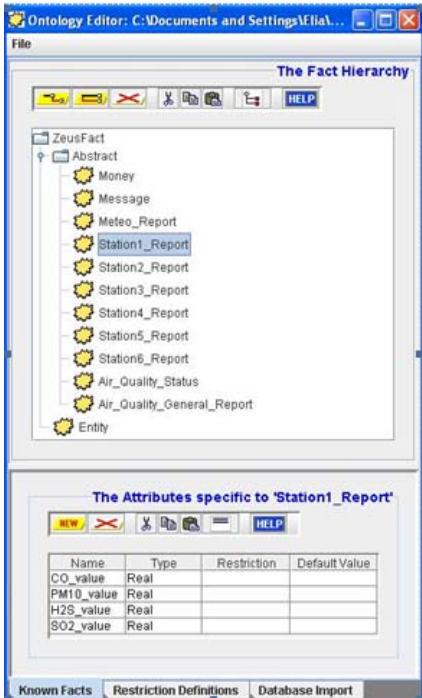


Fig.4. Screenshot with the ontology of the EnvMAS-Air1 system used in the first and the third scenarios.

In Figure 6 it is given the Air_agent DOS window with the reports received from all stations and the Meteo_agent, during the EnvMAS-Air1 system run.

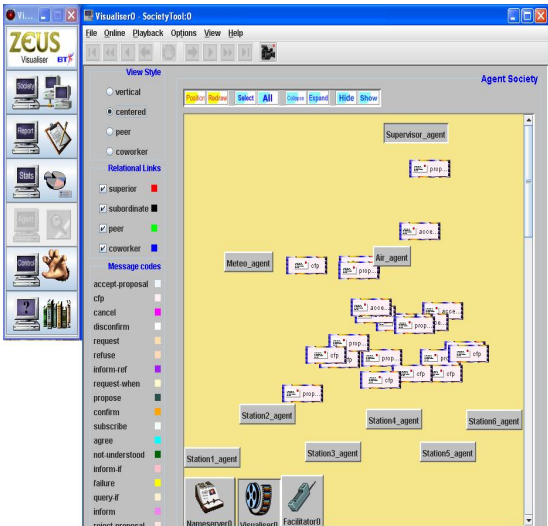


Fig.5. Screenshot of the EnvMAS-Air1 messages exchange between agents.

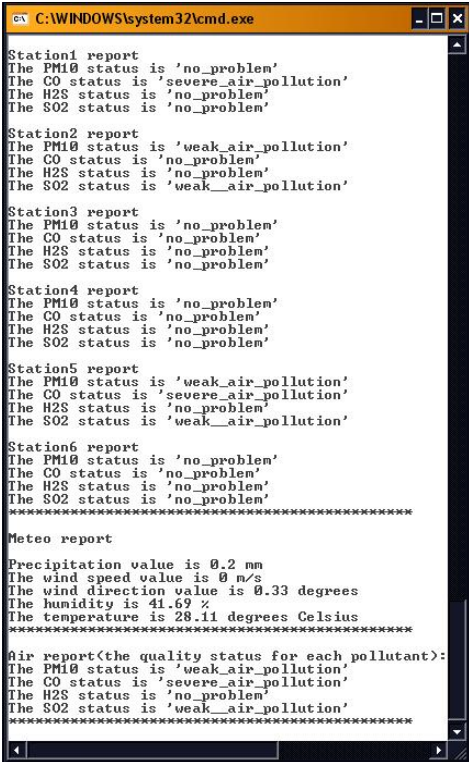


Fig. 6. Screenshot of the Air_agent DOS window during system run.

We have run the four scenarios of the EnvMAS-Air1 system, with collaborative and non-collaborative agents, for two experiments, the first one with no air pollution problems and the second one with severe air pollution problems. The first and the third scenarios, S1 and S3, were using collaborative agents, while the second and the fourth scenarios, S2 and S4, were using non-collaborative agents. The response time of the system during each scenario run were registered. The results are presented in Table 1. As expected, the response time is smaller in the cases when collaborative agents were used.

We have considered that in the case of agent-based environmental collaborative systems the response time of the system can measure the collaborative ability of agents to work jointly at their specific tasks. Some qualitative attributes that measure the collaborative ability of agent-based systems are proposed in [22] and [23].

Figure 7 shows a screenshot of the task graph during the EnvMAS-Air1 system run, obtained by the Visualiser agent, a utility agent provided by Zeus. As it can be seen, the Meteo_agent, the Air_agent and all the environmental agents associated to the air monitoring stations are collaborating in order to generate the *Air_Pollution_Report* that will be sent to the *Supervisor_agent*.

TABLE I
ENVMAS-AIR1 SYSTEM RESPONSE TIME [SECONDS]

Experiment	Collaborative		Non-Collaborative	
	S1	S3	S2	S4
1 st Experiment	10.23	11.24	13.44	15.20
2 nd Experiment	10.57	11.50	14.01	15.40

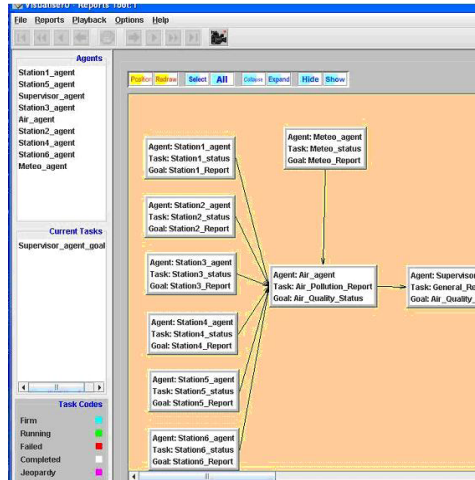


Fig. 7. Screenshot of the task graph obtained with the Visualiser agent during the EnvMAS-Air1 system run.

The experimental results obtained so far showed that agent-based collaborative environmental systems can be applied to real time air pollution monitoring and analysis, and the results provided by them can be used by the environmental decision factors in order to adopt control measures for the reduction of the pollution and of the impact on the human health. The system can be used also to inform the population about the air pollution state in a certain urban region.

V. CONCLUSIONS

We have presented a case study of using collaborative intelligence in agent-based environmental systems. The EnvMAS collaborative agent-based system, that was introduced in this paper, can be mapped to the environmental monitoring system of a given geographical area, for one or more problems, such as air pollution and water pollution. The benefits of using collaborative intelligence have been confirmed by the experimental results of the EnvMAS system run as simulation in different scenarios.

REFERENCES

- [1] J. E. Hernández, J. Mula, R. Poler and J. Pavón, "A Multiagent Negotiation-based Model to Support the Collaborative Supply Chain Planning", *Studies in Informatics and Control*, vol. 20, no. 1, pp. 43-54, 2011.
- [2] A. L. Osório, H. Afsarmanesh and L. M. Camarinha-Matos, "A Service Integration Platform for Collaborative Networks", *Studies in Informatics and Control*, vol. 20, no. 1, pp. 19-30, 2011.
- [3] J. Nishioka, "Collaborative agents for production planning and scheduling (CAPPS): a challenge to develop a new software system architecture for manufacturing management in Japan", *International Journal of Production Research*, vol. 42, no. 17, pp. 3355-3368, 2004.
- [4] S. H. Kong, S. D. Noh, Y.-G. Han, G. Kim and K. I. Lee, "An agent-based collaborative assembly process planning system", *International Journal Advance Manufacturing Technology*, vol. 28, pp. 176-173, 2006.
- [5] Q. Hao, W. Shan, Z. Zhang, S.-W. Park and J.-K. Lee, "Agent-based collaborative product design engineering: An industrial case study", *Computers in Industry*, vol. 57, no. 1, pp. 26-38, 2006.
- [6] C. Zhang, A. Hammad, "Toward Supporting Construction Equipment Operation Using Collaborative Agent-Based Systems", *Proceedings of the ASCE Int. Workshop on Computing in Civil Engineering*, 2007.
- [7] K. Lerman, A. Galstyan, A. Martinoli and A. Ijspeert, "A macroscopic analytical model of collaboration in distributed robotic systems", *Artificial Life Journal*, vol. 7, no. 4, pp. 375-393, 2001.
- [8] D. Xu, H. Wang, "Multi-agent collaboration for B2B workflow monitoring", *Knowledge Based Systems*, vol. 15, pp. 485-491, 2002.
- [9] Gh. C. Silaghi, "Coalition Formation Tools for Achieving Collaboration inside Agent Societies", *Economy Informatics*, vol. VI, no. 1-4, pp. 25-38, 2006.
- [10] Y. Xiong, G. Zheng, Q. Yang, and Y. Zhu, "A Collaborative Multiagent System for Mining Transcriptional Regulatory Elements", *IEEE Intelligent Systems*, vol. 24, no. 3, pp. 26-37, May/July 2009.
- [11] F. Enembreck, I. Thouvenim, M.-H. Abel, and J.-P. Barthès, "An Ontology-Based Multi-agent Environment to Improve Collaborative Design", ftp-sop.inria.fr/acacia/COOP2004, 2004.
- [12] J. Doran, "Agent-Based Modelling of Ecosystems for Sustainable Resource Management", M. Luck et al. (Eds), ACAI 2001, LNAI 2086, Springer, pp. 383-403, 2001.
- [13] L. Giannetti, F. P. Maturana and F. M. Discenzo, "Agent-Based Control of a Municipal Water System", M. Pěchouček, P. Petta, L. Z. Varga (Eds), Multi-Agent Systems and Applications IV, CEEMAS 2005, LNAI 3690, Springer, pp. 500-510, 2005.
- [14] C. Nichita and M. Oprea, "An agent-based model for water quality control", *Proceedings of the Int. Conf. ESCAPE17*, Elsevier, May 2007, pp. 1217-1222, 2007.
- [15] M. Oprea, A. Matei and E. Petre, "Agent-based modeling of a dam monitoring system", *Proceedings of the 17 Int. Conf. on Control Systems and Computer Science – CSCS 17*, May 2009, Politehnica Press, pp. 509-514, 2009.
- [16] G. Weiss, *Multiagent systems – A modern approach to distributed artificial intelligence*, The MIT Press, Cambridge, 1999.
- [17] C. Castelfranchi, "Modelling social action for AI agents", *Artificial Intelligence*, vol. 103, no 1-2, pp. 157-182, 1998.
- [18] V. Mařík, M. Pěchouček and O. Štěpánková, "Social knowledge in multiagent systems", *Multi-agent Systems and Applications*, ACAI, LNAI 2086, Springer, pp. 211-245, 2001.
- [19] E. Dragomir, "Development of a multi-agent system for air pollution analysis", Doctoral Research Report, University Petroleum-Gas of Ploiesti, 2011.
- [20] J. Collins and D. Ndumu, *The Application Realisation Guide*, The Zeus Agent Building Toolkit, Intelligent Systems Research Group, BT Labs, May 1999.
- [21] The Romanian National Air Quality Monitoring Network, <http://www.calitateair.ro>.
- [22] S.G. Woods and M. R. Barbacci, "Architectural Evaluation of Collaborative Agent-Based Systems", Technical Report, CMU/SEI-99-TR-025, ESC-TR-99-025, Oct. 1999.
- [23] W. Brenner, R. Zarnekow and H. Wittig, *Intelligent software agents, Foundations and Applications*, Berlin, Springer-Verla, 1998.
- [24] M. Oprea, *AIR_POLLUTION_Onto: an Ontology for Air Pollution Analysis and Control*, Artificial Intelligence Applications and Innovations III IFIP Advances in Information and Communication Technology, 2009, Volume 296/2009
- [25] SWEET ontology, <http://sweet.jpl.nasa.gov/ontology/>.
- [26] M.Oprea, Mapping Ontologies in an Air Pollution Monitoring and Control Agent-Based System, *Discovery Science Lecture Notes in Computer Science*, 2006, Volume 4265/2006, 342-346.

On Resource Allocation in a Holonic Manufacturing Execution System

Carlos Pascal and Doru Panescu

Abstract—One of the key design issues for distributed systems is to find the proper planning and coordination mechanisms when knowledge and decision capabilities are spread along the system. Holonic manufacturing execution systems regard a control architecture that is based on multi-agent systems. This contribution highlights the way a proper modeling method – the Petri nets are used – makes evident certain problems that can appear when agents have to simultaneously treat more goals. According to the holonic organization the planning phase is mainly dependent on finding an appropriate resource allocation mechanism. First, the type of weakness is made evident by the means of the proposed Petri net model and then further proved by simulation experiments. A solution to make the holonic scheme avoid a failure in resource allocation is mentioned, too.

I. INTRODUCTION

THE design and implementation of appropriate mechanisms for the control of manufacturing execution systems is still an open problem, and the research in Artificial Intelligence (AI) has had an important impact for this subject [1]-[3]. In this respect, an example is the holonic approach, which is considered in this contribution; it combines benefits of the hierarchical and heterarchical manufacturing control architectures. The Holonic Manufacturing Execution Systems (HMESs) are clearly influenced by planning and coordination mechanisms established in AI, primarily in the field of multi-agent systems [2]-[4]. An HMES regards a control scheme for the shop-floor level of a manufacturing company that is developed around autonomous, intelligent entities, named holons. The most often used holonic taxonomy is derived from the PROSA reference architecture [5], [6]. This takes into account four types of holons: products, resource, order and staff holons.

Without giving all the details of the proposed holonic scheme construction, this paper aims at showing how an appropriate modeling and analysis phase can reveal certain problems for the HMES functioning. Namely, due to the distributed nature of control within the HMES, special coordination mechanisms are needed. Though some protocols from multi-agent systems can be applied, they can determine definite drawbacks for the manufacturing control systems, requiring an appropriate tuning.

C. Pascal is with the Department of Automatic Control and Applied Informatics, “Gheorghe Asachi” Technical University of Iasi, Romania (e-mail: cpascal@tuiasi.ro).

D. Panescu is with the Department of Automatic Control and Applied Informatics, “Gheorghe Asachi” Technical University of Iasi, Romania (e-mail: dorup@tuiasi.ro).

First, the present contribution sketches a Petri net model for the HMES, which is able to show the possible faulty operation of the holonic scheme. This comes from the way the resource allocation is handled, so that this problem is analyzed, exemplified with some simulation experiments and a solution is proposed.

II. A PETRI NET MODEL OF THE HOLONIC AGENT OPERATION

As the main operation unit of the HMES, a holon is composed of three components [7] (see Fig. 1): a decisional part in charge with managing the received goals and finding solutions for them; this is materialized under the form of a holonic agent. This has to apply a combination between planning and coordination procedures, as within the HMES a goal is frequently solved by the cooperation of several holons. The holonic agent’s decisions are put into practice by the holon’s structural component. For a resource holon this is a proper physical device: a robot or machine tool controller, a PLC commanding a conveyor, etc. In the case of an order or a product holon the structural component becomes a holarchy, namely a group of holons that are conducted by the respective holon in order to solve a goal by cooperation. The information is changed between the holonic agent and the structural component by the means of a communication interface.

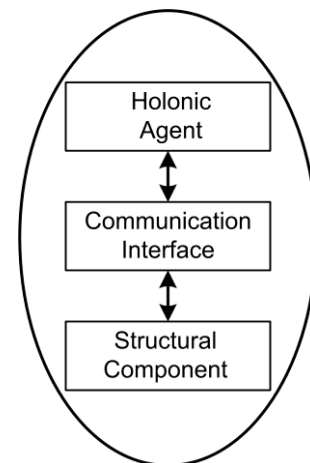


Fig. 1. Generic structure of a holon

The HMES operation is both goal and event driven: it has to solve all the goals received at the shop floor level, taking into account the events happening in the manufacturing environment (the state of various devices, the raw parts that are supplied, etc.). It is clear that the whole system operation is

determined by the holons' behavior, which is dictated by holonic agents. All these conduct to the necessity of a model for the holonic agent and indicate the Petri net formalism as a good choice [8], [9].

The proposed basic model is a general one (see Fig. 2), being applicable independent of the holons' type, be it an order, product or resource one. The model shows the two main processes that the holonic agent has to pass through as the decisional component, namely planning when it finds a plan for solving a goal and execution when it applies the decided plan and monitors its carrying out. Besides these, an idle agent's state is present, which shows the agent's availability, allowing the switching between the holonic agent's processes, too.

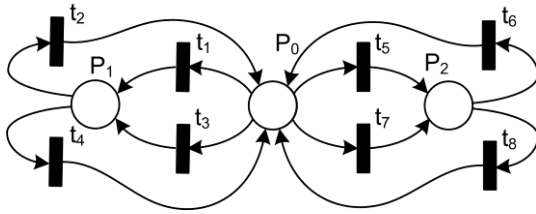


Fig. 2. The holonic agent basic Petri net model

In the Petri net model the corresponding places are: P_0 for the idle state, P_1 for the planning phase and P_2 represents the execution one. The transitions that determine the passing to planning are t_1 representing a goal receiving and t_3 modeling the receipt of a set of bids. This is explained by the way a goal is solved: besides the easy case when the goal can be worked out by a single resource holon through its own physical device activity, the other cases imply a cooperation between several holons. This is managed in the HMES through the use of the Contract Net Protocol (CNP), the common coordination method of multi-agent systems [10], [11]. It supposes that a holon not able to solve a goal by itself becomes a manager, asking the cooperation of the other holons, by sending appropriate goals/sub-goals. Those holons able to provide a solution reply with corresponding bids (in fact the holonic agents are sending the bids), the best one being selected by the manager. The holon that made the respective offer receives a contract from the manager holonic agent, to put into practice the solution it proposed. The usual multi-agent CNP has to be adapted and enhanced for a holonic use, in order to obtain a reliable and near to optimum HMES operation [12]. Moreover, a further tuning is needed when the agents' inference mechanism is based on the Belief Desire Intention architecture [13], [14]. Certain details regarding the way planning and coordination are supported by the holonic agent inference process will be further presented.

Knowing the main steps of the CNP it is clear that the planning stage is interrupted or finalized at two types of transitions: t_2 regards the sending of a goal (sub-goal) and t_4 appears when the agent releases an internal contract allowing the start of the execution phase or when it issues a bid. As the model aims at being a general one, it considers the case when

the same holonic agent can be a manager and a contractor, too. That is way the agent can receive and announce goals, while it can also propose bids and issue internal contracts. There are two cases when the internal contracts are used: when the holonic agent is only manager, as it can be for an order holon, and when the holonic agent of a resource holon can solve a goal by commanding its structural component. The usual contracts, those sent to other holons, are named external contracts or simply, contracts.

The Petri net model reflects both above mentioned cases, when the agent is able to solve by itself a goal (the corresponding succession is: $P_0 t_1 P_1 t_4 P_0$), and when it has to apply for collaboration, this evolution being modeled by the succession: $P_0 t_1 P_1 \{t_2 P_0 t_3 P_1\} t_4 P_0$. The notation within curly brackets means that the sequence $t_2 P_0 t_3 P_1$ can appear no time, when there is no goal considered for collaboration, or it can be used one or more times, depending on the number of goals that the agent issued and for which is waiting cooperation.

The execution stage represents the carrying out of a previous holonic agent's commitment, represented by a bid it made. This is started from the agent's idle state when it receives a contract. The respective event is modeled by the transition t_5 . After that, through the transition t_8 the holonic agent sends the contracts to its sub-contractors or, for a resource holon, commands towards its physical device. The transition t_7 models the feedback from the contractors or from the controlled device regarding the ending (with success or failure) of a contract or an action.

When an entire contract is ended, the transition t_6 designates the finalization of the execution phase. If the execution regarded an external contract t_6 provides a feedback towards the manager, and this includes the case when the contract could not be accomplished and the respective holon is not able to solve the failure by itself. When the holon can try a further solution to a failed contract or action, the transition t_6 determines the issuing of an internal (that is not sent by another holonic agent) goal, thus re-starting the planning phase. To conclude, the execution sequence is reflected by the proposed model as follows: $P_0 t_5 P_2 t_8 P_0 t_7 P_2 \{t_8 P_0 t_7 P_2\} t_6 P_0$. It is to further underline that the proposed model catches (through the transitions t_1 and t_3 , respectively t_5 and t_7) all the cases when a planning/execution stage is started or resumed for the holonic agent, thus being a general one. The devised Petri net model reflects a holonic agent behavior, but the whole HMES activity must be supported by a proper communication mechanism, as shown in the next section.

III. THE MODEL OF THE HOLONIC AGENTS' COMMUNICATION

When each holonic agent is represented by the Petri net model of Fig. 2, the communication between holons can be modeled according to Fig. 3a and b. It is considered a holonic interaction with three holons, named H_1 , H_2 and H_3 (see Fig. 3a). In this example, the communication starts when H_1 , as manager within the CNP, issues a goal through the transition

$t_{2(H1)}$ (the notation of Fig. 2 is respected, too). The result of this transition, as the agent's message, is placed in a buffer represented by the place P_{OUT2} in Fig. 3a. From here, by the means of the communication network, it is transmitted to all the possible contractor holonic agents, being inputted into their buffers, marked as the places P_{IN1} . Thus the transition t_1 regarding the presence of a new goal can be fired (in our example, two holons, H_2 and H_3 , activate the corresponding transitions: $t_{1(H2)}$ and $t_{1(H3)}$).

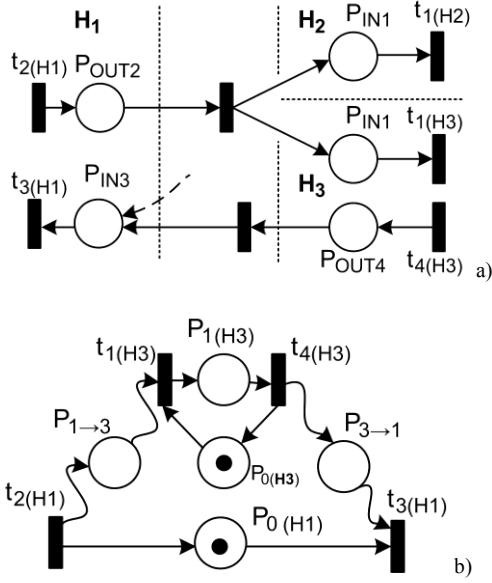


Fig. 3. Model of the holons' communication

Each contractor enters the planning phase only when the corresponding holonic agent is freed from other activities. As an example of the proposed model application, for the holon H_3 this means a token is present in the place $P_0(H3)$ (see Fig. 3b). The planning phase will have as result a message for a bid (an output buffer is used), corresponding in our model with the placement of a token in the place P_{OUT4} when the transition t_4 (the one for bid's sending) is fired ($t_{4(H3)}$ in Fig. 3a, b). From this buffer, the communication network transfers the message to the buffer (the place P_{IN3}) of the manager agent; thus the transition t_3 can be fired for this agent ($t_{3(H1)}$ in our example). In Fig. 3b the places $P_{1 \rightarrow 3}$ and $P_{3 \rightarrow 1}$ abstract the information (goals and bids) transfer from one holon to the other by the means of their buffers and the communication network.

A similar mechanism exists for the transmission of contracts and the feedbacks issued at the end of contracts. The agents' communication buffers should allow several goals/contracts to be received, and thus it can happen that a holonic agent has to treat more goals and/or contracts. Therefore, the issue of treating several goals by the same holon is to be discussed, and also the case when the solutions provided to a set of goals by some holons interfere. To understand these problems for a BDI based holonic agent, it is necessary to be aware of the operation principle for the BDI mechanism.

IV. THE PLANNING AND EXECUTION PROCESSES FOR A BDI HOLONIC AGENT

The planning phase of a holonic agent that uses the BDI inference mechanism is conducted according to a sequence consisting in the following steps (the planning cycle is started when a goal is received):

Step 1. Choose an execution workflow that could solve the goal. If there is no further choice in the agent's library of plans, then the cycle is ended by declining the goal.

Step 2. Try to validate the selected execution workflow (this means to find resources able to carry out the actions of the workflow).

Step 3. If the selected execution workflow has been validated then the planning phase is successfully ended (the corresponding bid/internal contract is sent), else the cycle is restarted with the Step 1.

The peculiar aspect regards the library of plans that endows the BDI agents (see, for example, the JACK software platform dedicated to implementing BDI agents, which allows the definition and use of plans [14]). For each type of goal the BDI agent must possess a set of plans that it can try when faced with a goal. In the proposed approach these plans are un-instantiated execution workflows. Here, we name as workflow the whole sequence of actions that the holonic agent uses for an entire execution phase. An un-instantiated execution workflow specifies the sequence of actions that can solve a certain goal, but with the actors to carry out the actions being un-specified. Thus, during the planning process the holonic agent tries to validate an execution workflow by establishing (mostly through the means of the CNP) which will be the entities (other holons or its own physical device) to perform the actions of the execution workflow. This can be regarded as a process of resource allocation and it can also be represented by an appropriate use of the holonic agent Petri net model.

As an example, Fig. 4 shows an execution workflow that is relevant for solving the goal g (this appears as superior index for the entities of the model). The workflow contains two actions – a_1 and a_2 , for which the holon that received the goal, acting as manager, has to find actors (contractors). These can be viewed as resources and they are modeled by the places ${}^sP_{Rai}$ added to the basic Petri net model. When the execution workflow is chosen in the first step of the planning process (the selection is based on the relevance for a goal and on certain optimum criteria) no token is present in the places ${}^sP_{Rai}$. If the planning cycle is a successful one, then a token will be present in each of the places ${}^sP_{Rai}$.

After the planning phase, the instantiated execution workflow is used to entirely guide the execution phase, as the model of Fig. 4 shows (one has to notice how this complies with the Petri model of Fig. 2). During execution no decision points appear, except for the case of an action's failure, when a planning phase must be restarted. The transition ${}^e t_5$ starts a

first execution stage (marked as ${}^gP_{2(1)}$ in Fig. 4) as result of the contract awarding (see t_5 in Fig. 2). According to the validated workflow, the execution continues with awarding a contract towards the chosen sub-contractor – the transition ${}^gt_{8(1)}$. The place ${}^gP_{a1}$ reflects the execution of the needed action (a_1) by sub-contractor, being ended at the transition ${}^gt_{7(1)}$, when the holonic agent receives the feedback concerning the action's end. The place ${}^gP_{Ra1}$ abstracts the state of the contractor; the token in ${}^gP_{Ra1}$ marks the commitment made at the moment of bidding, regarding the engagement of achieving the action a_1 . The place ${}^gP_{2(2)}$ represents a second execution stage, with the same progress as the first one. The places ${}^gP_{a2}$, ${}^gP_{Ra2}$ and the transitions ${}^gt_{8(2)}$, ${}^gt_{7(2)}$ have the same meaning as for the previous execution stage, this time regarding the action a_2 . The considered example illustrates the general case of a holon being both contractor and manager: it is contractor for the received goal g , and it is manager with respect to finding solutions to achieve the actions a_1 and a_2 .

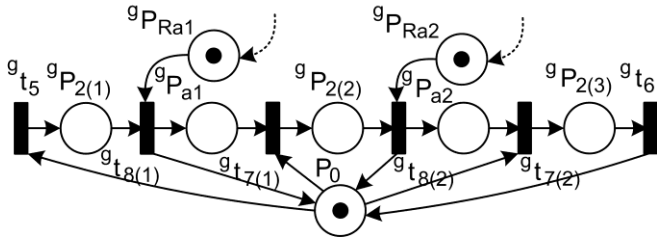


Fig. 4. An execution workflow with two actions to be solved by cooperation

V. ANALYSIS OF THE HOLONS' INTERACTIONS; EXPERIMENTAL RESULTS

As already explained in previous sections, a holonic agent is involved in two types of activities, for planning and execution. Because these activities are not continuous (after sending a goal to the potential contractors the agent has to wait for the bids, after sending a bid the agent waits for the manager's answer, after sending a contract the agent has to wait its accomplishing), it can happen that the processes regarding several activities are interleaved for the same holonic agent. It results that three types of combinations are possible: two planning processes are interleaved, two execution workflows are simultaneously undertaken, or one planning and one execution process are handled by the holonic agent. The last two cases do not need a special attention, because as long as an execution workflow was validated by the planning phase it cannot faultily influence another process.

The significant combination is when two planning processes of the same holon are in progress in the same time. Each planning process is started by a distinct goal. If the agent has received two goals and their treatment is interleaved, then the case of Fig. 5 can happen, where the superior indices 1 and 2 refer the two goals (see the index g in Fig. 4). The agent works with two execution workflows, which it tries to validate for the two goals. With respect to this, the agent has

announced goals (sub-goals) in order to find contractors for the actions of the execution workflows (in our example all these actions need other holons to carry them out). The problem is that the actions represented by the places ${}^1P_{a1}$ and ${}^2P_{a1}$ need the same type of resources, the same condition being true for the places ${}^1P_{a2}$ and ${}^2P_{a2}$. If the contractors managing the two types of resources happen to make the bids for the two workflows as shown by the tokens placed in Fig. 5, both planning processes fail, as they cannot transform the corresponding execution workflows into live Petri nets.

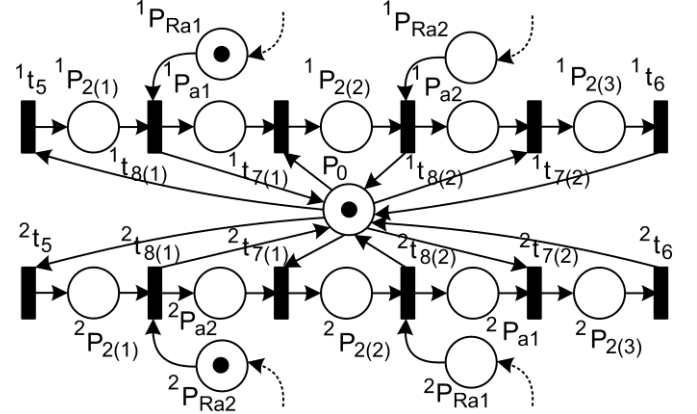


Fig. 5. An example of interaction during the planning process

The above case happens even the HMES could provide a solution at least for one goal. This occurs if a single planning process is allowed to start and only after its finalizing the second planning process begins. This rule is not to be always applied, because it reduces the HMES flexibility: it can also be possible that two planning processes are treated in the same time without deadlock. The solution is to restrict the simultaneous activation for validation (for planning) of more execution workflows that could determine a blockage, as they refer to common resources. This can be implemented by correspondingly marking the plans (execution workflows) within the agent's planning library. The proposed Petri net model can help this marking operation, by using it in a simulation developed before the start of the holon's activity within the HMES, during which the interaction of plans can be revealed.

A similar situation can happen when the two goals that create a conflict are handled by two distinct holonic agents. The case presented in Fig. 5 reflects these circumstances too, except for the fact that two places P_0 will exist, one for each holon. Now the solution is beyond the two agents' possibilities, as they represent distinct, autonomous entities. There is the need of another component, and this can be a staff holon in the PROSA architecture, which is supposed to take the decision on the management of goals. This type of interaction must be taken into account only between the product and order holons, because for the resource holons the solution can be obtained at their own level, as they can distinguish the goals so that the deadlock is avoided [8]. When

the product/order holons need to announce goals towards the resource holons, they should require the acceptance of a dedicated staff holon. This will send the approval only to one requesting holon and keep in a queue the other enquiries that refer to the same type of resources. At the moment of planning finalization, an agent of an order or product holon has to announce the staff holon about this, so that it can consider the next request. In this way two planning processes that refer some common resources are never interleaved and any deadlock is avoided.

The above theoretical points were validated through experimental results obtained by simulation. In order to conduct significant experiments, a complex model and a simulation environment able to represent an entire HMES were developed [15]. These could not be created by the means of monochrome Petri nets, but the use of Colour Petri Nets (CPNs) was necessary [16]. The constructed simulation environment appears as a hierarchical CPN, with the highest level abstracting various entities of the HMES in the form of transitions and places. These are expanded on successive levels, taking into account the proposed basic Petri net model (the one of Fig. 2), but with tokens that can carry different information, according with the formalism of CPNs. The experiments carried out to prove the above analysis considered an HMES with one product holon and some resource holons, together with the network necessary to handle the communication between holons. An important propriety of the CPNs is the way they combine the capabilities of monochrome Petri nets with the support of a high-level programming language, such as the Standard ML [17]. So, it was possible to obtain a highly configurable model-prototype, close to the real HMES implementation.

In principle, a simulation experiment can be used to explore only a finite number of HMES executions. The use of the reachability graph of a Petri net offers a possibility to surpass this difficulty. Indeed, this covers the entire state space of the modeled system, starting from a given initial state [16]. It means the reachability graph can be a powerful tool for assessing the properties of HMESs, and that is way it was chosen. More specific, regarding the problem of establishing the output of the holonic agents' planning process, the leaf nodes of the reachability graph contain information about the results of the resource allocation process. The leaf nodes represent the final or dead markings of the analyzed Petri net, and by their examination the results of the holonic planning process can be understood.

Thus, to study the interaction between two planning processes of the same holon (as the one presented in Fig. 5), four cases were considered. The difference between them is given by the availability of resource holons and the treatment of received goals. In cases 1 and 2, the product holon has to face a manufacturing environment with limited resources allowing only one goal to be fulfilled; this restriction is removed in the cases 3 and 4. On the other hand, the product holon tries similar plans in the cases 1 and 3, respectively 2

and 4 (it attempts to instantiate similar execution workflows).

The results of the considered cases are presented in Tables I-IV. These contain three data types. The first two rows give the initial HMES state: the goals received by the product holon, the actions included into the execution workflows that the product holon uses to solve the goals and the resource holons able to carry out the respective actions. This information is used to establish the initial marking of the Petri net model. Data on the reachability graph (the number of nodes, arcs and dead markings) are presented in the next two rows of the tables, while the following rows summarize the results of the planning process. Having two goals to solve, the planning results, reflected in the resource allocation, are classified into three classes. These mean the planning process succeeded to instantiate two execution workflows, one of them (with two sub-cases) or none. For each class, the indicated percentage represents the number of final states (dead markings) that correspond to the respective class from the total number of final states.

When both goals are solved by the same execution workflow (the succession of actions is a_1, a_2) and two resource holons (RH_1, RH_2) are present in the HMES, only one goal can be fulfilled, as Table I shows. In this first experiment no conflict between the planning processes appears. When the HMES context is the same, except for the execution workflows used to treat the goals, the results of Table II are obtained. These indicate that a conflict is possible: there are 3.23% of the total number of dead markings that represent cases when neither of the two goals is solved. It means the product holon fails to accomplish at least one goal, despite the fact that a solution exists. This is exactly the case of Fig. 5, when during the planning process two execution workflows are simultaneously treated and it happens that the resource holons make bids for the first action in each workflow; thus, no resource is available to complete a planning process. This situation does not appear in the experiment considered in Table I, because in that case one of the execution workflows is already abandoned when the product holon does not receive a bid for its first action, and thus the failure situation is avoided.

The case in Table III is similar with the first experiment, but this time there are dead markings representing the fulfillment of both goals, because more resource holons exist in the HMES. When the same experiment is made with different workflows used in planning, the results are worse: the

TABLE I
AN EXPERIMENT WITH TWO GOALS AND REDUCED RESOURCES

Product Holon's State		Goal 1 (a_1 - a_2)	Goal 2 (a_1 - a_2)
Resource Holons' State		$RH_1 \rightarrow a_1, RH_2 \rightarrow a_2$	
Nodes	13525	Arcs	23030
Dead Markings		80	
Goal 1		Goal 2	%
succeed		succeed	0
succeed		fail	50
fail		succeed	50
fail		fail	0

TABLE II
AN EXPERIMENT WITH TWO GOALS AND DIFFERENT EXECUTION
WORKFLOWS

Product Holon's State	Goal 1 (a1-a2)	Goal 2 (a2-a1)
Resource Holons' State	$RH_1 \rightarrow a_1, RH_2 \rightarrow a_2$	
Nodes	55271	116622
Dead Markings	186	
Goal 1	Goal 2	%
succeed	succeed	0
succeed	fail	48.39
fail	succeed	48.39
fail	fail	3.23

TABLE III
AN EXPERIMENT WITH TWO GOALS AND ENOUGH RESOURCES

Product Holon's State	Goal 1 (a1-a2)	Goal 2 (a1-a2)
Resource Holons' State	$RH_1 \rightarrow a_1, RH_2 \rightarrow a_2, RH_3 \rightarrow a_1, RH_4 \rightarrow a_2$	
Nodes	63257	139552
Dead Markings	164	
Goal 1	Goal 2	%
succeed	succeed	39.02
succeed	fail	30.49
fail	succeed	30.39
fail	fail	0

TABLE IV
AN EXPERIMENT WITH ENOUGH RESOURCES AND DIFFERENT EXECUTION
WORKFLOWS

Product Holon's State	Goal 1 (a1-a2)	Goal 2 (a2-a1)
Resource Holons' State	$RH_1 \rightarrow a_1, RH_2 \rightarrow a_2, RH_3 \rightarrow a_1, RH_4 \rightarrow a_2$	
Nodes	174575	396964
Dead Markings	324	
Goal 1	Goal 2	%
succeed	succeed	23.26
succeed	fail	38.37
fail	succeed	38.37
fail	fail	0

percentage of 23.26% dead markings representing the accomplishment of both goals in Table IV is less than the value in Table III. The explanation for the difference between the results of Table III and Table IV is the same as for the cases in Table I and Table II. It is to notice that the number of the dead markings is much higher than the number of planning results, which is explained by the internal mechanism that labels with distinct identifiers the messages sent between the holonic agents, conducting to different markings in the Petri net. Anyhow, the proper reading of the results obtained with these simulation experiments proves the theoretical points.

VI. CONCLUSION

The work described in this paper addresses the modeling and analyzing techniques capable of revealing certain planning and coordination issues of multi-agent systems included in HMESs. The proposed Petri net model describes

the internal and external behavior of holons. It can be used to construct the reachability graph, which provides important data on the states the HMES can pass through. The analysis has shown the necessity of an appropriate protocol that the holonic agents should use for the handling of plans, and moreover the need of a centralized component to manage the possible conflicts among the planning processes of different holons. These results led us to considering the staff holon as a required entity into an HMES, with the ability to protect the system against the potential conflicts.

The planned future work aims at better formalizing and evaluating the BDI mechanism when this is involved in the holonic agents' operation. Thus we are supposed to complete a systematic method for the application of multi-agent systems in manufacturing control.

REFERENCES

- [1] G. Candido and J. Barata, "A Multiagent Control System for Shop Floor Assembly" in *Proc. HoloMAS 2007*, Springer-Verlag, 2007, pp. 293-302.
- [2] J. Jarvis, D. Jarvis, R. Ronnquist, and L. Jain, *Holonic Execution: A BDI Approach*, Studies in Computational Intelligence, vol. 106, Springer-Verlag, Berlin, 2008, pp. 1 - 32.
- [3] G. Morel, P. Valckenaers, J-M Faure, C. Pereira, and C. Diedrich, "Manufacturing plant control challenges and issues", *Control Engineering Practice*, 15, 2007, pp.1321- 1331.
- [4] A. Giret and V. Botti, "Engineering Holonic Manufacturing Systems", *Computers in Industry*, 60, 2009, pp. 428 – 440.
- [5] H. Van Brussel, J. Wyns, P. Valckenaers, L. Bongaerts, and P. Peeters, "Reference architecture for holonic manufacturing systems: PROSA", *Computers in Industry*, 37, 1998, pp. 255 – 274.
- [6] F. T. Cheng, C. F. Chang, and S. L. Wu, "Development of holonic manufacturing execution systems", *J. Intelligent Manufacturing*, 15, 2004, pp. 253-267.
- [7] D. Panescu, M. Sutu, C. Pascal, "On the Design and Implementation of Holonic Manufacturing Systems", in *Proc. of the WRI World Congress on Computer Science & Information Engineering*, 2009, pp. 456– 461.
- [8] D. Panescu and C. Pascal, "Some Issues on Holonic Systems Analysis, Design and Implementation", in *Proc. of the RAAD 2010*, Budapest, 2010, pp. 199 – 204.
- [9] F. S. Hsieh, "Holarchy formation and optimization in holonic manufacturing systems with contract net", *Automatica*, 44, 2008, pp. 959–970.
- [10] R. G. Smith, "The contract net protocol: High level communication and control in a distributed problem solver", *IEEE Transactions on Computers*, C-29, 1980, pp. 1104 – 1113.
- [11] F. S. Hsieh, "Analysis of contract net in multi-agent systems", *Automatica*, 42, 2006, 733 – 740.
- [12] D. Panescu, C. Pascal, M. Sutu, and G. Varvara, "Collaborative Robotic System Obtained by Combining Planning and Holonic Architecture", in *Proc. of AT-EQUAL 2009*, pp 138-143.
- [13] A. Rao and M. Georgeff, "Modeling rational agents within a BDI-architecture", in *Principles of Knowledge Representation and Reasoning, Proc. of the Second Internat. Conference KR91*, 1991, pp 473-484.
- [14] R. Evertsz, M. Fletcher, R. Jones, J. Jarvis, J. Brusey and S. Dance, "Implementing Industrial Multiagent Systems using JACKTM", in *Proc. of PROMAS*, 2003, Australia, Springer, pp.18-48.
- [15] D. Panescu, C. Pascal, "On a holonic adaptive plan-based architecture: planning scheme and holons' life periods", *International Journal of Advanced Manufacturing Technology*, submitted for publication.
- [16] K. Jensen, L. Kristensen, *Coloured Petri Nets: Modeling and Validation of Concurrent Systems*. Springer-Verlag, New York, 2009.
- [17] R. Harper, *Programming in Standard ML*. Carnegie Mellon University, 2005.

Real Time Reconstruction of Volumes from Very Large Datasets Using CUDA

Lucian Petrescu, Anca Morar, Florica Moldoveanu, Victor Asavei

Abstract—This article presents a memory efficient implementation of the Marching Cubes algorithm using NVIDIA's CUDA technology. The algorithm can handle datasets that are normally too large for current hardware by splitting the initial volume into several smaller subvolumes while minimizing extra computations caused by subvolume overlapping. Moreover, our approach is scalable, making it easy to benefit from additional computational resources.

I. INTRODUCTION

ISOSURFACE extraction is a common technique applied in scientific visualization and medical imaging. Isosurfaces are surfaces obtained by uniting points which have the same scalar value within a volume dataset. They are often rendered to show particular structures characterized by the same value inside the volume. Usually, for a CT dataset, this value represents a radiodensity (measured in Hounsfield Units) of interest in the reconstruction of the volume. Additionally, it is often useful to have the mesh representation of these surfaces available for computation of quantities, such as surface area, or for processes that require the explicit form of the surface, such as collision detections or modeling problems.

Generally, the user has a clear idea of what radiodensity is of interest in the dataset. This value, also referred to as isovalue, is in a range considered to be reasonable for the extraction of features of interest from that dataset. However, the user might not know exactly what isovalue should be used; therefore a fast isosurface changing mechanism is vital for real time applications such as medical surface reconstructions or real-time deformation simulations.

If the surface needs only to be visualized, then it is sufficient to use one of the algorithms from the ray-casting family to achieve the wanted results. They have been proven to offer the least visual artifact prone implementations to the volume visualization problem. If, instead, the surface needs to be reconstructed, the most widely used algorithms for this kind of processing are Marching Cubes (MC) and Marching Tetrahedrons (MT). This paper presents an algorithm that gives the possibility to reconstruct surfaces from much larger datasets [4] than those reconstructed with classic Marching Cubes and Marching Tetrahedrons without using surface generalization or simplification.

This article presents our research within the SABIMAS project (Advanced Information System, based on medical

imaging, for personalized implants production in hip arthroplasty: <http://se.cs.pub.ro/SABIMAS>), financed by the Ministry of Education, Research and Innovation, ANCS, CNMP, Program 4 – “Partnerships in Priority Areas” from the National Plan II 2007-2013. Part of the work has been funded by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/88/1.5/S/61178.

The following chapter of this article lists the current volume rendering techniques, in order to represent the challenges of this domain. The third section presents a series of problems that led to the development of a new algorithm for 3D surface reconstruction. Chapter four describes the new approach to Marching Cubes using a partitioning, CUDA compatible method, designed for real-time applications that handle very large datasets. Next, a series of visual and numeric results are presented in order to show the advantages of this new approach. In the last two chapters of the paper we draw the conclusions and present our future research plans.

II. RELATED WORK

In medicine, volume rendering is used for 3D visualization of the internal structures of scanned body parts - especially for diagnostics - and for the geometric model reconstruction of the scanned body part, which can be used for prototyping, in plastic surgeries, hip arthroplasty, etc. A volume dataset is obtained from a medical imaging device, in the form of a stack of 2D images which represent slices (sections) through the scanned portion. The dataset is geometrically represented by a volume composed of equal cubic voxels, each voxel having a scalar value from the dataset. The volume is referred to as *data volume*. Figure 1 illustrates the geometrical representation of the dataset.

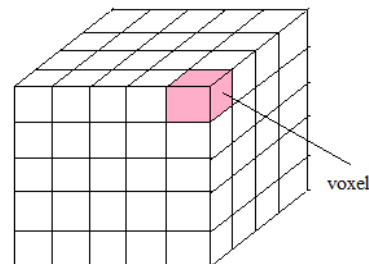


Fig. 1. Geometrical representation of the data volume

There are two methods for volume rendering [5]:

- Direct volume rendering, using special algorithms
- Indirect volume rendering: a 3D model of a surface is extracted from the data volume and then this is visualized by usual surface rendering methods.

In the following sections we briefly present one representative algorithm for each approach.

A. Direct Volume Rendering – Ray Casting

In the Volume Ray Casting algorithm, for each pixel, a ray from the eye to the pixel is cast into the data volume, sampling the volume at certain intervals. The color and opacity are computed for every sample, and then composited along the ray path.

This algorithm can be implemented in a parallel manner, due to the fact that the computations are the same for each ray. For example, using NVIDIA's CUDA [6], the process of obtaining and compositing the samples to obtain the final color for a pixel can be defined in a CUDA kernel that will be called for each pixel.

Among the advantages of this method we can mention that ray casting obtains images of high quality, with low memory usage. This algorithm is also easy to implement and very fast. But one of its main disadvantages is that it can only be used for visualization. For example, if we want to obtain the 3D model of the pelvic and femoral bones, in order to prototype the prosthesis at hip level, then we need a different volume rendering method.

B. 3D Surface Reconstruction - Marching Cubes

Marching Cubes [1] is a surface reconstruction algorithm. It extracts an isosurface from the volume of voxels, computing the intersections of this isosurface with the border voxels.

The vertices of a voxel can be:

- outside the isosurface (value of the vertex < isovalue)
- on the isosurface (value of the vertex = isovalue)
- inside the isosurface (value of the vertex > isovalue)

A border voxel does not have all of its vertices either inside or outside the isosurface. Figure 2 presents the intersection between a voxel and the isosurface, based on the values of the voxel's vertices.

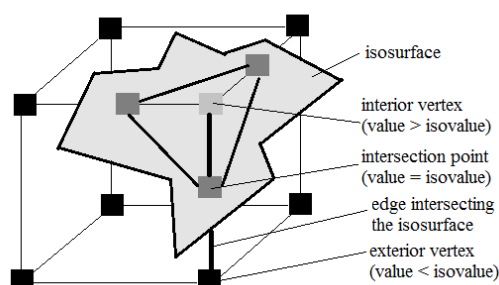


Fig. 2. Computing the intersection between the isosurface and a voxel

There are 256 possible configurations of the vertices of a

voxel relative to the isosurface. For each configuration, the triangle mesh approximating the part of the isosurface can be predefined. For the coding of the 256 configurations, two lookup tables are used:

- The edge table, which defines edges that intersect the isosurface, given a combination of vertices
- The triangle table, which defines how to connect the intersection points.

Marching Tetrahedrons is a variation to the Marching Cubes algorithm, splitting each cube into six irregular tetrahedrons. This method clarifies a minor ambiguity problem of Marching Cubes with some cube configurations. This algorithm generates more accurate isosurfaces, but because it divides the cubes into 6 sub-regions, the number of triangles generated by the Marching Tetrahedrons algorithm is greater than the number of triangles generated by the Marching Cubes algorithm, increasing the time to render the isosurfaces.

The main advantage of the 3D surface reconstruction method is in obtaining the geometric model of the scanned body, making it useful in medical domains, for prototyping prostheses or for predicting the resulting body parts after surgeries. But due to the increased complexity, the indirect visualization model (Marching Cubes/Tetrahedrons) implies high memory usage and is artifact prone and time consuming.

Figure 3 presents the reconstruction with Marching Cubes of a body part from a volume obtained from a set of CT images.

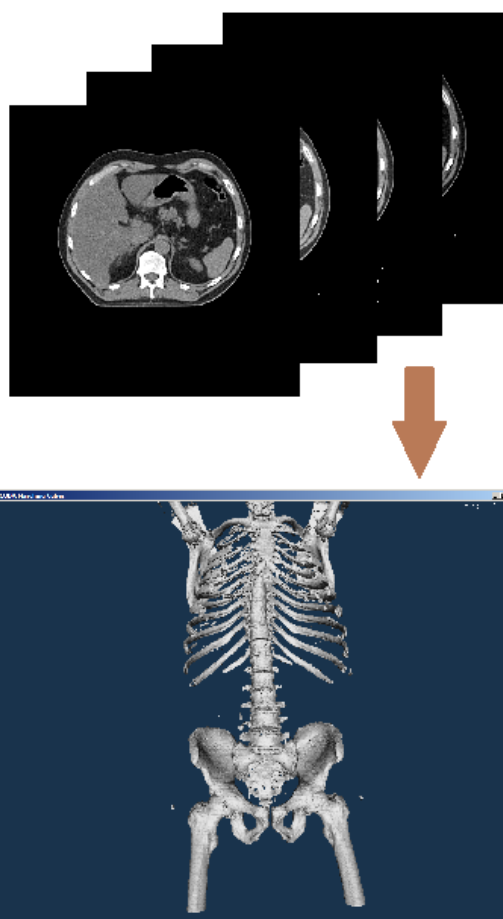


Fig. 3. 3D Reconstruction of the pelvic bone (second image) based on a CT data set (first image)

This algorithm works faster with voxel labeling, for sparse volumes in which many voxels have all vertices with the radiodensity of air (or other values that are not of interest). Using this technique the volume is traversed before running the algorithm and voxel relevancy is labeled. After this step the Marching Cubes algorithm is run only for the relevant voxels. This method yields a significant performance improvement for sparse volumes.

Marching Cubes and Marching Tetrahedrons can also be implemented in a parallel way [2], because the 3D surface is generated by computing its intersection with each of the volume voxels. Using CUDA, based on the radiodensity values of the voxel's vertices, the intersection with the surface and the generation of triangles and their normals for one voxel can be defined in a kernel that will be called for each of the voxels in the volume.

III. MOTIVATION

The main problem of 3D medical image processing is the fact that the datasets acquired from the CT or MRI devices are extremely large for volume reconstruction algorithms. One CT image usually has 512x512 pixels, and a CT data set can sometimes contain even 2048 slices.

Another problem of medical applications is that they are generally required to work in real time. Since the Marching Cubes algorithm can handle datasets of up to 512x512x64 – 1.2GB of GPU RAM required - on the current hardware, much larger datasets cannot be computed on the GPU using the original algorithm. The existing solution is to compute the algorithm on the CPU and use the much larger memory pool (the standard RAM) and the cache swapping mechanism, but this approach cannot offer real-time performance.

The parallel implementation of the 3D surface reconstruction algorithm would solve this problem, making the algorithm run in real-time through the usage of the mass parallel architecture of the GPU while minimizing the memory requirements. Since we need to handle very large volumes of up to 512x512x2048, almost 26GB of GPU RAM would be necessary to run the original GPU Marching Cubes algorithm, and this is obviously impossible on current hardware. This is the reason we propose a novel approach that will partition the initial volume into smaller sub-volumes (chunks), which can be computed serially on the GPU without exceeding the memory pool.

IV. A NOVEL APPROACH TO MARCHING CUBES

A. Algorithm

The proposed algorithm divides the data volume into maximum capacity sub-volumes, in order to be processed on the GPU. Initially, we keep on the GPU only the lookup MC tables (the edge table and the triangle table). After that, we partition the initial volume into chunks, as intuitively explained in Figure 4.

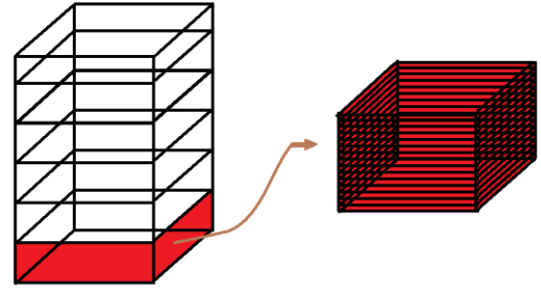


Fig. 4. Splitting the initial volume (left) into chunks

Thus, the initial volume is divided into sub-volumes on the maximum size axis, which usually is the Z axis (height). For example, if we have a CT data set of 2048 tomograms (each of 512x512 pixels), the volume is divided into 64 chunks, each with 32 slices of 512x512 pixels.

The GPGPU Marching Cubes algorithm is then applied on each of these sub-volumes, serially. The obtained results, that is, the triangle meshes, are stored in vertex buffer objects (VBOs) on the GPU. This way we obtain substantial performance gains over passing the data between CPU and GPU, due to the fact that the data (vertices, normal vectors) reside in the GPU memory rather than the system memory and can be rendered directly by the GPU.

The main advantage of this method is that it requires much less memory than the original CUDA Marching Cubes (without division of the data volume). Also, for sparse volumes, empty chunks are culled in one operation. If a chunk contains only empty voxels, it will be labeled as an irrelevant chunk at loading time. This is why the chunk will not be further processed (with voxel labeling or Marching Cubes algorithm). Therefore the speed and memory requirements offered by our method greatly outmatch those of the original CUDA Marching Cubes algorithm. Another advantage is that the division of the volume provides the possibility to reconstruct only parts of the volume. Thus, if a chunk is not of interest, it doesn't require reconstruction.

The size of the chunks can be easily changed, but we have to be careful for chunk overlapping. The data volume is represented by slices, and voxels contain vertices from two adjacent slices (four on one slice, and four on another slice). Therefore, in order to reconstruct the surface correctly with no free spaces between the surfaces reconstructed from adjacent chunks the border slices must be duplicated. As can be observed in figure 5, two adjacent chunks share a common slice, the first slice for a chunk, and the last one for the other chunk. Thus we have to choose the size of the chunks in such a manner that overlapping would not lead to very high memory usage. For example, if the chunks contain only two slices, the whole slices forming the data set will be duplicated.

Furthermore the algorithm can be specifically tailored to use different chunk sizes for different parts of the volume if

there is known that some areas are much more detailed than others.

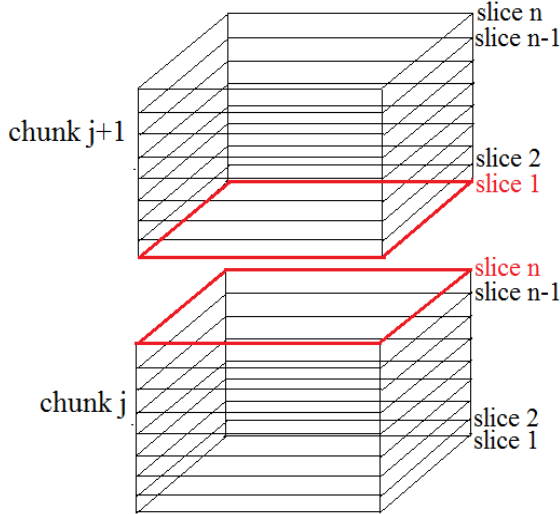


Fig. 5. Chunk overlapping. The first slice in chunk j+1(slice 1) is the same as the last slice in chunk j (slice n)

The only disadvantage of this method is that is slightly slower than the original CUDA Marching Cubes, because the chunks are computed serially. Even though in the original algorithm the CUDA driver will also serialize the process due to lack of cores, our serialization will swap volume data between CPU and GPU. Nevertheless the advantages greatly exceed this drawback.

B. Pseudocode

We present the pseudocode for this new algorithm, separated into two steps:

- The preprocessing step, executed before the modified surface reconstruction algorithm
- The reconstruction step

Preprocessing step:

1. Create textures to store the Marching Cubes tables (edge table and triangle table) on the GPU.
2. Determine the maximum size of a sub-volume (usually 32/16/8) based on the size of the GPU RAM.
3. For each sub-volume
 - 3.1. Scan every voxel
 - 3.2. If all of voxels from the sub-volume are not of interest than mark the whole sub-volume as unimportant

Reconstruction step:

4. For every sub-volume

If the sub-volume is of interest, then:

 - 4.1. Scan the sub-volume and remove the voxels from the input list of voxels for the Marching Cubes algorithm.
 - 4.2. Run the Marching Cubes algorithm with the isovalue given by the user. The reconstructed surface is represented by two vectors: one for

vertices and one for normals. The surface vectors are stored in VBOs

Else

4.3. Skip the sub-volume by completely culling it.

5. Render all the VBOs in order to draw the surface.

V. RESULTS

The implementation of the presented algorithm has been tested on a GTX 460M GPU with 1.5 GB GPU RAM. Various sizes have been chosen for the chunks, in order to compare the memory usage and draw some conclusions regarding the best choice for the chunk size. Table 1 presents the results of the runs for every size.

TABLE I
MEMORY USAGE FOR DIFFERENT CHUNK SIZES

Chunk size (number of slices)	RAM (MB)	Memory Usage (%)
32	1495	99.6
16	1440	96
8	1432	95.4
4	>1500	>100

From these results we can draw the conclusion that chunks of size 32 offer the best performance/overlapping rapport, thus being chosen for further runs. Furthermore it can be concluded that overlapping can become a major memory problem if the chunk size is too small.

For the 15 million vertices that need to be rendered, the application runs at 2.3 fps if the chunks are being reconstructed each frame. But, for faster results, it is better to reconstruct the chunks only when reconstruction is necessary, namely each time the isovalue for the isosurface is changed by the user. This way, for the same number of vertices, the application runs at 43.4 fps (which is more than 24 frames per second - the threshold for real-time applications).

Figure 6 presents the results for reconstructing a 3D surface based on a CT data set of 512x512x1024 representing the human body. The isovalue has been chosen so that the rendered isosurface would represent only bone tissue. The screenshots illustrate the reconstructed surface at different levels of detail, in order to observe that this application is relatively noise free. The accuracy of the 3D surface is limited only by the accuracy and correctness of the dataset, even though in our implementation we use filtering operations on the dataset to remove noise and improve the overall quality of the data volume that will be used by the algorithm.

Compared to the algorithm presented in [4] which is focused on handling datasets up to 77 GB but does not reconstruct the isosurface in real time, the aim of our algorithm is to reconstruct the isosurface in real time, using smaller datasets (26 GB). In our project, this allows the doctor examining the 3D model of the femur bone by viewing it from different positions (making rotations) and at different levels of detail.

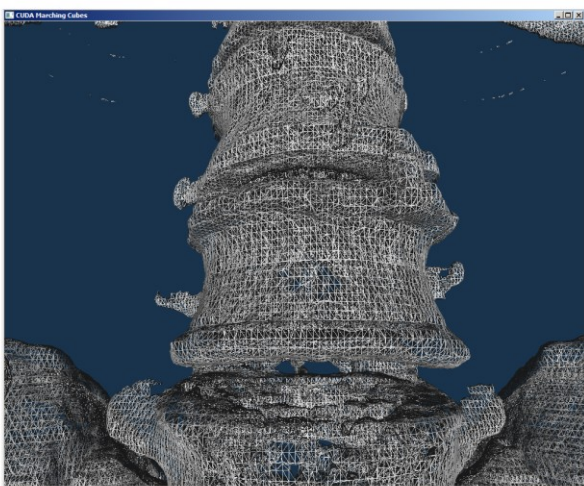
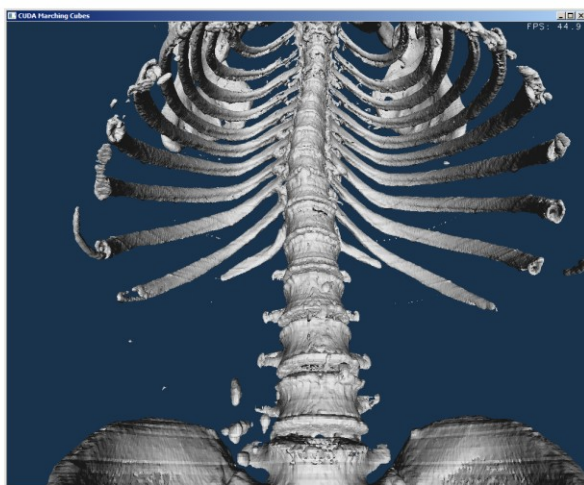
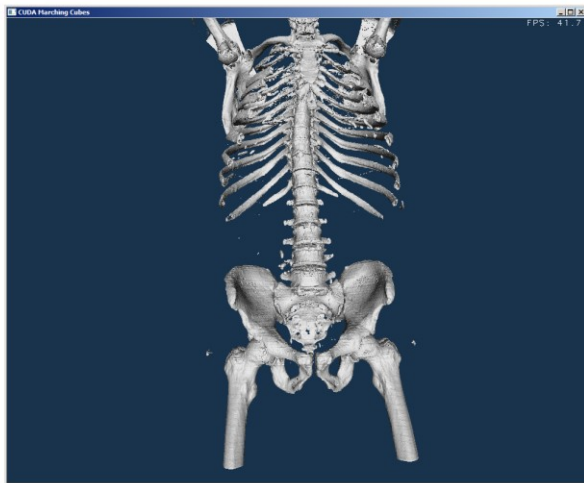


Fig. 6. 3D reconstructed surface at different levels of detail. The third image is rendered wireframe in order to show how detailed/large the dataset is.

VI. CONCLUSIONS

The proposed implementation of the Marching Cubes algorithm has proven to offer excellent memory usage and to solve the problem of memory limitedness in an optimal manner. The original CUDA implementation of the algorithm works fast enough, though it cannot handle very large volumes on the current hardware. Instead, our algorithm is slightly slower, but can reconstruct surfaces for very large volumes (512x512x2048), up to 32 times larger than the original algorithm. If the reconstruction of the isosurface is made only when the isovalue has been changed, the application runs in real time.

These results demonstrate that this variant of the algorithm represents an improvement for real-time 3D surface reconstruction applications, especially in the medical field, where the need for accuracy and speed is essential.

VII. FUTURE RESEARCH

We have implemented a real-time application that reconstructs 3D surfaces from very large data sets. This application can be used by doctors in obtaining the 3D geometry of human body parts like bones. But, based on the obtained 3D model, other operations can be made. For example, the application would provide the possibility to alter the surface, or make some automated measurements.

In our future research in 3D medical imaging we will try to implement a medical application that will help doctors in the hip arthroplasty domain. Our aim is to automatically generate prostheses (without any human intervention), based on the reconstructed surfaces (the bones) extracted from CT data sets. This task implies the research topic of metal detection in CTs.

We will also focus on finding a good tessellation algorithm in order to add more geometry to the reconstructed isosurfaces. An adaptive tessellation algorithm that depends on the density of the tessellated surface could more correctly depict the irregularities found in nature that would otherwise need extremely large data volumes to be represented.

REFERENCES

- [1] W. E. Lorensen, H. E. Cline, "Marching Cubes: A high resolution 3D surface reconstruction algorithm", *SIGGRAPH '87 Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, 1987, USA
- [2] Y. Heng, L. Gu, "GPU-based volume rendering for medical image visualization", *Proceedings of the IEEE Engineering in Medicine and Biology 27th Annual Conference*, 2005, Shanghai, China
- [3] S. Martin, H. Shen, P. McCormick, "Load-balanced isosurfacing on multi-GPU clusters", *Eurographics Symposium on Parallel Graphics and Visualization*, 2010
- [4] H.R. Nagel, "GPU optimized Marching Cubes algorithm for handling very large, temporal datasets", *CiteSeerX – Scientific Literature Digital Library and Search Engine*, 2010
- [5] B. Preim, D. Bratz, *Visualization in medicine*. Elsevier, Morgan Kaufmann Publishers, 2007, USA
- [6] "NVIDIA CUDA C – Programming guide". Version 3.2, 2010, NVIDIA Corporation

Learning strategies for wet clutch control

Gregory Pinte, Julian Stoev, Wim Symens,
Abhishek Dutta, Yu Zhong, Bart Wyns, Robin De Keyser,
Bruno Depraetere, Jan Swevers,
Matteo Gagliolo and Ann Nowé

Abstract—This paper presents an overview of model-based (Iterative Learning Control, Model Predictive Control and Iterative Optimization) and non-model-based (Genetic-based Machine Learning and Reinforcement Learning) learning strategies for the control of wet clutches. Based on theoretical considerations and a validation on an experimental test bench containing wet clutches, the benefits and drawbacks of the different strategies are compared. Although after convergence a good engagement quality can be obtained by all strategies, only model-based strategies are suited for online applicability. The convergence time for non-model-based strategies is too long such that they can only be applied during an offline calibration phase.

I. INTRODUCTION

A wet clutch is a device to transmit torque from one shaft to another by means of friction. As illustrated in Fig. 1, it contains two sets of friction plates, one that can slide in grooves on the inside of the drum, and another that can slide in grooves on the outgoing shaft. Torque can be transferred between the shafts by pressing both sets together with a hydraulic piston, which can be realized by sending an appropriate control signal to the servovalve in the line to the clutch. Initially, during the filling phase, the clutch chamber fills up with oil and the pressure builds up, until it is high enough to compress the return spring and accelerate the piston. When the piston advances far enough and presses the plates together, the filling phase ends and the slip phase begins. During the slip phase torque is transferred, causing the difference in rotation speeds between the shafts, called the slip, to decrease, until both shafts have the same rotation speed.

Wet clutches are commonly used in automatic transmissions for off-highway vehicles and agricultural machines to transfer torque from the engine to the load. By disengaging one clutch and engaging another, different transmission ratios can be realized. When a clutch engagement is requested, an

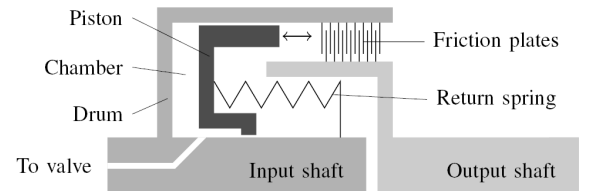


Fig. 1. Schematic representation of a wet clutch and its components.

operator expects a fast response without vibrations. Torque transfer should thus commence as soon as possible without introducing torque discontinuities and peaks, which can be realized by a short filling phase followed by a smooth transition into the slip phase. The two main challenges for wet clutch control are (i) the complex, non-linear behavior, with a sudden change in dynamics when the piston comes into contact with the plates [1], and (ii) the variation of these dynamics over time due to changes in oil temperature and wear [2].

When similar or repetitive operations have to be carried out such as the successive engagements of a clutch, learning can be introduced to address these issues. By gradually improving the performance with respect to the previous trial, the complex system behaviour can be learned at the cost of a convergence period. It also becomes possible to operate optimally during normal machine operation, as the learning algorithms will automatically correct for variations in the system behavior.

In this paper, the potential of several model-based (Model Predictive Control (MPC), Iterative Learning Control (ILC) and Iterative Optimization) and non-model based learning techniques (Genetic-based Machine Learning (GBML) and Reinforcement Learning (RL)) is investigated for the control of a wet clutch engagement. All the model-based approaches rely on a model of the clutch dynamics to update the control signals at each engagement. On the contrary, the non-model based approaches explore the input space of possible clutch control signals by trial and error. The performance and robustness of all the different learning strategies have been validated on an experimental test setup, where an electromotor (30kW) drives a flywheel (2.5 kgm²) via a torque converter and two mechanical transmissions (Fig. 2). The developed controllers have been tested on the first range clutch of the left transmission while the right transmission is used to vary the load observed by the first transmission and

This work has been carried out within the framework of IWT-SBO project 80032 (LECOPRO) of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen)

G. Pinte, J. Stoev and Wim Symens are with the Flanders' Mechatronics Technology Centre, 3001 Heverlee, Belgium gregory.pinte@fmtc.be

A. Dutta, Y. Zhong, B. Wyns and R. De Keyser are with the Electrical Energy, Systems and Automation group, Ghent University, 9052 Zwijnaarde, Belgium,

B. Depraetere and J. Swevers are with the Department of Mechanical Engineering, Division PMA, Katholieke Universiteit Leuven, 3001 Heverlee, Belgium

M. Gagliolo and A. Nowé are with the Department of Computer Science, Vrije Universiteit Brussel, 1050 Brussels, Belgium

to apply an adjustable braking torque. The transmission to be controlled is equipped with several sensors, measuring the speeds of the different shafts and the pressure of the oil in the line to the different clutches. All experiments are performed with a fixed engine speed, while the output always starts at standstill and is then accelerated by engaging the clutch for first gear in the controlled transmission. A dSPACE 1103 board is used to control the setup.

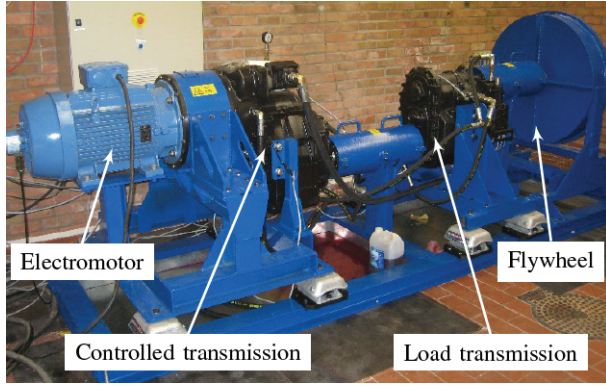


Fig. 2. Test setup: (from left to right) electromotor, controlled transmission, torque sensor, load transmission and flywheel.

The remaining content of this paper is laid out as follows. Sections II and III introduce respectively the model-based and non-model-based learning techniques and illustrate the practical applicability of each technique for clutch engagement control. The benefits and drawbacks of the different methods are compared in Section IV. Section V concludes the paper.

II. MODEL-BASED LEARNING CONTROL

This section discusses three model-based learning techniques for wet clutch control. First a two-level learning control scheme based on MPC is presented. Afterwards, an ILC controller is presented as an alternative for the MPC controller in this scheme. Finally, a novel iterative optimization scheme is discussed, which solves certain limitations of the other model-based techniques.

A. Model Predictive Control

MPC is an advanced control technique that computes the optimal system input online by solving an optimization problem and taking into account the system constraints. When the optimal input is determined, the first part of this optimal input is applied to the system. The next sampling time, the whole procedure is repeated, thereby introducing system feedback. As the MPC controller has to solve an optimization problem every sampling time, MPC became popular in the process and chemical industry, where sampling periods in the order of seconds or minutes allow sufficient computation time. Thanks to the increase in computation power and the development of more advanced algorithms, MPC is nowadays starting to become also applicable to mechatronic applications with faster dynamics.

Instead of developing a single, computationally more demanding nonlinear MPC algorithm for the control of a complete clutch engagement control including the hard nonlinearity between the filling and the slip phase, the idea is to apply two separate linear MPC algorithms. The first controller aims to track a pressure reference in the filling phase. It is deactivated once the slip phase begins, and the second controller is activated to track a slip reference. This way, each MPC controller acts in a relatively linear operating range. Constraints on the current to the servovalve and on the pressure can be taken into account in the MPC algorithm. As the availability of an accurate model of the controlled system is a requirement for the proper functioning of an MPC controller, an offline identification of the controlled plant between the current to the servovalve and respectively the pressure and the slip is required. Using separate controllers for the filling and slip phases makes identification easier, as it now suffices to develop a model for each phase separately instead of a global strongly nonlinear model.

MPC requires the availability of suitable reference trajectories to be tracked. In the case of clutch control, the trajectories are initially however unknown, as it is difficult to translate the specifications that an engagement has to be smooth and fast directly into suitable pressure and slip profiles. Furthermore the optimal reference trajectories vary with the operating conditions. Therefore a two-level control scheme (Fig. 3) is proposed, where good reference (pressure and slip) trajectories are learned at the high level and the MPC controllers aim to track these reference trajectories at the low level. For the high level reference adaptation, process knowledge is used to define the shape of suitable parameterized reference profiles. The obtained performance is assessed after each engagement and used by the high level controller to update the parameters of the reference profiles.

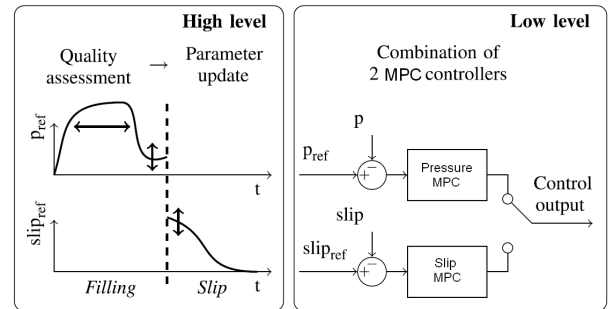


Fig. 3. The presented two-level control scheme: a high level controller updates reference trajectories to be tracked by low level MPC controllers.

For example, the reference pressure profile (Fig. 4) consists of three parts, with (i) a high pressure value to accelerate the piston, (ii) a strong pressure reduction to slow down the piston, and (iii) a small increase in pressure to gently press the piston into the plates. After each engagement, the duration of the high pressure phase w and the pressure at the end of the reference p_{end} are updated based on an assessment of the engagement quality determined by two performance indices $PI_{1,est}$ and $PI_{2,est}$. These performance indices are

derived from the output speed profile. The following update law is for example used for p_{end} :

$$p_{end} = p_{end,prev} + \lambda_1 \zeta (PI_{1,est} - PI_{1,ideal}), \quad (1)$$

where ζ is a dimensionless saturated function, with a profile as shown in Fig. 5, λ_1 is the updating gain and $PI_{1,est}$ is an assessment of the engagement smoothness and $PI_{1,ideal}$ is the desired smoothness. The profile of ζ is chosen to limit the correction on p_{end} and the changes in the reference signal. Small variations on p_{end} are avoided when $PI_{1,est}$ is close enough to $PI_{1,ideal}$ such that the references are not needlessly adapted.

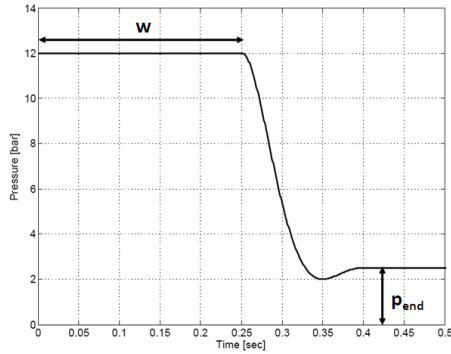


Fig. 4. A suitable reference trajectory for the pressure during a clutch engagement.

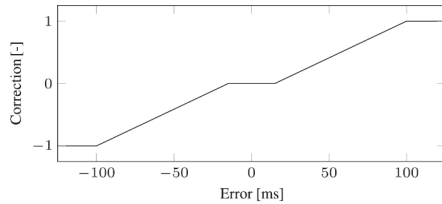


Fig. 5. Dimensionless learning function ζ of the high level controller.

A detailed description of the applied performance indices and the update law can be found in [3].

B. Iterative Learning Control

In this paragraph, ILC is presented as a valuable alternative for MPC at the low level of the developed two-level control scheme in Fig. 3. The main idea in ILC is to improve the tracking performance of repetitive systems using the results of previous iterations. While for MPC an accurate model determines the achievable tracking performance, an ILC algorithm can realize a good tracking accuracy despite a large model uncertainty due to its learning behaviour. As a consequence, an accurate plant model should not be identified, which is an advantage over MPC that can be exploited in the control of wet clutch engagements, where the plant dynamics are non-linear and vary significantly over time. For the ILC design, a single, linearized model approximating the plant dynamics in all conditions suffices, keeping the required modeling effort very small.

In this case, the high level controller does not update the reference profile every engagement but only after a certain number of engagements N when the low-level ILC algorithms have converged. At this moment, the performance can be assessed and is used by the high level controller to update the parameters of the reference profiles. The following algorithmic description summarizes the learning process:

```

While true do
    apply control input to system and measure output
    use ILC algorithm to calculate new control input
    if  $\text{mod}(i, N) = 0$  then
        evaluate performance
        update parameters and calculate new reference
    end if
     $i++$ 
end while

```

While MPC is actually a feedback control technique hereby introducing robustness to variations in the clutch dynamics from engagement to engagement, ILC is an open loop control technique which assumes the same controlled plant at each engagement. However in practice, the clutch dynamics will change significantly in the case of load and temperature variations. Furthermore, when the operating conditions change, the references also have to be adapted to maintain performance. When for example the load is changed, the amount of torque and hence the pressure needed to realize a change of slip changes as well. Similar observations hold for variations in the oil temperature. In order to cope with this problem the range of operating conditions is divided into several subranges, each corresponding to a limited interval of load and oil temperature. The described control strategy is then applied to each subrange separately.

C. Iterative Optimization

As mentioned in the previous two paragraphs, classical control techniques such as MPC and ILC focus on tracking control, and are of little use when no appropriate reference trajectories are available. A possible approach to overcome this restriction is the addition of a high-level layer where suitable reference signals are determined. In this paragraph an alternative approach is presented which uses the performance specifications directly in order to formulate a numerical optimization problem. Before each engagement, this problem is solved to determine the control signal, which is then applied to the system. The measured response is used to adapt the models and constraints in the optimization problem before the next control signal is calculated. This results in a two level control structure, as illustrated in Fig. 6 [4], with the optimization procedure on the low level, being fed with models and constraints by recursive estimators and learning laws on the high level.

In order to apply the proposed iterative optimization approach to a wet clutch engagement, a numerical optimization problem has to be defined. The goal is to minimize the time required before the load is synchronized, without exceeding a given value of the jerk and without violating any other

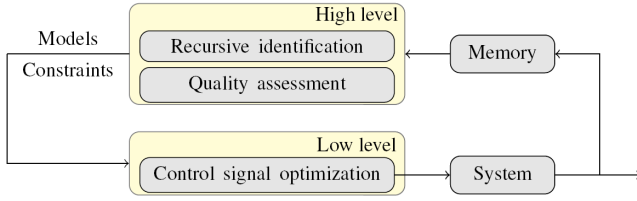


Fig. 6. Two-level iterative optimization scheme. At the high level, the models and constraints for the optimization problem are updated after each engagement. At the low level, these are used to optimize the control signal for the next engagement.

constraints (e.g. power dissipation to limit heat dissipation in the clutch). For the solution of this non-convex time-optimal control problem, a bisection algorithm can be used to calculate the minimum engagement time, for which all constraints are fulfilled [4].

Models need to be supplied to the low level optimization problem, such that the output variables can be predicted for a given control input. To avoid having to derive complex and accurate models a priori such as in the case of an MPC or ILC design, a recursive estimation scheme is employed. After each engagement the batch of measured data is used to update the model parameters, while also taking into the account the previous models. The estimation is performed using data measured online, using real control signals. As a result, simplified models with limited ranges of validity can be tuned to predict the behavior precisely for these types of control signals, and yield a good prediction accuracy. If needed, a piecewise structure can be used, using a succession of simplified models instead of a single more complex one. For the clutch control, analogously as in the MPC and ILC design, two linear models are identified respectively for the pressure (in the filling phase) and the slip (in the slip phase), as sensors are available to compare the predicted variables with the measured ones.

Next to the model updating, at the high level of the iterative optimization scheme a learning law updates two constraints based on an assessment of the observed engagement quality. These constraints fix the estimated piston position ($1/\gamma$) and the pressure (p_{end}) at the end of the filling phase, which guarantee a smooth transition into the slip phase. The updating laws for these constraints are similar to the law in (1), which are used to adapt the parameters of the reference signals for the ILC and MPC references.

D. Experimental results

The three above mentioned techniques have been experimentally tested on the set-up in Fig. 2.

While an extensive experimental validation has been performed for the ILC and iterative optimization algorithms, up till now only preliminary test results are available on the application of the MPC for wet clutch control. The low-level MPC controller for pressure tracking in the filling phase has been implemented successfully. A practical implementation and validation of the slip MPC control algorithm and the high-level adaptation algorithm still has to be performed.

The two-level ILC scheme as discussed in paragraph II-B is tested at 40°C with a load inertia fixed at 8.4 kg m². The parameter values for the reference trajectories are initialized poorly to illustrate the convergence process. Then the learning starts and these parameters are adapted every 5 engagements, with their evolution shown in Fig. 7. The measured pressure and torque signals are shown in Fig. 8. They show the results during the 5th, 10th and 30th engagement to illustrate the increasing performance as the reference trajectories are adapted and the low level controllers ensure that they are tracked accurately. During the fifth engagement (black lines in Fig. 8) the ILC controllers have converged towards the initial references. The performance however is very poor with a high torque peak due to an initial overfilling, resulting in an uncomfortable engagement for an operator. After the fifth engagement, the high level controller reacts by reducing w , as shown in Fig. 7. During the 10th engagement (grey lines in Fig. 8) these new reference trajectories are tracked accurately and the performance has already significantly improved. Over the course of the next engagements, the parameters for the pressure reference trajectory converge further, and by the 30th engagement they remain unaltered, as shown in Fig. 7. This results in a fast and smooth engagement. Other tests at 70°C and with a load of 28.6 kg m² lead to similar results, demonstrating the robustness of the developed algorithm to variations in operating conditions.

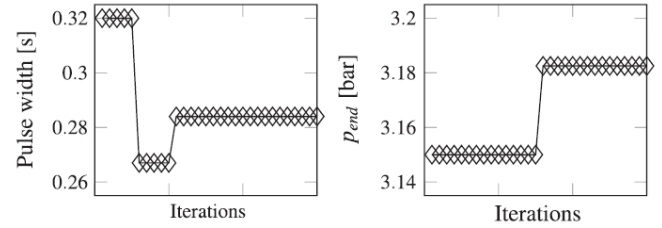


Fig. 7. Results for the two-level ILC scheme: Evolution of the reference signal parameters.

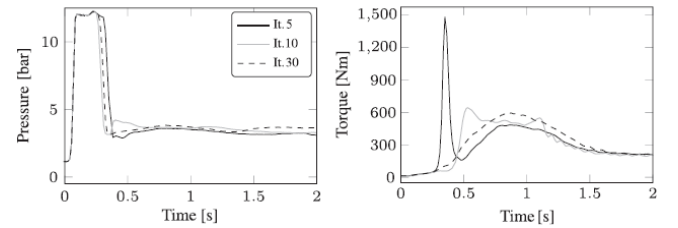


Fig. 8. Results for the two-level ILC scheme: the pressure and torque during iterations 5 (black), 10 (grey) and 30 (dashed).

Similar experiments in the same operational conditions have been carried out for the validation of the iterative optimization scheme. The control is initialized with suboptimal values for the optimization constraints p_{end} and γ . For the recursive estimators a feedforward engagement is performed, such that a first estimation of the model parameters for the applied models can be made. Using these newly estimated models and the initial values for the optimization constraints,

the first control signal is optimized and applied to the system. The results obtained after applying this first control signal are shown by the black lines in Fig. 9, where from left to right the pressure and torque are shown. Torque rapidly increases to a high initial value, but then dips before it increases again. As a result, the slip (Fig. 11) changes abruptly yielding a high absolute value of the jerk, and hence discomfort for the operator. Afterwards, the clutch opens again as the pressure is too low and hence the torque is insufficient to synchronize the load. Over the course of the following engagements the high level controller compensates for this by re-estimating the models based on more representative data and by adapting the optimization constraints. The improved model accuracy after 10 engagements is illustrated in Fig. 11, where the deviations between the measured (solid) and predicted (dashed) values are reduced significantly compared to the first engagement. The updating of the optimization constraints is illustrated in Fig. 10. As a result the performance of the clutch engagement improves. This is illustrated by the grey lines in Fig. 9, showing the performance during the 3rd iteration. The torque peak has almost been removed. The dashed lines show the results after 10 iterations, when the learning process has converged. A good engagement quality is now obtained, with smooth profiles for both the slip and torque, ensuring operator comfort without reducing the responsiveness of the system. The robustness of the iterative optimization algorithm has also been validated successfully by tests at other operating conditions.

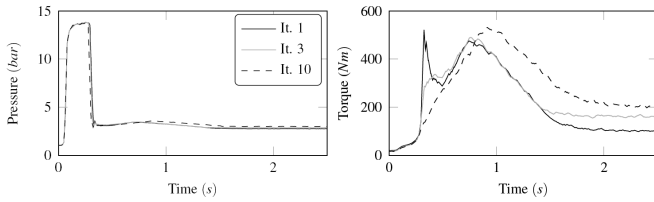


Fig. 9. Results for the two-level iterative optimization scheme: the pressure and torque during iterations 1 (black), 3 (grey) and 10 (dashed).

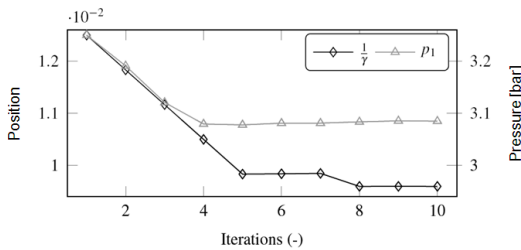


Fig. 10. Results for the two-level iterative optimization scheme: Evolution of the constraints p_{end} and $1/\gamma$.

Compared to the two-level ILC control scheme, the iterative optimization approach can significantly reduce the convergence time before an optimal engagement quality is obtained. Due to the convergence of the ILC algorithm, the high-level laws in the two-level ILC scheme can only be updated every 5 engagement while these laws can be updated

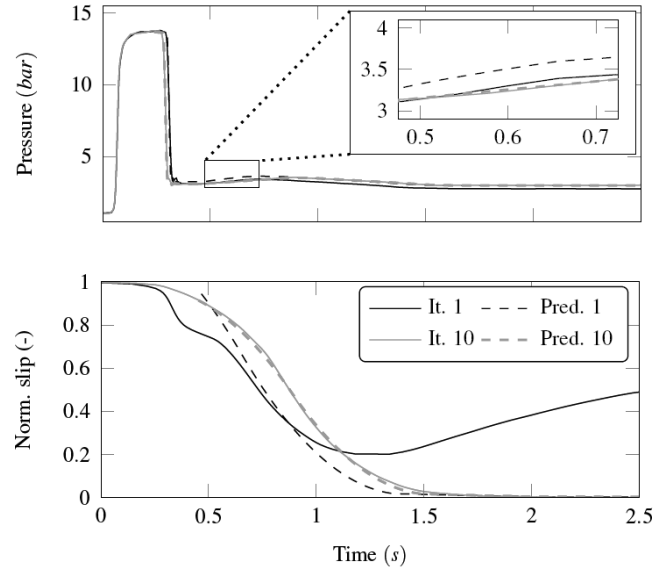


Fig. 11. Results for the two-level iterative optimization scheme: Increasing prediction accuracy as a function of iterations. The top figure shows the measured and predicted pressures during the 1st (dashed black) and 10th iteration (solid black). The bottom figure shows the same for the measured and predicted slips.

each engagement in the iterative optimization scheme. Next to this lower convergence time, a second significant benefit of the iterative optimization approach is that the offline identification of system models is avoided, since these models are recursively identified online.

III. NON-MODEL-BASED LEARNING CONTROL

Next to model-based learning algorithms, the potential of non-model-based algorithms for wet clutch control has also been investigated.

A. Genetic-Algorithm based Machine Learning

Genetic Algorithms (GAs) are generally used as an optimization technique to search the global optimum of a function. However, this is not the only possible use for GAs. Other fields of applications where robustness and global optimization are needed could also benefit greatly from the use of GAs. The two most important domains using GAs as the underlying methodology are Genetic Based Machine Learning (GBML) [5] and Genetic Programming [6]. There has been a growing interest in applying evolution programming techniques to machine learning. This was due to the attractive idea that chromosomes, representing knowledge, are treated as data to be manipulated by genetic operators, and, at the same time, as executable code to be used in performing some task. Next to GA and GP also a learning classifier system, or LCS, is very popular to solve engineering problems. An LCS is a rule-based machine learning system with close links to reinforcement learning and genetic algorithms. The first concept was described by John Holland in [7], and his LCS used a genetic algorithm to alter and select the best rules among a population of binary rules. Classifier systems are a special kind of production systems [8]. Initially the classifiers

or rules were binary, but recent research has expanded this representation to include real-valued, neural network, and functional (S-expression) conditions.

During a wet clutch engagement, two conflicting requirements are demanded by the machine operators. On the one hand a smooth engagement is requested while on the other hand the engagement should be fast. In contrast with the other presented learning approaches, which combine these two requirements in a single objective (possible with additional constraints), the improved genetic-based LCS is especially suitable to solve such a multi-objective problem. It introduces the NSGA-II as the rule discovering mechanism, so that the optimized classifier will find the Pareto front representing the optimal trade-off between the engagement smoothness and duration and the corresponding control signals. Moreover, since the required population size for a multi-objective problem cannot be very small, the number of classifiers will also be large corresponding to the population size. Meanwhile, because of the difficulty of selecting the niching parameters, it is very hard to make sure that the individuals will be evenly distributed along the Pareto front. This will lead to the situation that a group of classifiers will offer a similar engagement time and smoothness since they are too close to each other. Thus, after the Pareto front is formed, first the fuzzy clustering method (fuzzy C-mean method) will be used to cluster the whole front into desired zones/pools, then the reduced population size genetic algorithm will generate respective individuals for each zone/pool. This will significantly reduce the number of classifiers in the system without losing the information of the Pareto front.

B. Reinforcement Learning

As mentioned above already, during the engagement of a wet clutch, the control objective in these systems is not to track a complete (pressure or slip) trajectory accurately, but rather to achieve a given state at a certain location (e.g. to limit the piston velocity when it gets into contact with the friction plates). Model-based control strategies are not suited for the control of these systems, due to the lack of sensor data. The non-model-based RL strategy is naturally more suited in dealing with this kind of discrete sensor information.

RL problems [9] are a class of machine learning problems, where an agent must learn to interact with an unknown environment, using a trial and error approach. At a given timestep t , the agent may execute one of a set of actions, possibly causing the environment to change its state and generate a (scalar) reward. Both state and action spaces can be multidimensional, continuous or discrete. An agent is represented by a policy, mapping states to actions. The aim of a RL algorithm is to optimize the policy, maximizing the reward accumulated by the agent.

In the case of wet clutch control, an open loop control policy is being developed, where the same parameterized profile as considered in the GBML approach is learned. The RL problem explores the parameter space based on the

reward, which is a combination of the engagement time and the jerk as a parameter for the engagement smoothness. On the longer term also closed loop control policies will be investigated.

C. Experimental results

Nowadays, wet clutches in industrial transmissions are filled using a feed forward controller of the current to the electro-hydraulic valve, which regulates the oil pressure and hence the piston position in the clutch. Fig. 12 shows a typical parameterized, feed forward current signal, which is sent in the filling phase to the valve. Although nowadays more advanced feed forward signals with more tunable parameters are sometimes applied, the above mentioned parameterization perfectly illustrates the underlying idea behind the actual industrial control design. Both above discussed non-model-based learning approaches use this parameterization and aim to find the optimal parameters of this profiles.

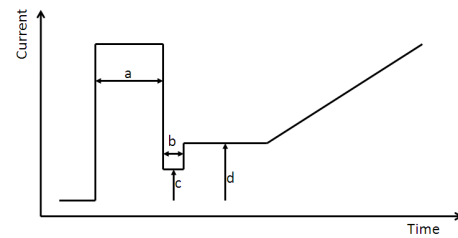


Fig. 12. A typical parameterized, feed forward current signal for clutch engagement control.

Unlike the conventional model based control techniques, the genetic based learning classifier system starts with a randomized population of different control signals instead of a single control signal (Fig. 13), and use evolutionary programming to explore the solution space. After accumulating individuals in regions of high fitness value, the population will converge after 10 generations (i.e. approximately 500 engagements) to the (or one of the) global maximum of this fitness landscape and form the optimized classifiers. Fig. 13 clearly shows that most of the individuals converge to one curve, which is the Pareto Front for the multi-objective wet clutch control problem. The control signals corresponding to one side of this front will result in slow but smooth engagements, while control signals corresponding to the other side of the front will lead to fast but more brutal engagements.

In order to figure out the Pareto front, all the individuals evaluated during the GA process are gathered together for a further sorting. The first four fronts are kept since even some identical profiles were given to the setup, the measurement can vary a little bit for every test run. Technically, the individuals remaining in the pool can all be thought of as different possible control signals, but for practical use a further generalization is necessary. First of all, to keep only the practically applicable control signals, individuals who lead to very large torque losses and very long engagements are dropped. Afterwards, a fuzzy clustering method (fuzzy

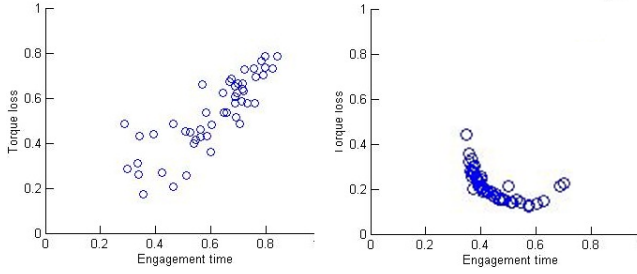


Fig. 13. The initial population and the population in the 10th generation.

C-means clustering) is used to identify the natural groupings of data from a large data set to produce a concise representation of the clutch behavior. In this case, it is reasonable to generate three characteristic classifiers, the first one representing a fast engagement with large torque loss (in stars), the second one stands for a median engagement time with median torque loss (in cycles), and the last one is slow engagement with small torque loss (in crosses). The result from fuzzy clustering is shown in Fig. 14.

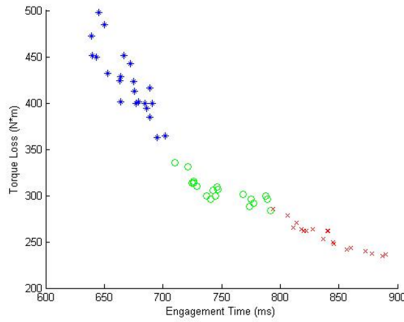


Fig. 14. Fuzzy clustering of the elements in the 10th generation.

The reduced population size genetic algorithm mentioned before is then used to generate the divided zones/pools into a single classifier. The algorithm will stop when the population size of each zone/pool is reduced to one.

Up till now, the potential of RL has only been validated in simulation, but no practical tests have been performed yet.

IV. COMPARISON BETWEEN MODEL-BASED AND NON-MODEL-BASED LEARNING CONTROL

This section compares the benefits and drawbacks of model-based and non-model-based learning algorithms for wet clutch control.

At first sight the application of non-model-based algorithms seems very attractive for the control of a wet clutch engagement, since the huge identification effort can be avoided for modeling the complex, non-linear behavior of the clutch dynamics. Due to the lack of a model in these algorithms however, a long exploration phase is required to check the applicability of all possible control signals. To restrict the input space of possible control signals and in this way limit the exploration, the non-model-based algorithms

use a parameterization of the control signal with limitations on the signal parameters. Although this parameterization is typically based on physical system knowledge, it is practically impossible to select beforehand without system modeling the shape of the control signal that will lead to the optimal performance. In this way, restrictions are inherently imposed on the achievable performance of the non-model-based learning approaches. In the case of clutch control, the parameterized control signal in Fig. 12 will even with optimally selected parameters result in a suboptimal engagement. Contrary to this, the model-based learning algorithms leave complete freedom in the determination of the shape of the control signals such that the optimal control signals can be found ultimately. In many applications however, the suboptimal performance obtained by non-model-based parameterized control will be acceptable since the deviation from the optimal performance will be limited.

Furthermore, even if the input space of control signals is limited, the exploration phase of the non-model-based algorithms lasts much longer than the convergence time required by the model-based algorithms. For example, while 500 engagements are required before the optimal Pareto front is found by the GA-based ML algorithm, convergence to the optimal control signal is obtained by the model-based algorithms after less than 30 engagements. It is clear that the incorporation of model information significantly speeds up the learning phase of the algorithms. The unavoidable long exploration phase of non-model-based algorithms makes them only applicable in the (re)calibration phase of clutches. Compared to the currently applied random search methods, the structured exploration of the control input space in the presented non-model-based algorithms offers a possibility to limit calibration time.

As a consequence, the model-based approaches are the only online implementable methods for wet clutch control. Amongst these methods, the iterative optimization approach seems at the moment the most promising, as the specification of the user expectations in the optimization algorithm is straightforward and the achieved convergence time is limited. The experimental results prove the practical applicability of this approach. Compared to the MPC and ILC methods, the computational demand to solve the optimization algorithm will be higher. For clutch control, this is typically not an issue, since sufficient calculation time is available between two successive engagements. The main difficulty in the definition of the iterative optimization algorithm and the MPC and ILC two-level control scheme, is the derivation of the high-level heuristic updating laws of respectively the constraints and the reference signals. These laws, which try to relate the engagement performance to certain measurable system parameters, are based on physical knowledge and are not always straightforward to define. At this high level, the potential benefits of the usage of non-model-based algorithms could be explored.

V. CONCLUSIONS AND FUTURE WORK

Since experimental results show that the convergence time for non-model-based learning approaches is very high, only model-based learning techniques will be applicable in practise for the online control of wet clutches. Non-model-based techniques could however be used to decrease the offline calibration time of the controllers for these clutches. While the incorporation of available model information in learning control algorithms seems unavoidable to limit the exploration time, certain aspects, which are difficult / not possible to model due to a lack of sensors, cannot be tackled by model-based algorithms. At the moment, these aspects are tackled by physically inspired, heuristic laws on top of the model-based control algorithms. Non-model-based approaches could be an alternative for these heuristic laws. Therefore, in the future, it is worthwhile to investigate how non-model based approaches could be integrated at the top level in the developed model-based control schemes.

Other future work consists of a further practical validation of the MPC and RL control scheme. Furthermore, the different model-based and non-model-based learning approaches will be applied to the global control of gearshifts, where two clutches are controlled simultaneously, such that one engages while another disengages.

REFERENCES

- [1] M. Montanari, F. Ronchi, C. Rossi, A. Tilli and A. Tonielli, Control and performance evaluation of a clutch servo system with hydraulic actuation, *Control Eng. Practice*, vol. 12(11), 2004, pp 1369-79.
- [2] M. Watson, C. Byington, D. Edwards and S. Amin, Dynamic modeling and wear-based remaining useful life prediction of high power clutch systems, *Tribol. Trans.*, vol. 48(2), 2005, pp 208-17.
- [3] B. Depraetere, G. Pinte, W. Symens and J. Swevers, A two-level Iterative Learning Control scheme for the engagement of wet clutches, *Mechatronics*, vol. 21, 2011, pp 501-8.
- [4] B. Depraetere, G. Pinte, and J. Swevers, A reference free iterative learning strategy for wet clutch control, in *Proceedings of the American Control Conference 2011*, 2011.
- [5] D.E. Goldberg, *Genetic algorithms in search, optimisation, and machine learning*, Addison-Wesley, 1989.
- [6] J.R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, MA, USA, 1992.
- [7] J.H., Holland, *Adaptation in Natural and Artificial Systems*. University Press of Michigan, Ann Arbor, 1975.
- [8] R. Davis and J. King, An Overview of Production Systems. *Machine Intelligence*, vol. 8, 1976, pp. 300-332.
- [9] R. Sutton and A. Barto, *Reinforcement learning: An introduction*. Cambridge, MA, MIT Press, 1998.

NUMERICAL CONTROL FOR PETROCHEMICAL INSTALLATIONS

Dumitru Popescu, Mădălina Mircioiu, Adrian Moise

University "Politehnica" of Bucharest
Faculty of Automatic Control and Computer Engineering
Splaiul Independentei 313, Bucuresti
Cod Postal 060042, ROMANIA

Abstract: The main objective of this work is the design and implementation of a hierarchical control solution for chemical and petrochemical processes, namely the control and optimization of the ethylene pyrolysis reactor, the key- installation in the petrochemical industry. The hierarchical control structures are organized on two levels of automation: the execution level, which has to fulfill two important tasks: the data acquisition from the physical plant and the regulation of the major parameters of the process and the supervisory level, referred to the optimization of the production.

Keywords: ethylene pyrolysis, hierarchical control, optimization, multivariable nonlinear structure.

1. INTRODUCTION

During the past 20 years, the chemical and petrochemical industries have faced major challenges due to global competition and rapid changes in economic conditions. Advanced automation, process control, and on- line optimization have played key roles in improving plant performance. The chemical and petrochemical industry has experienced an unprecedented development by upgrading the equipment and the production lines and also, by the expansion of production capacity. Petrochemical industry is an important provider of products for population and industry (food, pharmaceutical, mechanic, electronic, textiles and so on) and remains a priority for the Romanian economy.

Ethylene, the simplest of olefins, is used as a base product for many syntheses in the petrochemical industry: plastics, solvents, cosmetics, pneumatics, paints, etc. Today, the demand for ethylene is over 125 million tons per year with a growth rate of 3.5 % per year. Between 1995 and 2005, there has been a very significant shift in the pattern of global ethylene manufacturing, a trend that is continuing.

The ethylene production process is one of the most important aspects of a petrochemical plant and the cracking reactor is the heart of the process. Since, ethylene is one of the raw materials in the chemical industry and the market situation of not only the feed and the product, but also the utility is rapidly changing, the optimal operation and control of the plant is important.

If in the past the main purpose of a control solution was the stability of the system, in recent years the modern control strategies of the large scale processes focus on the efficiency and productivity of industrial plants. They must enhance product quality, master the whole range of products, improve process safety and plant availability, efficiently utilize resources and also, lower emissions.

A method for the practical implementation of an on-line optimization and control scheme for an industrial process considers the overall design as a two layer hierarchical system.

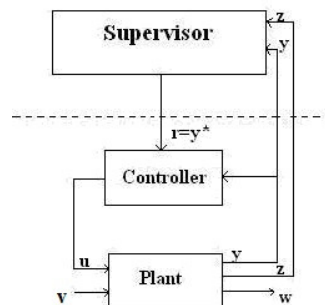


Fig.1.1 .Hierarchical control schematic

In a normal technological evolution, the plant P functions in a nominal mode. This is assured by the control systems from the lower level, considered as the execution level. The supervisor must develop an efficient control decision so that the process has an optimal operating mode, by

imposing and respecting a quality criterion. In other words, the optimum control strategy is determined on this level based on the on-line, real time information to achieve the control command by adjusting the manipulated variables or set points at the execution level.

The decision level has a slow dynamics because the changes in the global evolution of the process depend on the temporary changes of the perturbation v . Contrary to this, the dynamics of the execution level is faster because of the temporary operating mode of the control systems.

In the first stage, the process model must be evaluated by an identification procedure. In general, this model is multivariable and nonlinear.

In another stage, the objective is to define and resolve an optimization problem. The process model is used to construct the quality criterion $J(y)$. The solution of the problem:

$$\max_{y \in D} J(y),$$

(where D is the admissible domain of the parameters variation), represents the optimal decision.

$$r^* = y^* = \arg \max_{y \in D} J(y).$$

2. TECHNOLOGICAL ASPECTS

In steam cracking, a gaseous or liquid hydrocarbon feed like naphtha, LPG or ethane is diluted with steam and briefly heated in a reactor without the presence of oxygen. Typically, the reaction temperature is very high, at around 850°C , the pressure is low (4 bar), but the reaction is only allowed to take place very briefly. In modern cracking furnaces, the residence time is even reduced to milliseconds, resulting in gas velocities faster than the speed of sound, to improve yield. After the cracking temperature has been reached, the gas is quickly quenched to stop the reaction in a transfer line heat exchanger.

As a general rule, a steam cracking furnace uses radiant heat delivered by burners such as conventional gas or oil burners as its source of heat. The burners are often placed on the floor and/or the walls of the furnace and they define a high temperature zone in the furnace, also referred to as the "radiation" zone of the furnace. Immediately above said zone, there is a convection zone through which the hot combustion gases escape from the radiation zone, which convection zone is generally used for preheating the mixture of hydrocarbons to be cracked, also known as the "feedstock" to be cracked. Thus, the mixture is usually preheated to about 500°C in the convection zone of a furnace, after which it enters the radiation zone of the furnace where it reaches the reaction temperature, in particular a temperature in the range 700°C to 900°C . The mixture is then cooled

and compressed for recovering a certain amount of heat and to purify the waste products. After that, it follows a complex sequence of separation techniques and different chemical treatments.

To assure the correct conditions for the evolution of the pyrolysis reaction, it is important to proportion the reactants at a 3:1 ratio ($1500\text{m}^3/\text{h}$ of fuel and $500\text{m}^3/\text{h}$ of steam).

3. THE CONTROL LEVEL

The control level represents the base level in any hierarchical control system and includes the pieces of equipment used to implement the control algorithms, as well as the devices that connect them to the physical plant, namely the sensors and the actuators.

The control level has to fulfill two important tasks: the data acquisition from the physical plant and the regulation of the major parameters of the process.

The physical data acquired from the plant is further processed and used at this level for the identification of the mathematical models based on which the commands associated to the various regulation systems are computed.

Finally, the control algorithms may be improved in order to preserve the performances obtained in a simulated environment for the closed loop system with the process model identified and the control algorithm computed based on that model when it comes to implementing it on a physical platform.

This means that solutions must be found in order to compensate the uncertainties that appear during the process modeling stage and to ensure stability for the real system as well as the preservation of the performances of the nominal system for the physical plant. One such solution is to compute robust control algorithms, as one will be able to see further in this paper.

3.1 The automation solution

A standard automation solution for the process plant described in the first section of this article should be able to handle two major aspects concerning the proper operation of the process. On the one hand, it should offer the possibility to maintain a good proportion between the quantities of reactants that are fed to the installation and, on the other hand, it should provide the means to maintain the temperature and pressure conditions within the limits imposed by the nature of the chemical reactions that should take place inside the reactor.

Therefore, we have chosen an automation solution with four control loops, as shown in Figure 3.1.

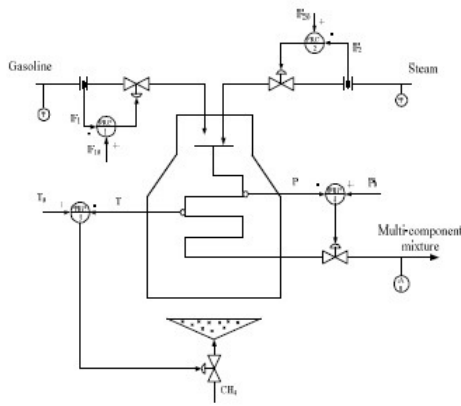


Fig. 3.1. Automation solution for the ethylene reactor

The first two control loops ensure the control of the gasoline and steam flow to the ethylene reactor. Besides the gasoline and the steam flow, another controlled technological parameter is the pressure inside the reactor. It is very important that the pressure value is kept within the boundaries of the admissible operating range so that the process functions correctly and the reactor walls are not submitted to any risk of deterioration. In this case, the pressure inside the reactor should be of at least 3.3 bar and of at most 4.5 bar.

Finally, the temperature within the median section of the reactor is regulated by the control of the quantity of CH_4 used in the combustion process that heats the plant. The automation solution should be able to provide the possibility of maintaining the temperature values within $820^{\circ}C$ and $860^{\circ}C$.

3.2. Gasoline and steam flow control

In this subsection of the article we will briefly present the steps that were followed in order to determine the algorithms suited for the control of the quantities of gasoline and steam that are pumped into the reactor.

First of all, we have analytically determined the mathematical models of the two processes based on a series of process data provided by the plant's owners.

These mathematical models are:

$$H_{F1}(s) = \frac{0.24}{(8s+1)(0.67s+1)} \cong \frac{0.24}{8s+1} \quad (3.1)$$

for the gasoline flow process and:

$$H_{F2}(s) = \frac{0.21}{(4s+1)(0.2s+1)} \cong \frac{0.21}{4s+1} \quad (3.2)$$

for the steam flow process.

Next, we proceeded to design the controllers for the two nominated systems using a PI (Proportional - Integrator) control algorithm. We used the Poles Placement Method in order to determine the

controller parameters that would insure the desired tracking and regulation performances for the closed loop systems.

The two controllers that we designed can be described by the following associated transfer functions:

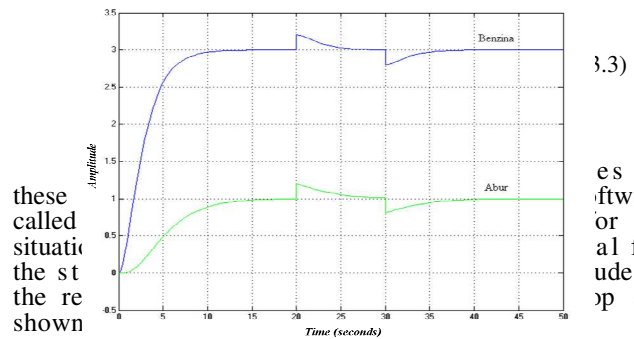


Fig.3.2 .Step response of the gasoline and steam flow control systems

The tracking performances are met and the controller provides the rejection of the step type disturbance signals applied to the output of the system. However, in order to be able to implement these control algorithms on a physical processing device, we need to obtain their corresponding numerical representation.

Using the Euler approximation method for a sampling period of 0.1 seconds, we have the following discrete representations of the two control algorithms ready for implementation:

$$\begin{aligned} H_{R1}(z^{-1}) &= \frac{11.24 - 11.11z^{-1}}{1 - z^{-1}} \\ H_{R2}(z^{-1}) &= \frac{7.92 - 6.34z^{-1}}{1 - z^{-1}} \end{aligned} \quad (3.4)$$

3.3 Temperature and pressure control

For the pressure and the temperature control systems we directly designed the numerical versions of the associated controllers following the steps briefly described below.

First of all, the process models were obtained through an experimental identification method using the data acquired from the physical plant. The most common parametric adaptation method is the

Least Squares Method in its Recursive form. After identification, the models have to be validated using different data sets. For the model validation we used the Whiteness of the Residuals Test, which states that the difference between the output of the physical process after being submitted to disturbances and the model's output should estimate a white noise.

For the temperature control system we have identified and validated the model in (3.5) using a sampling period of 5 seconds. The dynamics of the process is slow, so we can afford to consider a larger sampling period.

$$H_{p3}(z^{-1}) = \frac{0.04711z^{-1}}{1 - 1.61402z^{-1} + 0.65344z^{-2}} \quad (3.5)$$

In the case of the pressure control system, the model that was identified and validated for a sampling period of 2 seconds is the following:

$$H_{p4}(z^{-1}) = \frac{0.00597z^{-1}}{1 - 1.68364z^{-1} + 0.70730z^{-2}} \quad (3.6)$$

Next, we designed two controllers that provide a series of desired tracking and regulation performances using an RST control algorithm whose structure is shown in Figure 3.3.

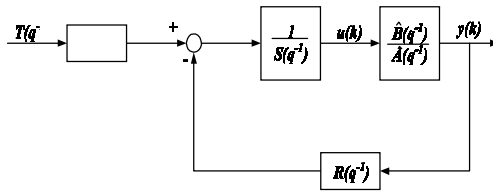


Fig. 3.3. Standard RST controller structure

The reasons for choosing an RST control algorithm over a classical PID control algorithm were the fact that the first one allows for different tracking and regulation performances to be imposed, as well as the fact that it can lead to a robust control algorithm, which is essential given the work context.

The three control polynomials R, S and T were computed using the Poles Placement Method.

Even though the two nominal control systems obtained ensured the desired tracking and regulation performances, the close loop systems were only stable for disturbances with amplitudes of maximum 3% of the reference signal amplitude. Therefore, we had to redesign the control algorithms so that a robustness gauge is met.

3.4 Robustness analysis

The presence of a disturbance with amplitude higher than 3% of the amplitude of the reference signals for the temperature and pressure control systems causes high amplitude variations of the control signal within a short period of time. This kind of shocks can lead to the malfunctioning of the

actuator and has a strong negative influence as far as the physical process is concerned.

Our goal is to design robust controllers that can provide a way of preserving the performances obtained in simulation when it comes to the physical system as well as insuring that the closed loop control system is stable in the presence of disturbing signals or modeling uncertainties.

There are two ways to measure the robustness of a controller. The first one is the output sensitivity function, and the second one is the modulus margin. The output sensitivity function represents the influence of the perturbation on the output:

$$S_{py}(\omega) = \frac{\hat{A}(z^{-1})S(z^{-1})}{\hat{A}(z^{-1})S(z^{-1}) + \hat{B}(z^{-1})R(z^{-1})} \quad (3.7)$$

$\hat{A}(z^{-1})$ and $\hat{B}(z^{-1})$ are two polynomials of the identified process model. $R(z^{-1})$ and $S(z^{-1})$ are the control polynomials computed earlier. It is desirable for the value of the output sensitivity function to be as low as possible.

The modulus margin ΔM represents the minimal distance of the open loop Nyquist plot of the control system and the critical point $[-1, 0j]$. For the closed loop system to be robust, modulus margin has to be maximized.

The output sensitivity function and the modulus margin are related through the following formula:

$$\Delta M = |S_{py}^{-1}|_{\min} = |S_{py}|_{\max}^{-1} \quad (3.8)$$

For a robust system it is necessary to have a modulus margin greater or equal to 0.5, which implies a maximum of 6dB for the output sensitivity function.

The output sensitivity function of the previously designed nominal temperature control system does not meet this requirement, as shown in Figure 3.4.

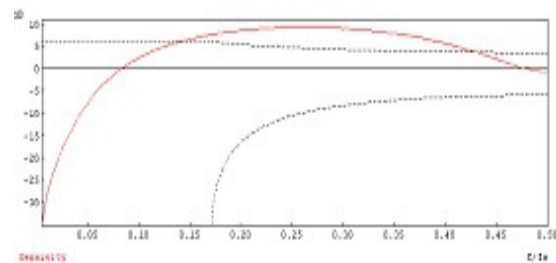


Fig. 3.4 Frequency – amplitude representation of the output sensitivity function for the nominal temperature control system

In order to improve the controller we have added a pair of complex auxiliary poles to the characteristic polynomial given by the denominator of the sensitivity function in (3.7). The new control polynomials are:

$$\begin{cases} R_3(z^{-1}) = 3.42 - 5.53z^{-1} + 2.24z^{-2} \\ S_3(z^{-1}) = 1 - 1.24z^{-1} + 0.24z^{-2} \\ T_3(z^{-1}) = 21.22 - 60.68z^{-1} + 65.15z^{-2} - \\ - 31.19z^{-3} + 5.63z^{-4} \end{cases} \quad (3.9)$$

The output sensitivity function of the new closed loop system meets the imposed 6dB limit, which makes the new controller a robust one.

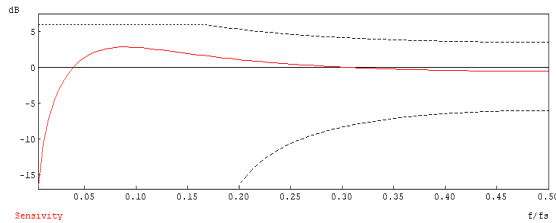


Fig. 3.5 Frequency – amplitude representation of the output sensitivity function for the robust temperature control system

The tracking and the regulation performances from the nominal case are preserved, but the control signal does not undergo such violent variations in the presence of a perturbation.

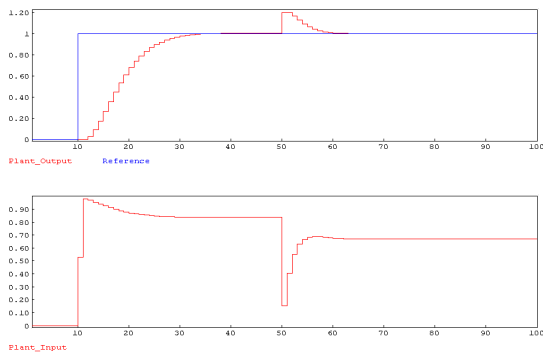


Fig. 3.6 Step response and process input for the robust temperature control system

As far as the pressure control system goes, we have encountered a similar situation.

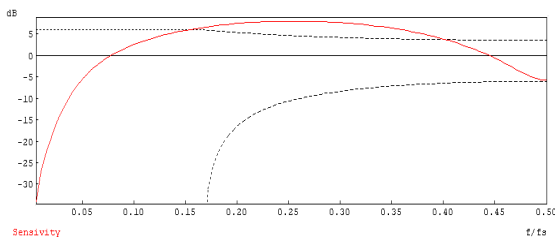


Fig. 3.7 Frequency – amplitude representation of the output sensitivity function for the nominal pressure control system

This time we added two real poles to the characteristic polynomial and obtained the following control polynomials:

$$\begin{cases} R_4(z^{-1}) = 14.34 - 24.15z^{-1} + 10.14z^{-2} \\ S_4(z^{-1}) = 1 - 1.45z^{-1} + 0.45z^{-2} \\ T_4(z^{-1}) = 167.5 - 526.68z^{-1} + 620.1z^{-2} - \\ - 325.5z^{-3} + 64.24z^{-4} \end{cases} \quad (3.10)$$

The new controller is also a robust one.

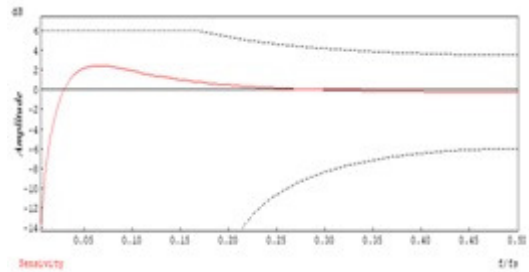


Fig. 3.8 .Frequency – amplitude representation of the output sensitivity function for the robust pressure control system

Figure 3.9 shows that the tracking and the regulation performances imposed for the nominal system are preserved and that the perturbation does not have such important influence over the process input signal anymore.

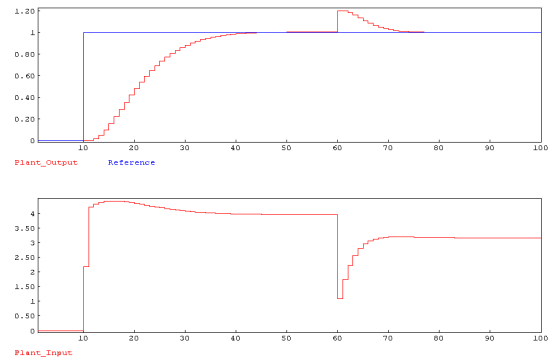


Fig. 3.9. Step response and process input for the robust pressure control system

4. OPTIMIZATION LEVEL

The optimization problem of the operating conditions supposes to determine the point in the admissible domain where the process performances are optimal. In this case study, the optimization objective is to maximize the concentration of ethylene from the plant output (z), computing the best choice for the reference signals. For solving this problem, we utilize a number of experimental data collected directly from the plant, data representing values for the gas flow (y_1)(m^3/h), the steam flow (y_2), the reaction pressure (y_3)(bar), and the reactor temperature (y_4)[$^{\circ}C$]. We estimated a model,

in which the quality variable z represents the concentration of ethylene in the final multicomposite mixture.

The structure of the multivariable nonlinear model has this form:

$$\hat{z} = \hat{a}_0 + \hat{a}_1 y_1 + \hat{a}_2 \frac{1}{y_2} + \hat{a}_3 y_3 + \hat{a}_4 y_4^2 \quad (4.1)$$

where

$$\hat{\theta}^T = \left[\hat{a}_0 \ \hat{a}_1 \ \hat{a}_2 \ \hat{a}_3 \ \hat{a}_4 \right] \quad (4.2)$$

represents the model parameters determined using the Least Squares method (LS) on a set of gathered data. The optimization problem is formulated in this form:

$$\max \hat{z}(\hat{\theta}, y) \quad (4.3)$$

with the restrictions:

$$\begin{aligned} 1000 &\leq y_1 \leq 1600 \left[\frac{m^3}{h} \right] \\ 430 &\leq y_2 \leq 540 \left[\frac{m^3}{h} \right] \\ 3.3 &\leq y_3 \leq 4.5 [bar] \\ 820 &\leq y_4 \leq 860 [^{\circ}C] \end{aligned} \quad (4.4)$$

The results obtained from a Matlab calculus routine are conforming to the limitations imposed, looking in this form: $y_1^* = 1599.6 [m^3/h]$, $y_2^* = 430.03 [m^3/h]$, $y_3^* = 3.3 [bar]$ and $y_4^* = 859.85 [^{\circ}C]$. The quantity of ethylene produced in this case represents the optimal value $\hat{z} = 34.16\%$ from the total of the pyrolysis reaction final products.

The optimal decision (y_1^* , y_2^* , y_3^* , y_4^*) from this supervisory level is automatically transferred as references to the plant. After implementing this method, the concentration of ethylene has been increased with approx. 4% compared with the mean value.

5. CONCLUSIONS

1. One of the major problems of the reactive systems safety is the thermal instability whose consequences may be: low selectivity, the occurrence of side effects or, in some cases, the reactor's explosion. A reactor design should be based on a careful study of thermal stability. It is important to determine the areas where the reactor's behavior is uncertain. In practice, most studies of this type are based on a well-known stability criterion: the van Heerden criterion.

It is possible to have many steady-states, so an important question of concern is whether all these states are stable, allowing to the perturbations that always occur in a real system. In case of an unstable thermal operation, additional measures may be taken for introducing a controller that will ensure the closed loop heat-stable operation.

2. Concerning the optimization problem, a real reactor operation is not always characterized by the same leading

model, being necessary a new identification in different operating conditions. Considering that the reactor's supply is not always using the same type of gasoline, its quality being one of the process efficiency criteria and also, because we do not know when the petrol quality change occurs, the optimization problem gets a stochastic character. As the concentration of ethylene - one of the main reaction products which interest us in particular - is dependent on the gas entering the reactor, it is necessary to determine an adequate process model in each case.

The restrictions are dependent on a random variable, ω , the quality of petrol, which is amending in each case. The main idea is to transform the problem into a linear stochastic equivalent, deterministic in nature, with linear restrictions, resolvable with nonlinear programming methods, so that requiring a minimum risk, the criterion function can not take values lower than a given value (in our case, the criterion function is represented by the concentration of ethylene that must be maximized).

3. Finally, the results obtained in this study can be implemented on specialized equipment. For the acquisition and numerical control functions a microcontroller of a certain configuration is used, as in the case of the ethylene reactor from Pitesti. Also, an operator console is used for the optimization function.

4. The first two points of the conclusions represents for us researches in progress which will be published later.

REFERENCES

- [1] D.Popescu, D.Stefanoiu, C.Lupu, C.Petrescu, B.Ciubotaru, C.Dimon - *Industrial Process Control*, AGIR Press, Bucharest, 2006.
- [2] I.D.Landau, Gianluca Zito - „*Digital Control Systems-Design, Identification and Implementation*”, Springer-Verlag London Limited, 2006.
- [3] Mark E. Davis, Robert J. Davis - „*Fundamentals of Chemical Reaction Engineering*”, McGraw-Hill Companies, Inc., 2003.
- [4] William H. Green - „*Lecture 11: Nonisothermal Reactors, equilibrium limitations, and stability*”, Chemical and Biological Reaction Engineering, Spring 2007.
- [5] A. Kazode Coker - „*Modeling of Chemical Kinetics and reactor design*”, Gulf Publishing Companz Houston, Texas, 2001.
- [6] Al. C. Dimian, Costin S. Bildea -, „*Chemical Process Design*”, WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim, 2008.
- [7] M.J. Willis -, „*Continous stirred tank reactor models*”, Dept. of Chemical and Process Engineering, University of Newcastle, 2000

Repurposing a Wiki for Collaborative Learning - Pedagogical and Technical View

Elvira Popescu, *Member, IEEE* and Laurentiu Manafu

Abstract—Many Web 2.0 tools can be repurposed for the educational context. Wikis, in particular, can be used as support tool for collaborative learning among the members of a team, e.g., for co-writing tasks as well as for creating and organizing knowledge and resources. The first part of the paper features a series of case studies about the use of wikis in educational settings, including our own successful experience in the framework of an undergraduate course taught at the University of Craiova, Romania. The second part of the paper focuses on enhancing an existing wiki platform with some functionalities specifically tailored for the educational context. In particular, these features provide support to the instructor in the evaluation and grading process, as assessment represents a challenge in collaborative pedagogical approaches.

I. INTRODUCTION

IN recent years, Web 2.0 tools (also known as "social software tools", e.g., blog, wiki, social bookmarking systems, media sharing tools) have found their way into the educational landscape [1, 5, 7], with encouraging results with respect to student satisfaction, knowledge gain and/or learning efficiency. This is motivated by the fact that the principles Web 2.0 is based on (user-centered, participative architecture, openness, interaction, social networks, collaboration) are in line with modern educational theories such as socio-constructivism [18]. According to it, knowledge cannot be transmitted but has to be constructed by the individual, by means of collaborative efforts of groups of learners [17].

A comprehensive and most up-to-date literature review of the use of Web 2.0 tools in higher education is included in [3]. The paper provides also a typology of Web 2.0 tools, derived from [4]: *i) media sharing (creating and exchanging media with peers or wider audiences); ii) media manipulation and data/web mash ups (using web-accessible tools to design and edit digital media files and combining data from multiple sources to create a new application, tool or service); iii) instant messaging, chat and conversational arena (one-to-one or one-to-many conversations between Internet users); iv) online games*

and virtual worlds (rule-governed games or themed environments that invite live interaction with other Internet users); v) social networking (websites that structure social interaction between members who form subgroups of 'friends'); vi) blogging (an Internet-based journal or diary in which a user can post text and digital material while others can comment); vii) social bookmarking (users submit their bookmarked web pages to a central site where they can be tagged and found by other users); viii) recommender systems (websites that aggregate and tag user preferences for items in some domain and thereby make novel recommendations); ix) wikis and collaborative editing tools (web-based services that allow users unrestricted access to create, edit and link pages); x) syndication (users can 'subscribe' to RSS feed enabled websites so that they are automatically notified of any changes or updates in content via an aggregator) [3]. It could be argued that some of these tools also existed in the first stages of the Web; however, their evolved versions allow for better integration and community building (e.g., by means of tagging, commenting, rating, syndication, development of "friendships").

The focus of this paper is on the wiki, a platform which allows users to easily create and edit pages collaboratively, enabling the co-construction of content. The term "wiki" is derived from the Hawaiian phrase for "quick", referring to the fast and easy way in which pages can be created and updated by users; edits can be made in real-time and appear almost instantly online, generally not requiring any login or review (some wikis may however restrict access to a group of members, requiring user authentication to edit or sometimes even read pages) [19].

According to the "inventor" of the wiki [8], its main features are: *i) pages can be edited or created using a simple Web browser without any extra add-ons; ii) meaningful topic associations between different pages are promoted by making page link creation very easy and showing whether an intended target page exists or not; iii) the role of the wiki is not to provide a carefully crafted site for casual visitors, but to involve them in the ongoing process of collaborative content creation.*

Currently, there is a large number of wiki software packages available [20], and also a large number of websites using them [21]. Wikipedia is a prominent example, but there are also other initiatives, such as Wikitravel (worldwide travel guide with almost 25.000 destinations), TV IV (television compendium), Ganfyd

This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID 61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007 – 2013.

E. Popescu is with the Software Engineering Department, University of Craiova, Romania (e-mail: popescu_elvira@software.ucv.ro).

L. Manafu was student at the University of Craiova, Romania.

(medical knowledge base) or WikiCFP (Calls for Papers in science and technology fields, with more than 15.000 events). Apart from community websites, wikis can also be used for corporate intranets, knowledge management systems, note services or personal notetaking.

Wikis have started to be used also in educational settings, as a support tool for collaborative learning [10]. The second section of the paper focuses on this educational use, including a pedagogical scenario for a collaborative writing task in an online course, as well as several case studies. We also report on our own successful experience with using a wiki for a collaborative project assignment, in the framework of an undergraduate course taught at the University of Craiova, Romania. The third section of the paper deals with the extension of an existing wiki platform with some functionalities particularly tailored for an educational context. More specifically, some of these features provide support to the instructor in the evaluation and grading process, as teachers often experience difficulty in evaluation when proposing co-writing activities to their students [14]. Not only the final products developed by students must be assessed, but also the overall contribution process, therefore assessment is a major challenge in collaborative pedagogical approaches [3]. To further alleviate the burden for the instructors, some future improvements are proposed in the fourth section of the paper, which also concludes the work.

II. PEDAGOGICAL REPURPOSING – CASE STUDIES

As stated in the introduction, many Web 2.0 tools can be pedagogically repurposed to meet specific course design contexts and requirements [6]. Wikis play an important role in this respect, due to the proven effectiveness of co-writing as a strategy of collaborative learning [16].

Thus, in educational settings, wikis can be successfully used for collaborative writing tasks among the members of a team, as well as for creating and maintaining learning content, both by students and teachers [5]. Paper [10] offers a comprehensive review on wiki as teaching tool, exploring its contribution to various learning paradigms (collaborative / cooperative learning, constructivism, cognitive apprenticeship), as well as actual use cases, as reported in the literature.

According to [16], co-writing processes stimulate reflection, knowledge sharing and critical thinking, in a strong social context. The use of wikis as a support tool to implement distributed collaborative writing presents the advantage of built-in monitoring facilities (e.g., versions, comments, linkers). Furthermore, wikis are nowadays used for internal knowledge management by many companies, so the practical experience gained by the students would prove useful in the workplace. Moreover, the use of wikis has an advantage for the teachers also, since it allows for a more complex evaluation; they can take into account not

only the quality of each individual production, but also the evolution of the work, its improvement throughout each stage (by means of versions), the interactions among the participants (by means of comments), the reticular structuring of the final work (by means of links).

A detailed scenario of educational wiki use for a collaborative writing task in the framework of an online course is presented in [16]. It includes several steps: i) individual study of recommended materials; ii) co-planning of the hypertext's general structure (sections and first level subsections) and division of work; iii) development of the various parts of the wiki (working individually); iv) links to pages created by others; v) peer review and improvement suggestions made to colleagues. The fourth step (performed throughout the development process, not just at the end) insures that students don't focus exclusively on their own part of the text; instead, they check the progress of the other's work constantly, understanding the conceptual links inside the subject matter, getting new ideas for their own parts and avoiding duplications.

Wikis have also found their place in Computer Science education, as reported in the following studies:

- Paper [15] presents a wiki environment used for the "Software Requirements for Business Systems" post-graduate course. Students' task was to populate the wiki with a set of requirements from the perspectives of a variety of stakeholders and identify conflicts, duplicates and ambiguities. Furthermore, students had to agree on a fit-criterion for each requirement, a task which was very suitable for a team of requirements engineers working collaboratively.

- Paper [12] reports on designing a course around a wiki, rather than simply introducing a wiki in an existing course. A whole collaborative context was created for the "Introduction to Media Computation" course - the wiki featured student home pages, pages for homework question and answers, exam reviews, design galleries in which students could share their work etc.

Similar empirical reports within various fields of study include: [2], [9], [13].

Based on the successful results reported in the literature, we performed our own case study, which involved the use of both a wiki and a blog [11]. The context of our study was the project activity pertaining to a course on "Web Applications' Design" (WAD) for fourth (last) year undergraduate students in Computer Science from the University of Craiova, Romania. 36 students were enrolled in this course during the first semester of the 2009-2010 academic year. It was a team-based project and consequently students were split in 7 groups, each having between 4 and 6 members (depending on the complexity of the assignment). The project assignment consisted in the design and implementation of a Web application (e.g., online bookstore, virtual learning class, online auction

system etc.).

The project activity was done in a blended mode, with weekly face-to-face meetings in which students could directly interact with the instructor, asking for help and clarifications, doing hands-on exercises, discussing contributions, checking the assignment progress, and presenting the final project results. Furthermore, students were trained to use a wiki and a blog as support tools for the project tasks. Thus they were instructed to use the wiki for: i) gathering and organizing their knowledge and resources regarding the project theme; ii) collaboratively writing the project documentation. They were also instructed to use the blog in order to: i) communicate and collaborate with team members; ii) interact with other teams and exchange experience; iii) publish ideas, thoughts, interesting findings (project and course-related); iv) report on the progress of the project (daily/weekly activity report: what did they do, what problems did they encounter, what solutions did they find); iv) by means of comments, provide

help to peers, critical and constructive feedback etc. Using the blog and the wiki were compulsory activities; an active contribution (in terms of actual content, frequency, periodicity, feedback for peers) counted for 50% of the final mark.

Both the blog and the wiki were team-oriented (i.e., each team had their own blog and their own namespace on the wiki). However, as far as the wiki is concerned, the collaboration between teams was restricted to visualizing each others' contributions, but without having the possibility to intervene (only the members of a team had write access within their namespace). This was done in order to boost cooperativeness inside teams but also competitiveness between teams. From a technical viewpoint, DokuWiki was chosen as wiki software (because of the ease of use and simple and flexible user access control).

Figure 1 includes a snapshot from Team 1's wiki, capturing the starting page with the team components and an excerpt from the project assignment.

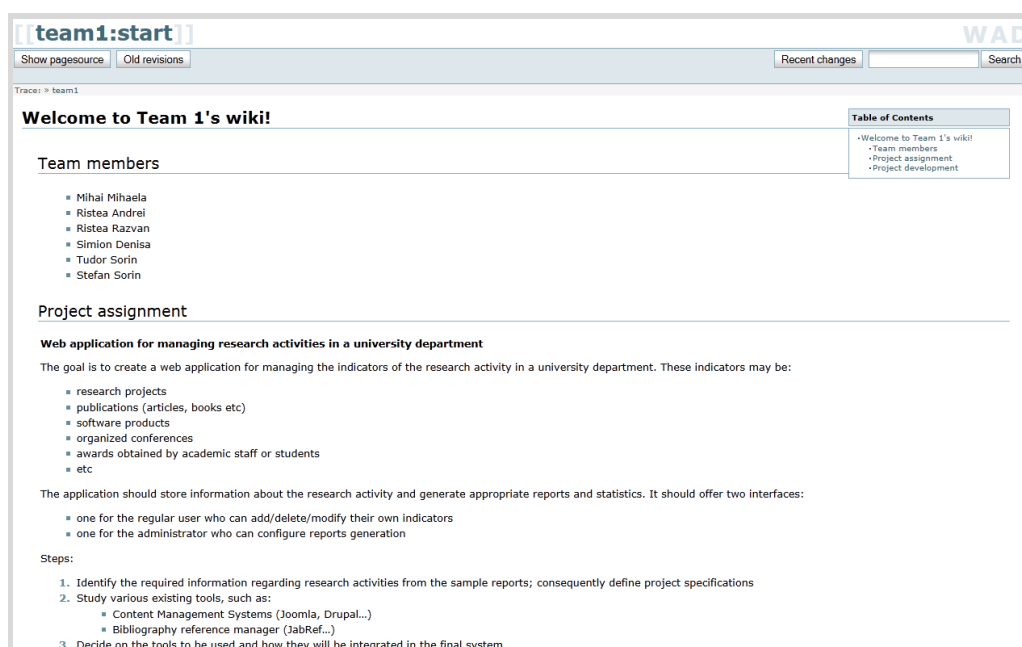


Fig. 1. Snapshot from a team's wiki

According to the questionnaires applied at the end of the course, students found the wiki easy to learn and use, with virtually no technical problems encountered. Furthermore, they enjoyed working with the wiki and found it very useful, particularly in: i) helping to organize knowledge; ii) improving collaborative skills; and iii) facilitating communication and collaboration between team members. More details regarding the results of the study can be found in [11].

III. TECHNICAL REPURPOSING – PROOF OF CONCEPT

While the use of a general purpose wiki in the

instructional process proved helpful, it would be undoubtedly useful to extend it with some features specifically designed for collaborative learning and automatic evaluation. These functionalities would provide the required support both for students and instructors. Thus, student's work could be rated both by peers and teacher; the learner could receive valuable feedback on each wiki page and have the opportunity to rate this feedback. Furthermore, all students' interactions with the wiki could be tracked and recorded (i.e., time and frequency of system accesses, pages created by each student, pages edited, internal and external links added etc.). The instructors could be given access to

an aggregated summary of this activity, which could provide support in the grading process. Thus the wiki would not only be a place for collaborative writing and knowledge creation, but also a powerful student evaluation tool.

This approach is supported also by Trentin [16], who proposes a methodology for using a wiki to evaluate individual student contributions to a collaborative learning project. The authors introduce a series of ad hoc formulae to assess each student's involvement in the project. Four criteria are considered relevant, namely: i) student's contribution to a forum discussion during the collaborative planning of the document's overall structure; ii) student's peer review of a colleague's page (as assessed by the page creator); iii) number of links created by a student between her own page cluster and other page clusters (i.e., contribution to the reticularity of the final hypertext); iv) quantitative measure of content developed by a student (i.e., number of wiki pages and total number of characters). The analysis proposed in the paper is performed manually, but the authors suggest a series of functionalities that could be embedded in a wiki to automate parts of the evaluation. Starting from these suggestions, we devised our own set of functionalities to be added to the wiki.

The first step of this extension process consisted in choosing a wiki as a starting point. After careful analysis we decided to use DokuWiki, both because of its ease of use by the end-users (as already pointed out in section 2),

but also by the developers. DokuWiki is a standards compliant wiki, with a simple but powerful syntax which makes sure the data files remain readable outside the Wiki (i.e., all data is stored in plain text files – no database is required) [22]. DokuWiki is written in PHP and released under GPL 2 license; it has well documented source code and an active development community. Also, it has a flexible plugin API which allows to extend its functionality [23]. Five types of plugins can be defined: i) Syntax Plugins (which extend DokuWiki's basic syntax); ii) Action Plugins (which can be used to extend or replace many aspects of DokuWiki's core operations, from saving wikipages to adding new action modes); iii) Admin Plugins (which can provide administration functionality for DokuWiki, being accessible to superusers and managers); iv) Helper Plugins (which can be used to provide support functionality to many other plugins); v) Renderer Plugins (which allow to create new export modes and to replace the standard DokuWiki XHTML renderer) [23].

Starting from DokuWiki, we have thus built a prototype implementation that we named EdDokuWiki. The additional functionalities were added as *action plugins* into the original wiki; furthermore, a separate instructor module was added, implemented in PHP and using MySQL 5.0 as DBMS.

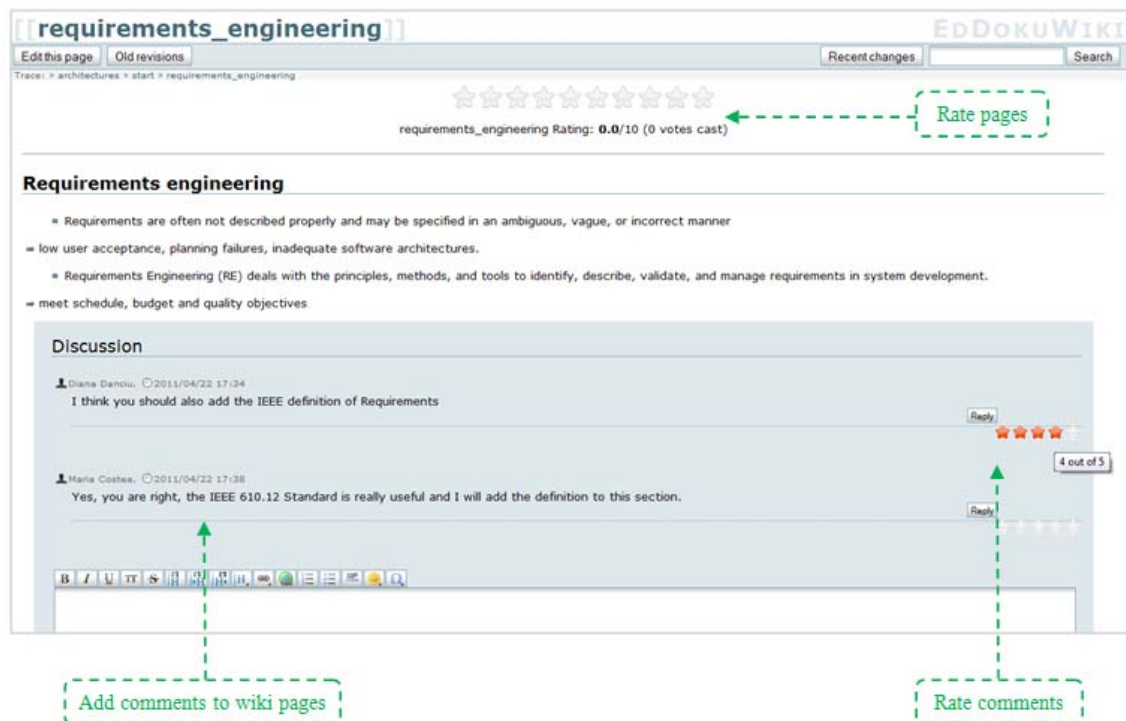


Fig. 2. EdDokuWiki extensions - peer evaluation support

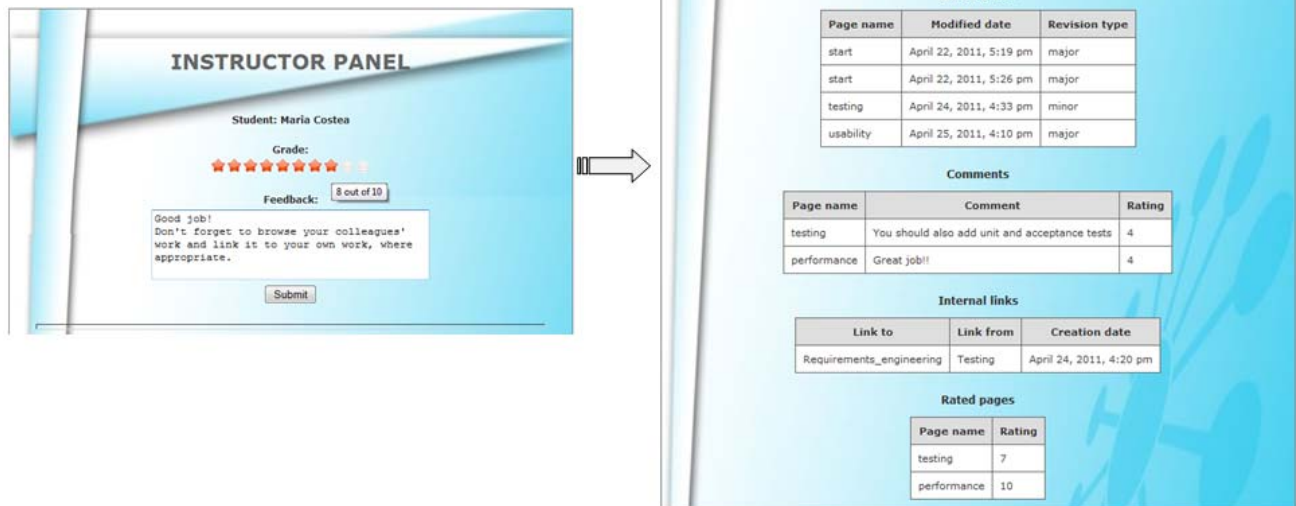


Fig. 3. Additional instructor module:

i) grade student & provide feedback (left-hand side); ii) visualize a summary of student's contributions (right-hand side)

Currently, EdDokuWiki supports the following set of functionalities:

- Peer evaluation support: add comments to wiki pages, rate pages, assess the utility of the comments received
- Monitor and record all student actions in a database: *login*, *logout*, *create_page*, *edit_page*, *add_internal_link*, *add_external_link*, *add_comment*, *rate_page*, *rate_comment*. Each action has a corresponding timestamp and description.
- Automatic (quantitative) evaluation of student contributions: number of pages, number of characters on each page, number of internal/external links, amount of time spent on the wiki
- Instructor evaluation support: visualize a summary of each student contributions, grade student, provide individual feedback

Some of these functionalities are illustrated in figures 2 and 3 respectively.

EdDokuwiki is currently at prototype level and its large-scale evaluation is still unfolding. We are currently testing it in an experimental study similar to the one described in section 2. Apart from evaluating students' satisfaction with the improved tool, we also aim at finding out instructor's opinion on the dedicated evaluation module. Furthermore, we will perform comparisons between the scores automatically computed by the system for each student (based on the quantitative aspect of the contributions) and the actual evaluation performed by the instructor (taking

into account also the quality of the contributions).

IV. CONCLUSION

We started the paper with a series of case studies about the use of wikis in educational settings. While these have shown that using a general purpose wiki as a collaborative learning support tool can yield successful results, they have also uncovered the need for some supplementary functionalities, specifically tailored for an educational context. In particular, some of these additional features pertain to the learner tracking category, providing support to the instructor in the evaluation and grading process. The second part of the paper focused on the extension of an existing wiki platform (DokuWiki) with these desired functionalities, yielding a special purpose wiki (EdDokuWiki).

As future work, we plan to add further extensions to the wiki, to better monitor the participation of the students during group work (e.g., graphical visualizations of students' interactions and social network analysis). We also plan to offer instructors the possibility to define their own formulas to automatically compute the "wiki score" for each student (e.g., by means of a configuration panel). Depending on the particularities of each course and collaborative task, the instructor might want to assign different weights to the different available parameters (i.e., number and length of revisions, number of internal/external links, amount of time spent on the wiki, peer evaluation scores etc.). In fact, experimenting with various formulas

for computing student's "wiki score" and statistically analyzing the results is another research direction that we plan to address.

REFERENCES

- [1] P. Anderson, *What is Web 2.0? Ideas, technologies and implications for education*, technical report, JISC, 2007, available at: www.jisc.ac.uk/media/documents/techwatch/tsw0701b.pdf
- [2] M. Byron, "Teaching with Tiki", *Teaching Philosophy* 28(2), 2005, pp. 108-113.
- [3] G. Conole and P. Alevizou, *A literature review of the use of Web 2.0 tools in Higher Education*, report commissioned by the Higher Education Academy, The Open University, UK, Aug. 2010, available at: http://www.heacademy.ac.uk/assets/EvidenceNet/Conole_Alevizou_2010.pdf
- [4] C. Crook et al., *Web 2.0 technologies for learning: the current landscape – opportunities, challenges and tensions*, Becta report, 2008, available at: http://dera.ioe.ac.uk/1475/2/becta_2008_web2_currentlandscapeadditional_litrev.pdf
- [5] K. Grodecka, F. Wild, and B. Kieslinger, *How to use social software in higher education*, iCamp handbook, 2009, available at: <http://www.icamp.eu/wp-content/uploads/2009/01/icamp-handbook-web.pdf>
- [6] A. Hemmi, S. Bayne, and R. Land, "The appropriation and repurposing of social technologies in higher education", *Journal of Computer Assisted Learning*, vol. 25, 2009, pp. 19–30.
- [7] M. Homola and Z. Kubincova, "Taking Advantage of Web 2.0 in Organized Education (A Survey)", *Proc. ICL 2009*, pp. 741-752.
- [8] B. Leuf and W. Cunningham, *The Wiki Way: Quick Collaboration on the Web*, Addison-Wesley, 2001.
- [9] D.L. Neumann and M. Hood, "The effects of using a wiki on student engagement and learning of report writing skills in a university statistics course". *Australasian Journal of Educational Technology* 25(3), 2009, pp. 382-398.
- [10] K.R. Parker and J.T. Chao, "Wiki as a Teaching Tool", *Interdisciplinary Journal of Knowledge and Learning Objects*, 3, 2007, pp. 57-72.
- [11] E. Popescu, "Students' Acceptance of Web 2.0 Technologies in Higher Education: Findings from a Survey in a Romanian University", *Proc. DEXA 2010 Workshops*, IEEE Computer Society Press, 2010, pp. 92-96.
- [12] J. Rick and M. Guzdial, "Situating CoWeb: a Scholarship of Application", *International Journal of Computer-Supported Collaborative Learning* 1(1), 2006, Springer, pp. 89-115.
- [13] A. Ruth, L. Houghton, "The wiki way of learning", *Australasian Journal of Educational Technology* 25(2), 2009, pp. 135-152.
- [14] L. Swan, J. Shen, and S.R. Hiltz, "Assessment and collaboration in online learning", *Journal of Asynchronous Learning Network*, 10, 2006.
- [15] P. Thomas, S. Minocha, D. King, J. Taylor, N. Sclater, and M. Schencks, *Collaborative Learning in a Wiki Environment: Experiences from a Software Engineering Course*. Technical report no. 2008/04, The Open University, UK, 2008.
- [16] G. Trentin, "Using a wiki to evaluate individual contribution to a collaborative learning project", *Journal of Computer Assisted Learning*, 25(1), 2009, pp. 43–55.
- [17] C. Ullrich, K. Borau, H. Luo, X. Tan, L. Shen, and R. Shen, "Why Web 2.0 is good for learning and for research: principles and prototypes", *Proc. WWW 2008*, ACM Press, 2008, pp. 705-714.
- [18] L. S. Vygotsky, *Mind in society*, Harvard University Press, Cambridge, MA, 1978.
- [19] ***, Wiki, available at: <http://en.wikipedia.org/wiki/Wiki> (last accessed on May 15, 2011)
- [20] ***, List of wiki software, available at: http://en.wikipedia.org/wiki/List_of_wiki_software (last accessed on May 15, 2011)
- [21] ***, List of wikis, available at: http://en.wikipedia.org/wiki/List_of_wikis (last accessed on May 15, 2011)
- [22] ***, DokuWiki Manual, available at: <http://www.dokuwiki.org/manual> (last accessed on May 15, 2011)
- [23] ***, DokuWiki Development, available at: <http://www.dokuwiki.org/development> (last accessed on May 15, 2011)

Real-time Path Planner for Arm Robot Collision Avoidance

D. Puiu¹ and F. Moldoveanu¹

¹Transilvania University of Brasov, Automation Department
puiudan@unitbv.ro, moldof@unitbv.ro

Abstract— The paper presents a real-time path planner for a redundant manipulator. The robot manipulator joints references positions are modified, by the control structure, when an unexpected moving obstacle is getting closer to one of the robot components (links and joints). The position of the moving objects is determined using a digital image processing system. The control system determines if the moving obstacle and the arm robot are in collision using an interference detection method based on oriented bounding boxes. When the arm robot control system detects a collision, it modifies the arm robot joints trajectories in order to increase the distance between the robot and the obstacle and, in the same time, maintain the initial trajectory of the end-effector. Computer simulation and experimental results are given for a 7 DOF redundant arm and demonstrate the collision avoidance capability.

I. INTRODUCTION

One key requirement in the field of autonomous robotics is the real-time calculation of the obstacles from the visualized scene [1]. As a result of progress in the field of robot control and robot vision, a number of control and path planning methods have been reported in the literature. Nevertheless, there is still a lack of on-line robust, vision based, obstacle avoidance algorithms for autonomous robots.

Random algorithms are a very used technique for finding a collision free path for an articulated arm robot. In this category falls the Probabilistic Roadmap (PRM) method [10], as well as the Rapid exploring Random Tree (RRT) approach [11]. Both of them determine efficient collision-free trajectories, on the expense of high computation time.

PRM has two phases, that is, a learning and a query phase. Initially, a probabilistic roadmap, which contains all possible routes, is constructed and it is saved within an undirected graph. When a robot has to execute a path planning task, it only has to choose the appropriate route from the probabilistic roadmap [10]. Because the RRT method uses a tree instead of a graph, it can directly, generate the manipulator's trajectory using a set of computation constraints [11]. Another important group of path planners, related to PRM and RRT, are based on Genetic Algorithms (GAs), or evolutionary computation [2], [4], [13].

Although many path planning methods for object manipulation in a-priori known environments have been published, there is still a lack of reliable real-time planners that could optimally operate in unknown and dynamic scenes. A step in developing such a strategy was reported by Jaillet in [8], where he used an RRT method for the generation of the arm's trajectory and a PRM approach for

the on-line recalculation of the robot's probabilistic roadmap when an obstacle is detected.

In case of the mobile manipulators, the method proposed by Brock in the paper [3] allows the modification of the robot base to avoid a collision. In the same time, the algorithm modifies the arm robot configuration to maintain constant the end-effector position. The algorithm elaborated by Tan in the paper [16] stops the base of the mobile robot when an obstacle is getting closer to it and modifies the arm robot trajectory to control the end-effector position.

Another method to generate the trajectory in unknown environments uses the reactive manipulation strategy and it is presented by Ogren in the paper [16]. He considers that the arm robot works in a potential field. During the motion the arm robot has to respect the end-effector trajectory and to maintain the arm joints and links in the low potential areas, considering that obstacles have high potential. In the reactive potential technique can be also included the intelligent motion controller [14] developed by Mbede. He uses the PRM method to generate the global trajectory and a reactive fuzzy controller to adapt it to the environment changes.

Vannoy and Xiao have developed the real-time adaptive motion planning (RAMP) method [17] that facilitate the real-time modification of the trajectory of a mobile manipulator. The algorithm minimizes the time and energy consumption and maximize the manipulability.

There is also an online trajectory planner that controls the motion of several arm robots that work in the same working space [12]. In this case the planning structure knows the current position of each arm robot and it generates the trajectories in order to avoid the collisions.

In this paper, the attention is focused on the problem of controlling in real-time an articulated arm robot that works in a dynamic environment with unforeseen changes. In the robot working space are moving objects that can collide with one of its components (joints or links). The position of the moving objects is determined using a digital image processing system. The paper contains an algorithm that determines (if it is possible) the arm robot collision free configuration when a moving obstacle is getting closer to the robot. In the last part of the paper is presented a strategy that uses the algorithm to change in real-time the joints trajectories during a transition. Simulations and practical tests results are provided.

II. COLLISION DETECTION

In order to avoid collisions in a dynamic scene, a robotic system has to reliably detect the poses of objects and obstacles in order to correct its movement in real-time. The Cartesian position of the moving obstacles is

detected using the vision architecture developed by Grigorescu [7] that has the scene geometry estimation and the scene understanding modules.

It will be considered that the moving obstacle is in collision with the articulated arm robot if the smallest distance between robot components and the obstacle is smaller than $dist_1^*$.

The shape of the moving obstacle, and also the shape of the arm robot components, can be very complex. To simplify the collision detection procedure, the arm robot components and the moving objects were represented using oriented bounding boxes (OBBs) [6].

As is represented in figure 1 a), it was created an OBB model for the moving object. The model contains an array of small triangles that represents the surface of the object. To detect if the distance between the robot and object is smaller than $dist_1^*$ the surface of the object model is bigger, with $dist_1^*$, than the real surface.

The links, respective the joints, of the arm robot have their own OBB models, as is represented in figure 1 b). Having the position of the OBB models, it was detected, using the interference detection method [6] elaborated by Gottschalk, if there is a collision. If the collision occurs the method determines which triangles of the two OBB models have common points.

Each triangle of the moving obstacle OBB model has a normal vector that respects the following constraints:

- it starts from the centre of the incircle;
- it is perpendicular on the surface of the triangle;
- it is orientated from the inside to the outside of the obstacle.

It will be considered that the vector of the m moving obstacle OBB triangle has the orientation:

$$\vec{v}_m = x_m \vec{i} + y_m \vec{j} + z_m \vec{k} \quad (1)$$

and it passes through the point S_m of the arm robot base Cartesian frame (which is considered as reference frame). The vectors $\vec{i}$, $\vec{j}$ and $\vec{k}$ are the normal vectors of the reference frame.

The surface of the moving can be very complex and, as a result of that, it contains many small triangles. In the example from figure 1, the surface of the moving obstacle is very simple and in OBB models collision are involved only two triangles.

III. COLLISION AVOIDANCE

When the control structure know that the arm robot it is in collision, it determines the triangles of the moving obstacle OBB model that have common points with the arm robot OBB model. In the next step, it determines the orientation of the normal vector from the collision surface of the moving obstacle, as:

$$\vec{v}_c = \frac{\sum \vec{v}_m}{n} \quad (2)$$

where $\vec{v}_m$ represent the moving obstacle OBB triangles that are in collision and n is the total number of collision triangles.

The starting point of the $\vec{v}_c$ vector is calculated in a similar way as follows:

$$S_c = \frac{\sum S_m}{n} \quad (3)$$

In the next step the control structure calculates, using the Euclidian vector geometry one of the tangents to the moving obstacle collision surface (it should be perpendicular on $\vec{v}_c$ and pass through S_c). It will be considered that P_{SI} is the point from the tangent at 10 cm

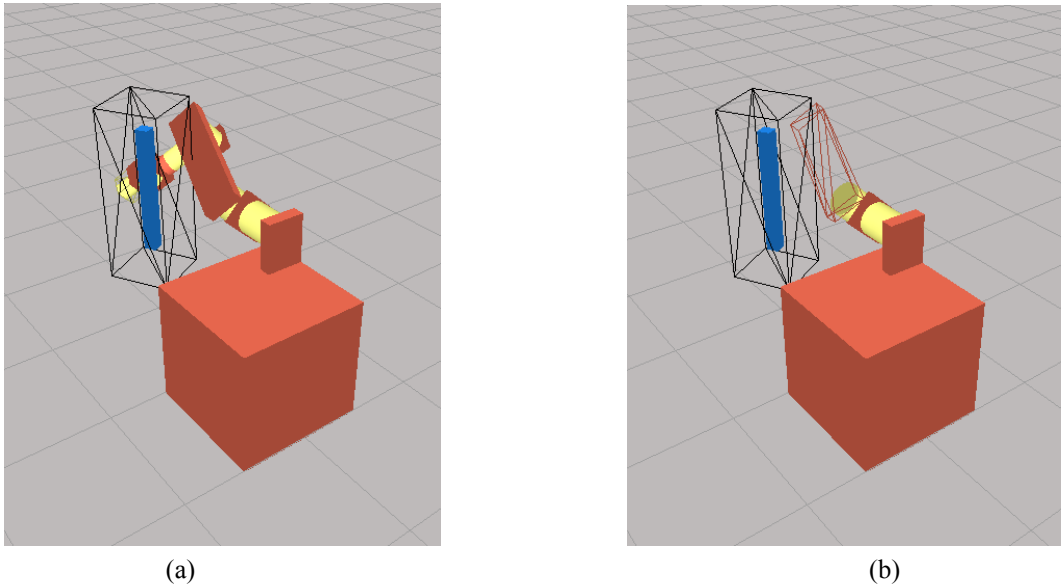


Figure 1. The OBBs models for the articulated arm robots and for the moving object; a) moving object; b) link 2 of the arm robot.

at the left of S_c and the P_{s2} point at 10 cm at the right of S_c .

Also, it will be considered that the points P_k and P_{k+1} represents the references of the coordinate frames of the two consecutive joints that are near the collision point. The P_k point represents the reference of the coordinate frame attached to the joint closer to the base of the robot, respective P_{k+1} for the joint closed to the end effector.

To determine the configuration of the arm robot that avoids the collision state, the joints of the arm must be modified in order to be simultaneous respected the following two constraints:

- increase the distance between the two lines $P_{s1}P_{s2}$ and P_kP_{k+1} until it is equal with $dist_1^*$ – for avoiding the collision state;
- maintain the same distance between P_{k+1} and the current position of the end-effector (P_d) – for increasing the chances to maintain the same position and orientation of the end effector.

The shortest distance between two screw lines from space is determined using the equation developed by Eberly in [5]. It will be considered that the function $d_1(P_{s1}, P_{s2}, P_k, P_{k+1})$ returns the shortest distance between the lines $P_{s1}P_{s2}$ and P_kP_{k+1} .

The distance between the points P_{k+1} and P_d is determined using the Euclid method:

$$d_2(P_{k+1}, P_d) = \sqrt{P_{k+1}P_d \cdot P_{k+1}P_d} \quad (4)$$

Depending on the positions of the arm robot and of the moving obstacle, the collision avoidance algorithm must determine the joints that increase $dist_1$ and maintain constant $dist_2$. For a numeric example was considered the RESCUER platform [18] that has a 7 degree of freedom (DOF) arm robot and an obstacle, that is near the link 4 of the arm robot. The variation of the distances d_1 and d_2 can be determined by calculating the derivate of the distances (see tab. 1).

From table 1 results that the joints of the arm robot can be included in three categories, as follows:

- joints that have a big impact on $dist_1$ and small

impact on d_2 (joint 1 and joint 2);

- joints that have a big impact on d_1 and d_2 (joint 4);
- joints that have no impact on d_1 and d_2 (joint 3 and joints 5÷7).

From the joint classification, it results that not all the joints can be used to determine the solution. In order to increase the success chances of the algorithm, taking into account the previous two constraints, it should be used only joints from the first category.

After the algorithm determines the joints that can be used to find the solution, it should modify them in order to solve the following system of equations:

$$\begin{cases} d_1(\theta_a, \theta_b) = d_1^* \\ d_2(\theta_a, \theta_b) = d_2^* \end{cases} \quad (5)$$

where θ_a and θ_b are the angles of two joint from the first category and d_2^* is initial distance between the points P_{k+1} and P_d .

The solution of the system of equations 2 can be found using the *Netwon-Rapson* iterative method. The criteria function of the system is:

$$F(\theta_a, \theta_b) = \begin{bmatrix} d_1(\theta_a, \theta_b) \\ d_2(\theta_a, \theta_b) \end{bmatrix} \quad (6)$$

and it should be equal with:

$$F^* = \begin{bmatrix} d_1^* \\ d_2^* \end{bmatrix} \quad (7)$$

For the system of equations $F = F^*$ was determined the Jacobian:

$$J(\theta_a, \theta_b) = \begin{pmatrix} \frac{\partial d_1}{\partial \theta_a} & \frac{\partial d_1}{\partial \theta_b} \\ \frac{\partial d_2}{\partial \theta_a} & \frac{\partial d_2}{\partial \theta_b} \end{pmatrix} \quad (8)$$

The residue from the *Netwon-Rapson* method was considered:

$$\delta F = F - F^* \quad (9)$$

and resulted the following iterative equation:

$$q^{(i+1)} = q^{(i)} + J^{-1}(q^{(i)}) \delta F(q^{(i)}) \quad (10)$$

where

TABLE 1. The derivate of d_1 and d_2 .

	Joint (x)						
	1	2	3	4	5	6	7
$\frac{\partial d_1}{\partial \theta_x}$	-10.7	-9.8	0.1	-15.9	0.0	0.0	0.0
$\frac{\partial d_2}{\partial \theta_x}$	0.0	0.1	-0.1	14.1	0.0	0.0	0.0

$$\mathbf{q}^{(i)} = \begin{bmatrix} \theta_a^{(i)} & \theta_b^{(i)} \end{bmatrix}^T \quad (10).$$

The algorithm that calculates the collision free configuration of the arm robot with constant position and the orientation of the end-effector is outlined as Algorithm 1.

Algorithm 1 – Arm robot collision free configuration

```

determine  $\mathbf{P}_{s1}, \mathbf{P}_{s2}, \mathbf{P}_d, d_1^*, d_2^*$ ;
determine the collision point and  $\mathbf{P}_k, \mathbf{P}_{k+1}$ ;
determine initial joint values  $\theta_1^{(0)}, \theta_2^{(0)}, \dots, \theta_n^{(0)}$ ;
calculate the two joints  $a$  and  $b$ ;
set  $\mathbf{q}^{(0)} \leftarrow [\theta_a^{(0)} \quad \theta_b^{(0)}]^T$ ;
set  $\mathbf{F}^* \leftarrow [d_1^* \quad d_2^*]^T$ ;
 $i \leftarrow 0$ ;
repeat
    calculate  $\mathbf{F}(\mathbf{q}^{(i)}) \leftarrow [d_1(\mathbf{q}^{(i)}) \quad d_2(\mathbf{q}^{(i)})]^T$ ;
    calculate  $\delta \mathbf{F}(\mathbf{q}^{(i)}) \leftarrow \mathbf{F}(\mathbf{q}^{(i)}) - \mathbf{F}^*$ ;
    calculate  $\mathbf{q}^{(i+1)} \leftarrow \mathbf{q}^{(i)} + \mathbf{J}^{-1}(\mathbf{q}^{(i)}) \delta \mathbf{F}(\mathbf{q}^{(i)})$ ;
     $i \leftarrow i + 1$ ;
until  $i > \text{maxIterations}$  or every element of  $\delta \mathbf{F}$  is less than  $\varepsilon$ ;
if  $i < \text{maxIterations}$  then
    calculate the final solution:  $\theta_1, \theta_2, \dots, \theta_n$ 
    ( $\theta_a \leftarrow \theta_a^{(i)}, \theta_b \leftarrow \theta_b^{(i)}$  and for the other joints use the reverse kinematic model);
else
    the solution do not exist.

```

For the algorithms numeric simulation was used the Denavit-Harterberg model of the 7-DOF arm robot from the RESCUER platform (see fig 3). In the first example was considered that an object is getting closer to the link 2 of the arm robot.

Because the collision point belongs to the link 2 it resulted that $\mathbf{P}_k = \mathbf{P}_2$ $\mathbf{P}_{k+1} = \mathbf{P}_3$ and the joints that

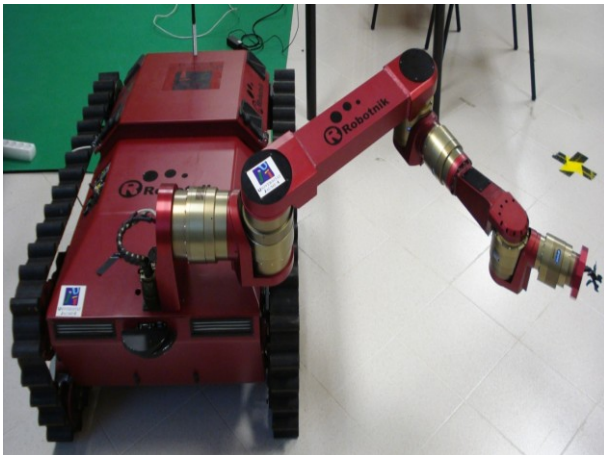


Figure 3. The RESCUER robotic platform.

should be modified are joint 1 and 2. In this case the joint selection was simple because joint 1 and 2 are the only ones that are between the robot base and the collision point.

Figure 2 shows the variation of the joints 1 and 2, respective the variation of the $\mathbf{F}$ function, during the algorithm execution. It can be observed that the variation of the angles θ_1 and θ_2 increase the distance between the moving object and the link 2 while the distance between the end of link 2 and the destination point of the end-effector remains constant. The algorithm stops when $d_1 - d_1^* < \varepsilon = 0.01$.

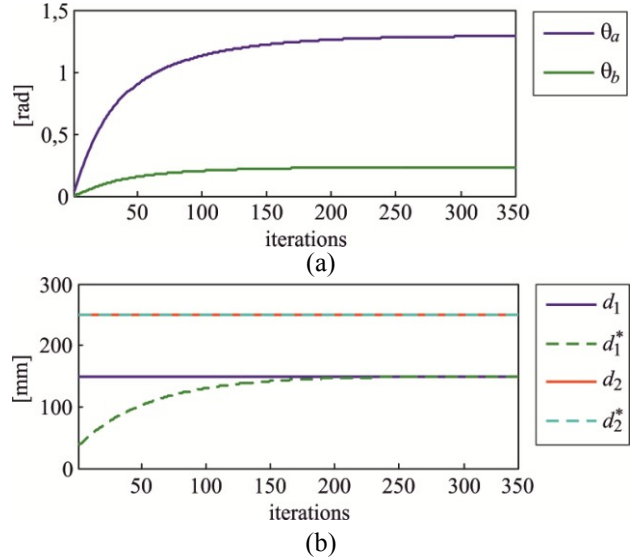


Figure 2. Simulation results of the algorithm 1. (a) the variation of the joint 1 and 2 positions; (b) the variation of d_1 and d_2 .

In the second example was considered that the intersection point belongs to the link 4. In this situation the points $\mathbf{P}_k$ and $\mathbf{P}_{k+1}$ are $\mathbf{P}_4$ respective $\mathbf{P}_5$ and the joints that can be modified are joint 1 to joint 4. From the available joints should the selected two that modifies the d_1 and maintain constant d_2 . In table 1 are presented the

derivates of the functions d_1 and d_2 on the robot joint angles. In the table can be observed that the joint 1 and joint 2 respect the above constraints. The simulation results for this case are similar with previous one.

The algorithm was also implemented in C++ development environment and the execution time was measured. During the practical tests, the algorithm found the solution in about 20÷30 mS. The time variation depends on the distance between the moving obstacle and the arm robot. These results enabled us to develop a real-time path planner for collision avoidance.

IV. THE PATH PLANNER

The arm robot control structure decides when the end-effector must be moved and generates its trajectory. In order to control the RESCUER arm robot, the control structure must send every 50mS to the joints their reference position.

Because initially the arm robot does not have any obstacle the control structure generates the end-effector trajectory and using the reverse kinematic model it generates the joint trajectories.

During the transition, if the control structure detects an moving obstacle, it corrects, using the algorithm 1, the trajectory of the arm robot joints. To respect this, the control structure has to calculate the following positions of the joints every 50mS using the algorithm 2, where was considered that:

- $q^{(k)}$ is the current positions of the joints;
- $q^{(k+1)}$ is the next positions of the joints;
- $P_d^{(k+1)}$ is the next position of the end-effector (from the end-effector initial trajectory);
- *invKine* is the standard inverse kinematic model of the arm robot.

Algorithm 2 – Collision avoidance joint trajectory generator.

```

if collision=TRUE
     $q^{(k+1)} \leftarrow \text{algorithm\_1}(q^{(k)}, P_{s1}, P_{s2}, P_d^{(k+1)});$ 
else
     $q^{(k+1)} \leftarrow \text{invKine}(P_d^{(k+1)}).$ 

```

V. PRACTICAL TESTS RESULTS

In the practical tests was considered that the robot should move in a straight line between the points A(24;0;10) and B(24;10;10) with smooth start and stop accelerations in 3 seconds (see fig. 4 (a)). Having the end-effector trajectory the control structure determines the initial joint trajectory (see fig. 4 (b)).

After the control structure starts the transition a cylindrical bar starts to move from point C(5;15;12) to D(5,5,12) with the velocity of 0.1m/sec. Initially the moving object is not near the arm robot components and the control structure uses the initial joint trajectory to control the arm robot (see fig. 5).

After 1 seconds the moving obstacle is near the link 2 of the arm robot and the control structure the algorithm 1 to change the arm robot joints trajectories in order to avoid the collision (see fig. 5) and maintain the same trajectory for the end-effector. The resulted joint accelerations are smaller than 1 rad*sec⁻². At the end of the motion the accelerations are different than 0 because the moving obstacle is getting closer to the arm robot and it has to avoid the collision.

In figure 6 is represented the configuration of the redundant arm robot during the collision avoidance procedure.

VI. CONCLUSIONS

This paper presents a flexible, efficient and robust scheme to avoid, in real-time, a collision of an articulated arm robot with a moving obstacle when it operates in dynamic environments with unforeseen changes.

The advantages of the current control structure are:

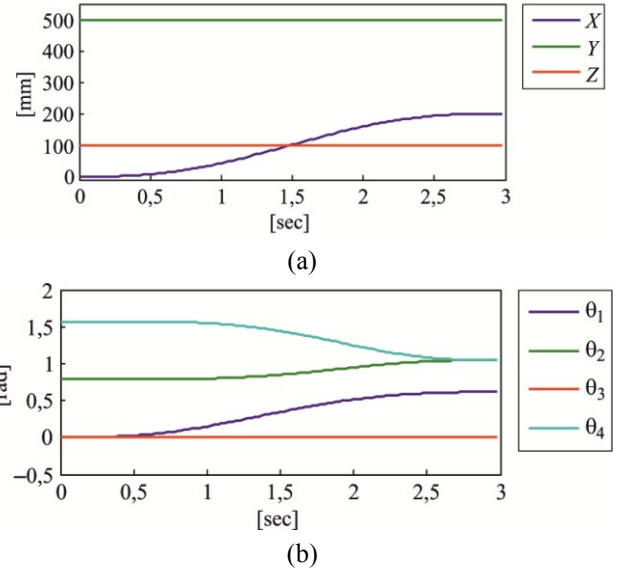


Fig. 4. The practical initial conditions: (a) the desired end-effector trajectory; (b) the initial joint trajectories (when there is no moving obstacle in the arm robot area).

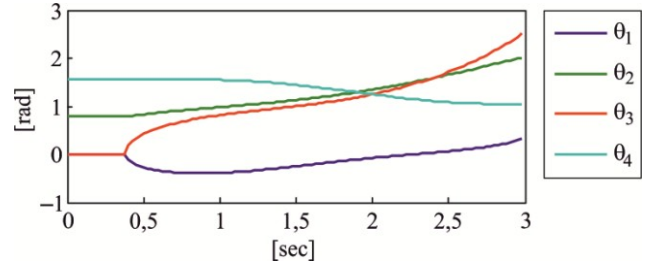


Fig. 5. The practical tests results: (a) the resulted joint accelerations.

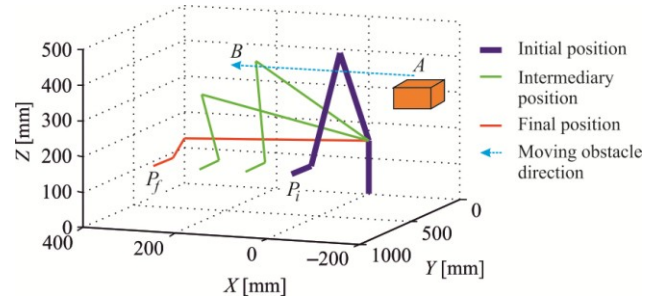


Fig. 6. The redundant arm robot configuration during the collision avoidance procedure.

- shape of the moving obstacle and of the arm robot components can be very complex (because the collision detection algorithm uses the OBBs interference detection method);
- the algorithm 1 determines the arm robot collision free configuration in real-time;
- the joint space trajectories can be modified during a transition;
- maintains the same end-effector trajectory, if it is possible.

The real-time path generator algorithm must be improved in the following areas:

- the joint stop acceleration are not 0m/sec;

- the algorithm 1 iterative method can be optimised in order to obtain smaller execution times.

ACKNOWLEDGEMENT

This paper is supported by the Sectoral Operational Programme Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the contract number POSDRU/6/1.5/S/6.

REFERENCES

- [1] T. Asfour, P. Azad, N., Vahrenkamp, K. Regenstein, A. Bierbaum, K. Welke, J. Schrder and R. Dillmann, "Toward humanoid manipulation in human-centered environments. Robotics and Autonomous Systems", vol. 56, no. 1, pp. 54-65, 2008.
- [2] T. Back, U. Hammel, and H.-P. Schwefel, "Evolutionary computation: Comments on the history and current state," IEEE Trans. Evol. Comput., vol. 1, no. 1, pp. 3–17, Apr. 1997.
- [3] O. Brock, O. Khatib, and S. Viji, "Task-consistent obstacle avoidance and motion behavior for mobile manipulation," in Proc. IEEE Int. Conf. Robot. Autom., May 2002, vol. 1, pp. 388–393.
- [4] P. P. Bonissone, R. Subbu, N. Eklund, and T. R. Kiehl, "Evolutionary algorithms + domain knowledge = real-world evolutionary computation," IEEE Trans. Evol. Comput., vol. 10, no. 3, pp. 256–280, Apr. 2006.
- [5] D. H. Eberly, "3D Game Engine Design", Ed. Morgan Kaufmann, pp. 561, 2001.
- [6] S. Gottschalk, M. C. Lin and D. Manocha "OBBTree: A Hierarchical Structure for Rapid Interference Detection" in Proc. Int. Conf. Computer Graphics and Interactive Techniques, August 1996, pp. 346-356.
- [7] S.M. Grigorescu, "Robust Machine Vision for Service Robotics", PhD thesis, Shaker Verlag, Aachen, Germany, 2010.
- [8] L. Jaillet and T. Simeon, "A PRM-based motion planner for dynamically changing environments," in Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst., 2004, vol. 2, pp. 1606–1611.
- [9] R. N. Jazar "Theory of Applied Robotics Kinematics, Dynamics, and Control", Ed. Springer, second edition, 2010.
- [10] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," IEEE Trans. Robot. Autom., vol. 12, no. 4, pp. 566–580, 1996.
- [11] S. M. LaValle and J. J. Kuffner, "Rapidly-exploring random trees: Progress and prospects," in Algorithmic and Computational Robotics: New Directions, B. R. Donald, K. M. Lynch, and D. Rus, Eds. Wellesley, MA: A K Peters, 2001, pp. 293–308.
- [12] T.-Y. Li and J.-C. Latombe, "On-line manipulation planning for two robot arms in a dynamic environment," in Proc. IEEE Int. Conf. Robot. Autom., May 1995, vol. 1, pp. 1048–1055.
- [13] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, 3rd ed. New York: Springer-Verlag, 1996.
- [14] J. Mbeye, S. Ma, Y. Toure, V. Graefe, and L. Zhang, "Robust neuro-fuzzy navigation of mobile manipulator among dynamic obstacles," in Proc. IEEE Int. Conf. Robot. Autom., May 2004, vol. 5, pp. 5051–5057.
- [15] P. Ogren, N. Egerstedt, and X. Hu, "Reactive mobile manipulation using dynamic trajectory tracking," in Proc. IEEE Int. Conf. Robot. Autom., Apr. 2000, vol. 4, pp. 3473–3478.
- [16] J. Tan and N. Xi, "Unified model approach for planning and control of mobile manipulators," in Proc. IEEE Int. Conf. Robot. Autom., 2001, vol. 3, pp. 3145–3152.
- [17] J. Vannoy and J. Xiao, "Real-Time Adaptive Motion Planning (RAMP) of Mobile Manipulators in Dynamic Environments with Unforeseen Changes" IEEE Trans. on Robotics, vol 24, no. 5, pp. 1199-1212, 2008.
- [18] ***, <http://www.robotnik.es/en/products/mobile-robots/rescuer>

Mobile Robot Control Using Face Recognition Algorithms

Alexandru Ionut Pustianu, Adriana Serbencu and Daniela Cristina Cernega

Abstract—In this paper is presented an algorithm to detect human faces using the orthogonal projection principle PCA (Principal Component Analysis) and a neural network classifier trained offline with a set of training images acquired from a webcam as well as from the CBLC and Manchester databases. The algorithm for the detection and tracking of faces will be implemented in a Visual C++ application, which will be tested on the mobile robot. The purpose of this paper is to implement an image processing algorithm in robotics thus making a human-robot interaction. The experimental results have been achieved on the PeopleBot, a two-wheeled differently driven mobile robot with a castor wheel for stability.

I. INTRODUCTION

Face detection is one of the most active research areas in computer vision, because of many interesting applications in areas such as security, surveillance, media recovery, and human-robot interaction. For example, face detection is combined with other modules in order to identify a person in a video sequence [1]. The face locations, the results of a face detection system can be used for applications such as face recognition and video indexing [2]. Although this area has been studied for more than 30 years, making a fast and robust face detection system that can implement the changes found in different faces, such as facial expressions, changes in lighting and low resolution [3]. Neural networks [4], [5], support vector machine (SVM) [6], [7],[8] and AdaBoost [9], [10], [11], [12], [13] are chosen to achieve robust face detectors.

In this paper an algorithm is used to detect faces using the orthogonal projection principle PCA (Principal Component Analysis) and a neural network classifier trained offline with a set of training images acquired from the webcam and the CBLC and Manchester database. The algorithm for the detection and tracking of faces will be implemented in a Visual C++ application.

II. HARDWARE AND SOFTWARE EQUIPMENT USED. THE PEOPLEBOT MOBILE ROBOT

PeopleBot is an intelligent autonomous mobile robot (see Fig. 1), capable of mapping the surrounding environment, to get back to its original position and perform sophisticated tasks of path planning. The internal structure is shown in Fig. 2.

This work was supported by CNCIS-UEFISCSU, Romania, project PNII-IDEI 506/2008.

A. I. Pustianu is student at the Faculty of Automatic Control, Computers, Electric and Electronic Engineering, Dunarea de Jos University of Galati, Romania alexandru.pustianu@yahoo.com

A. Serbencu and D.C. Cernega are with the Faculty of Automatic Control, Computers, Electric and Electronic Engineering, Dunarea de Jos University, Galati, Romania adriana.serbencu@ugal.ro; daniela.cernega@ugal.ro

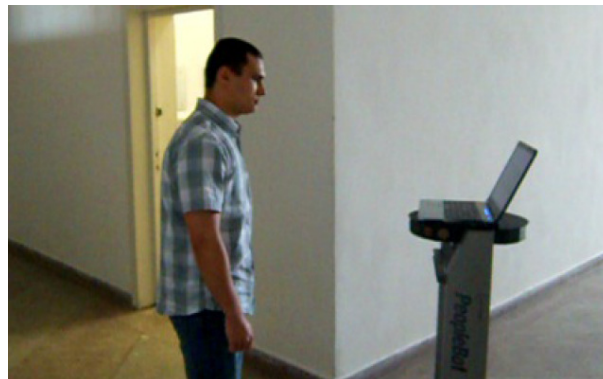


Fig. 1. PeopleBot mobile robot, used in the experiments

The program is developed using Visual Studio.NET 2008 [14] in C++ with the ARIA library (ActivMedia Robotics Interface for Application) for robot communication and control and a derivation of the OpenCV library for filters, normalization and image scaling. The neural networks were trained using a set of Matlab toolbox for neural networks with backpropagation gradient descent with momentum and adaptive learning rate.

Image acquisition from a webcam is performed using IsampleGrabber and DirectShow for raw data capture parts of the Windows SDK and Direct X SDK 2007 libraries. For these implementations is made an MFC application that supports a large number of web cameras, offers a preview of the video stream, capture image data in the array buffer that are displayed in a window using GDI+ classes. With this application as the core, it was able to implement the face tracking algorithm.

For simulations it used the simulation environment MobileSim [15], [16] (see Fig. 3) thus facilitating connection with a TCP port (similar to the serial port connection to the robot)

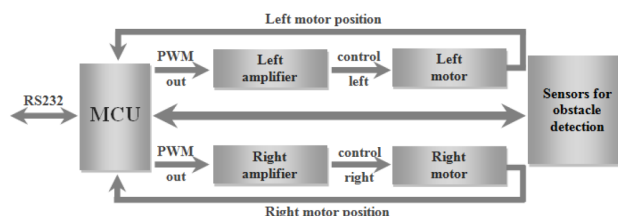


Fig. 2. Internal structure of PeopleBot mobile robot

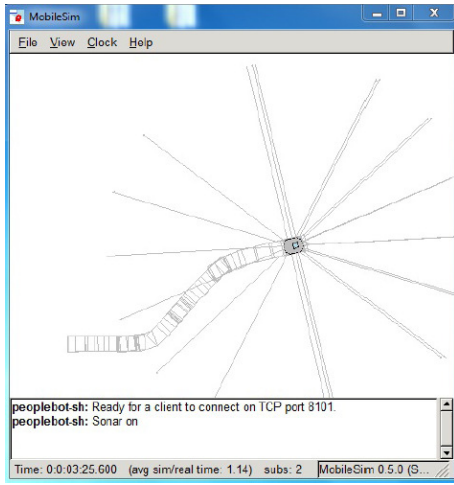


Fig. 3. MobileSim software

III. DESCRIPTION OF KINEMATIC MODEL, ORTHOGONAL PROJECTION PRINCIPLE AND NN CLASSIFIER

A. Kinematic Model of Mobile Robot

The kinematics of a two wheeled mobile robot are equivalent to those of a single rolling wheel. Therefore, this system is often referred to as the unicycle-type mobile robot. The kinematic modeling problem of different vehicle has been extensively treated in literature. Kinematic variables of the PeopleBot can be seen below in Fig. 4, b is the distance between the planes wheels and ω is the angular velocity of the vehicle. Assuming that the motion of the wheel is pure rolling without slipping, on a flat surface, the valid relations are:

$$\omega = \frac{v_R(t) - v_L(t)}{b/2} \quad (1)$$

$$v = \frac{v_R(t) + v_L(t)}{2} \quad (2)$$

The kinematics equations of the vehicle can be written:

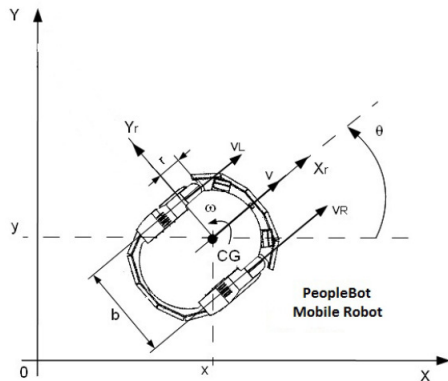


Fig. 4. Kinematic variables of two-wheel drive vehicle and a guiding wheel

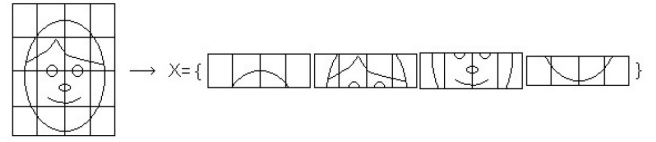


Fig. 5. Face vector formation from face image

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} 1/2 \cdot (v_R + v_L) \cdot \cos(\theta) \\ 1/2 \cdot (v_R + v_L) \cdot \sin(\theta) \\ (v_R - v_L)/(b/2) \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 1/2 \cdot \cos(\theta) & 1/2 \cdot \cos(\theta) \\ 1/2 \cdot \sin(\theta) & 1/2 \cdot \sin(\theta) \\ -b/2 & b/2 \end{bmatrix} \cdot \begin{bmatrix} v_R \\ v_L \end{bmatrix} \quad (4)$$

The nonholonomic constraint in ideal situations (without sliding) is:

$$\dot{x} \cdot \sin(\theta) - \dot{y} \cdot \cos(\theta) = 0 \quad (5)$$

B. Principal Component Analysis Orthogonal Projection and Neural Network Classifier

A face, which is an image, can be seen as a vector. If the image height and width is h pixels and w pixels, the number of components of this vector will be $h \times w$. Each pixel is encoded by a vector component. The construction of a vector image is made by a simple concatenation - image rows are placed side by side (Fig. 5).

Thus this vector belongs to an image space of all size images $w \cdot h$. All the faces look the same, they have two eyes, a nose, a mouth, etc.. Thus, all vectors faces are located in a narrow cluster in the image space. So, the whole image space

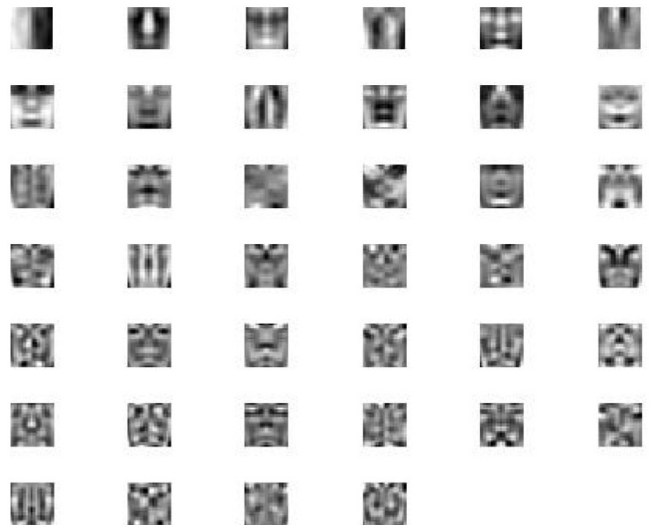


Fig. 6. Image reconstruction

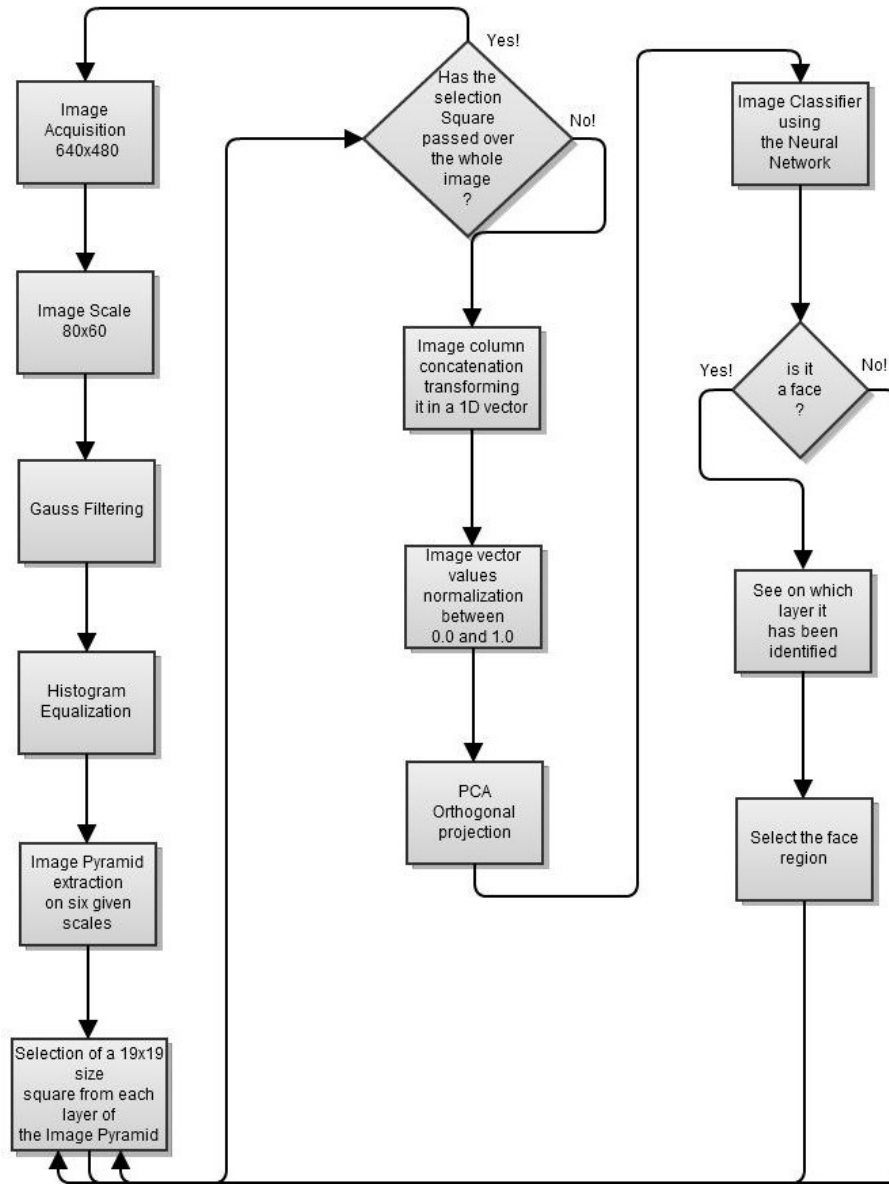


Fig. 7. Face detection algorithm

is not an optimal space for faces description. The task of this algorithm is to achieve a better face space to describe these faces. The basic vectors of this space are called principal components [17], [18].

In this algorithm, is used the face space and its main vectors to achieve orthogonal projection PCA (Principal Component Analysis). Thus it was chosen a training set of neural networks with non-faces and faces of 19×19 sizes that it have applied a Gaussian filter to reduce noise and an equalized histogram. Then from the matrix of size 19×19 it was made a column vector of dimension 361, by concatenating the columns.

In total it was used 800 face vectors and 4000 non-face vectors for PCA transformation. The transformation matrix size used to obtain the projection is 4800×361 . This matrix

was scaled after the orthogonal projection from PCA to 361×40 . The aim of this paper is to move in face space and extract the features of faces or patterns forming the so-called eigenfaces.

To form these eigenfaces we need to calculate the average mean of the values from the 4800×361 size matrix, to be then subtracted. After the subtraction, the matrix was multiplied with its transposed to the left side to form a square matrix, on which a covariance operation is applied to calculate eigenvectors and eigenvalues. These eigenvectors have the same size in number of components as the original image vectors, so that they can be seen as images. Thus the eigenvectors of the covariance matrix are called eigenfaces. To choose the 40 principal components, we select the eigenvectors with the largest associated eigenvalues.

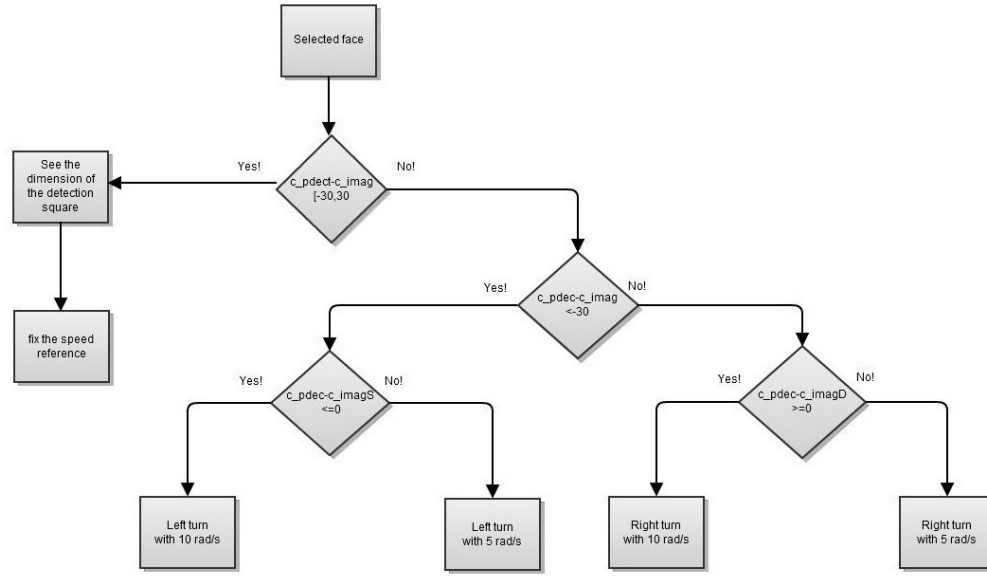


Fig. 8. Face tracking algorithm

Once the eigenfaces were calculated, each face in image space can be seen in face space. The transformation from image space to face space is simple:

Let:

- E be the matrix of the first 40 eigenfaces, where the first column is the first eigenface and so forth,
- f_I be a face in the image space, and
- f_F be the same face in the face space.

$$f_F = f_I \cdot E^T \quad (6)$$

Because the image space dimension is much larger than the size of the face space, the transformation introduces an error that is visible when looking at a reconstructed face (Fig. 6). The reconstruction is performed using the inverse transformation formula (6)

$$f_I = f_E \cdot E \quad (7)$$

To achieve the transformation from image space to face space (PCA projection) of images acquired from the webcam, it was packed the covariance matrix and the eigenvalues in a two-layer neural network with 361 neurons on one layer and 40 on the second. To finally achieve face detection it was created a neural network classifier with four layers, having 40, 20, 10 and a neuron respectively. For the training set, validation and testing it was used face and non-face images on which PCA orthogonal projection was applied. As for classification, the neural networks will calculate the Euclidean distance between the validated image and the image to be classified, by returning values of 0.1 for non-faces and 0.9 for faces, according to the Euclidean distance value.

IV. FACE DETECTION ALGORITHM

To achieve effective detection, in real time, it must first acquire an image from the webcam. The main constraint that appears at this moment is the image resolution, because the detection process uses a large number of mathematical calculations. So the best solution here is to scale the image by 8 times from a resolution of 640×480 to 80×60 (see Fig. 7).

Then, because it was used a single camera in different lighting conditions and the face detection takes more into account the pixel values in an image, a good solution to approximate the distance between the webcam and the target detection is to extract the image pyramid according to six predefined scales. The selection of a 19×19 square size starts with the top left edge of each layer of the pyramid image, after that it is transformed into a column vector and it is normalized with values between 0.0 and 1.0. The normalized column vector is projected in face space by the PCA projecting orthogonal to be classified by the neural network classifiers. If the algorithm found a face, it will be verify from which layer the selection square was taken, after that it frame it and select another square or a new image from the webcam, if not it will only select another 19×19 size square from the pyramid layers.

V. FACE TRACKING ALGORITHM

To implement the algorithm for face tracking of the robot (see Fig. 8), we must create a function that detects the face movement. The implementation of this function is simple: we save the area from where the identified face was taken, and then look for another one in the area around it. This is very useful because it considerably reduces the number of mathematical calculations related to the process of detection

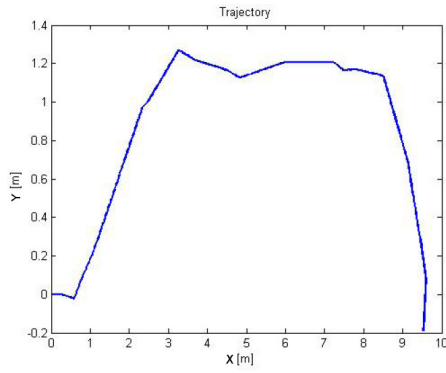


Fig. 10. Example of trajectory followed by the mobile robot

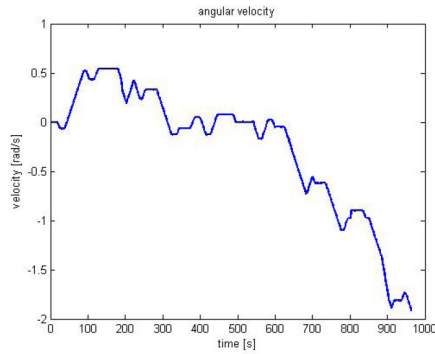
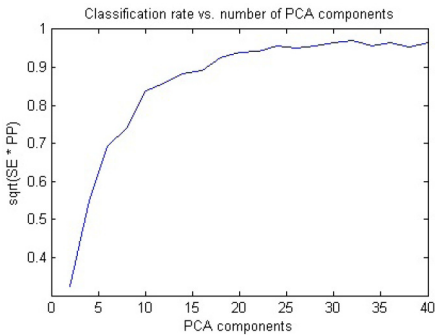


Fig. 11. Angular velocity of PeopleBot mobile robot



```

1139 se: 99.55
1140 sp: 100.00
1141 pp: 100.00
1142 np: 99.98
1143 ac: 99.98
1144 er: 0.000519
1145
1146 validation set: 447 2578
1147 se: 96.42
1148 sp: 99.90
1149 pp: 97.95
1150 np: 99.81
1151 ac: 99.72
1152 er: 0.001824
1153
1154 test set: 447 2578
1155 se: 97.32
1156 sp: 99.90
1157 pp: 97.97
1158 np: 99.86
1159 ac: 99.77
1160 er: 0.001746
1161
1162

```

face.nn* x pca.nn x rezultat.m x

Fig. 9. Neural network classifier performance

robot.

For detected face tracking by the robot, are taken into account both its physical constraints and the classical problem in image processing. The solution for the last one was implemented in the face detection algorithm by extracting the image pyramid, which provides a close approximation of that distance. Basically close objects appear larger than the ones in the distance.

Using an image pyramid with different scales as layers and extracting from each of these layers a 19×19 size selection square, one of these after classification is a face; after its framing and rescaling of the image from the layer the successfully detected, we can notice that the framed face has a larger or a smaller size. Thus we can estimate the relative distance according to six levels (six scales) for each scale, giving it a speed threshold.

For left/right tracking three virtual centers are created on the webcam acquired images: the center of the image, the center of the left side of the image and the center of the right side of the image. In light of these centers, the reference moving forward, rotate left and rotate right is taken. Thus it will make the difference between the acquired image center and the detected face framing square center, if its outcome is in the range $[-30, 30]$ then the robot will move forward. If not, the difference between the detection square center and the left side image center will be made, respectively the right side image center, resulting in a robot rotary motion to the left or respectively to the right.

VI. EXPERIMENTAL RESULTS

In the experimental results of the face detection algorithm for the training of the neural network classifier it was used the MATLAB `traindx` method because it provides the best classification results compared with other training algorithms, which tend to converge very quickly without a good performance generalization for the set of validation and test.

Although a rejection algorithm for non-face images was not taken in consideration, in Fig. 9 the robustness of the classification neural network is easily noticed, taking into account the sensitivity parameters (*se*), specificity (*sp*), positive and negative prediction (*pp* and *np*) accuracy (*ac*) and error (*er*). As for the classification rate per number of PCA components it can be observed that after the first 25 projections there is not any much change.

The humans operator position it is changed to observe the closed loop correction performed by the application. The algorithm calculates the center of destination only after it identifies the face. The trajectory and angular velocities of the PeopleBot mobile robot made from the experimentation of the face tracking and detection algorithm can be seen in Figs. 10 and 11.

Some experimental results are shown in Figs. 12.

VII. CONCLUSIONS AND FUTURE WORKS

This paper presents the implementation of data processing algorithms in robotics thus making a human-robot connec-

and helping to a smoother tracking of detection target by the

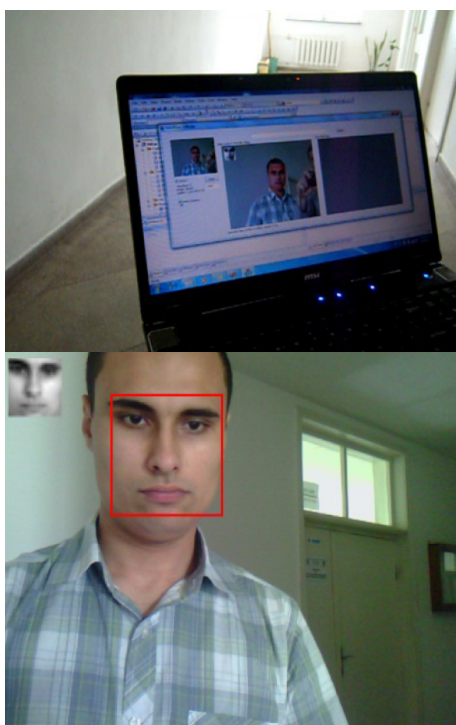


Fig. 12. Pictures from real experiments

tion. Experimental results show the reliability and robustness of the detection process, an example of face tracking by a mobile robot and the possibilities for future development of new practical applications under the "guardianship" of these methods. For the detection part, as well as for future activities, the authors would like to add to this algorithm a new neural network that will classify the RGB information, solving some of the problems caused by external lighting.

Another addition to this algorithm would be to use two SVM (Support Vector Machine) for non-rejection of face, thus ensuring a more robust detection and finally to classify faces in groups, the authors would implement a Kohonen neural network.

A further study of the PeopleBot mobile robot from an experimental point of view will allow us to design a new control algorithm for a more robust tracking of faces and with it the addition of voice communication.

VIII. ACKNOWLEDGMENTS

This work was supported by CNCIS-UEFISCSU, Romania, project PNII-IDEI 506/2008.

REFERENCES

- [1] S. Satoh and T. Kanade, Name-it: Association of face and name in video, in *Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 1997, pp.368373.
- [2] W. Zhao, R. Chellappa, P.J. Phillips, and A. Rosenfeld, Face recognition: A literature survey, *ACM Comput. Surv.*, vol.35, no.4, 2003, pp.399458.
- [3] M.H. Yang, D. Kriegman, and N. Ahuja, Detecting faces in images: A survey in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.24, no.1, Jan. 2002, pp.3458.
- [4] H. Rowley, S. Baluja, and T. Kanade, Neural network-based face detection, in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.20, no.1, Jan. 1998, pp.2338.
- [5] K.K. Sung and T. Poggio, Example-based learning for view-based human face detection, in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.20, no.1, Jan. 1998, pp.3951.
- [6] E. Osuna, R. Freund, and F. Girosi, Training support vector machines: An application to face detection, in *Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 1997, pp.130136.
- [7] S. Romdhani, P.H.S. Torr, B. Scholkopf, and A. Blake, Computationally efficient face detection, in *Proc. Intl. Conf. on Computer Vision (ICCV)*, July 2001, pp.695700.
- [8] B. Heisele, T. Serre, S. Prentice, and T. Poggio, Hierarchical classification and feature reduction for fast face detection with support vector machines, *Pattern Recognit.*, vol.36, no.9, Sept. 2003, pp.20072017.
- [9] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features, in *Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2001, pp.511518.
- [10] S. Li and Z. Zhang, Floatboost learning and statistical face detection, in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.26, no.9, Sept. 2004, pp.23 38.
- [11] C. Huang, H. Ai, B. Wu, and S. Lao, Boosting nested cascade detector for multi-view face detection, in *Proc. Intl. Conf. on Pattern Recognition (ICPR)*, Aug. 2004, pp.415418.
- [12] C. Liu and H. Shum, Kullback-leibler boosting, in *Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2003, pp.587594.
- [13] Y.Y. Lin and T.L. Liu, Robust face detection with multi-class boosting, in *Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2005, pp.680687.
- [14] <http://msdn.microsoft.com/en-us/visualc/default.aspx>
- [15] http://www.mobilerobots.com/Mobile_Robots.aspx
- [16] PeopleBot Operations Manual, <http://robots.mobilerobots.com/wiki/Manuals>.
- [17] J.E. Jackson, *A user's guide to principal components*. New York: Wiley, 1991.
- [18] S. Romdhani, *Face recognition using Principal Components Analysis*, Ph.D. Thesis, 1997.

Adaptive disassembly sequence control by using mobile robots and system information

A. Radaschin, A. Filipescu, V. Minzu, E. Minca, A. Filipescu Jr.

Abstract— Assembly and disassembly processes are similar from the viewpoint of automation. Assembly process is more or less a “classical” subject; disassembly process is more a quite new field. Flexible system are very important issue in the current industry when disassembling and recycling tasks have to be performed. These tasks can be performed by a human operator or by a robot system. In this paper an autonomous robot with manipulator system perform the required task for different operation. This system takes into consideration the necessary task to perform the disassembly of a component using robots synchronization with flexible line process. The proposed algorithm is to distribute the task among the robot and flexible line and takes into consideration the characteristics of each sequence that needs to be followed to perform the required disassembly of the product. The proposed system is validated by experiments using several types of components.

I. INTRODUCTION

CURRENTLY, flexible disassembly of mass products is an unsolved task, which, as part of an environmentally sound product policy and political regulations, is gaining more and more relevance [1]. To cope with large product quantities, the use of automated assembly systems with flexible disassembly tools seems to be a suitable approach.

In an assembly factory, different damaged products arrive continuously for disassembly, and individual decisions regarding optimal disassembly sequences [2] have to be made for every product. Detailed information on how to disassemble each arriving product is needed.

This information can be located centralized in databases or decentralized with the product.

This work of Adrian Radaschin was supported by Project POSDRU/88/1.5/S/61445.

Adrian Radaschin is PhD Student in Control Systems with the Department of Automation and Electrical Engineering, University “Dunarea de Jos” of Galati, Domneasca 47, 800008, Galati, Romania (e-mail: radaschinadrian@yahoo.com).

Adrian Filipescu is professor PhD supervisor with the Department of Automation and Electrical Engineering, “University Dunarea de Jos” of Galati, Domneasca 47, 800008 Romania (email: Adrian.Filipescu@ugal.ro).

Viorel Minzu is professor with the Department of Automation and Electrical Engineering, University “Dunarea de Jos” of Galati, (e-mail: Viorel.Minzu@ugal.ro).

Eugenia Minca is associate professor with the Department of Automation, Computer Science and Electrical Engineering, “Valahia” University of Targoviste 18 Unirii Av., Targoviste, Romania (e-mail: Eugenia.Minca@gmail.com).

Adriana Filipescu Jr. is master student in Control Systems with the Department of Automation and Electrical Engineering, University “Dunarea de Jos” of Galati, (e-mail: adriana_filipescu@yahoo.ca).

As a result of regulatory and consumer pressures, there has been an increasing emphasis on environmentally conscious manufacturing [3]. This involves the entire life cycle of products, from conceptual design to final delivery, and ultimately to the end-of-life (EOL) disposal of the products, such that environmental standard and requirements are satisfied [4]. A major element of EOL is product recovery which includes recycling and remanufacturing [5]. Both operation involve product disassembly in order to retrieve the desired parts and/or subassemblies [6].

The aim of this study is to bestow such an ability of planning or decision-making on robot assembly/disassembly task. Thus the planning level can be raised and the planning ability improved.

This research build upon some of the procedures for disassembly [7] and task planning as mentioned above. Issues will be addressed on how a detailed operation plan could be automatically [8] synthesized and simulated given a high-level description of a product to be desassembled.

This paper will develop a new expert Petri net model [9] for intelligent robotic with manipulator use for disassembly task planning and simulation at both high and low level [10]. In this paper, the expert Petri net principles of assembly/disassembly task representation and planning with special emphasis on the field of robotic desassembly planning will be illustrated.

II. ARCHITECTURE OF OUR SYSTEM

In figure 1 and 2, the architecture of the system is shown. It is composed of one an autonomous mobile robot, with two independent drive wheels, and an additional caster wheel.

The robot has its own odometric system, and an on-board embedded microcontroller is able to read the position information and to send it over a radio communication link, according to a specific protocol, and send the data to PLC on flexible line.

The robot is Pionner P3-DX, manufactured by MobileRobots and is equipped with manipulator with 3 articulations and one gripper paddles.

The flexible line is equipped with Siemens Simatic S7 300 PLC (Programmable Logic Controller), with 5 distributed modules connected by Profibus DP.



Fig 1. Robot Pioneer P3-DX



Fig 2. Flexible line for assembled/disassembled component

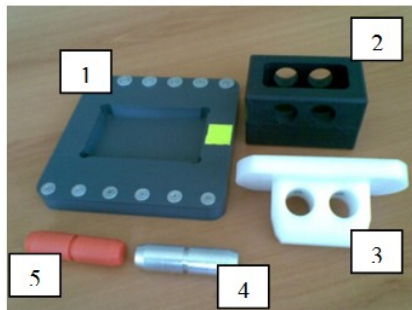


Fig 3. Product components to be assembled or disassembled



Fig 4. Fully assembled product

Flexible line includes five individual workstations with different tasks, carrying and transporting, pneumatic workstations, conveyor belt, sorting unit, test station and hay bay rack. The work part carrier is used for carrying and transporting the four-piece work part on conveyor belt

system. The work part carrier is equipped with 6-Bit identification which provides large number of possible codes, read out by inductive sensors. The four-piece work part enable workflow model such as assemblies, testing, sorting, storage and disassemblies.

III. THE NEED FOR PRODUCT INFORMATION

Before a product disassembly control sequence can be automatically generated, knowledge about the product, its components and their actual condition is needed. For each of the product's components, a decision has to be made whether to disassemble that specific component. The disassembly level depends on especially the actual condition of a component.

For the disassembly control sequence generation, the following aspects are relevance:

- Identification of the product and its components

As different products are allowed to arrive for disassembly all the time, a unique identification of the product to be disassembled is needed. In the case of plastic cylinder component (figure 3), the simple identification on product level is sufficient as a database may contain a detailed description of the product and its components.

When the product is unique or when parts of the product have been changed plastic or metal component in our case, more detailed information about the product and its components beyond the simple identification is needed.

- Configuration of component

For disassembly operation, the configuration of the product's components is needed. This includes position and orientation of components and the material it is made, plastic or metal, elements that are relevant for activation the disassembly operation. Here, disassembly precedence graphs and information about the components like needed disassembly tools are important for disassembly control sequence generation.

IV. TASK PLANNING

The disassembly strategy is based on the hierarchical model proposed in [12][13], which uses a graph representation of the product in which the relations among components are expressed by means of arrows. With this model, the relation existing between two components is represented with an arrow between the two nodes that symbolize the components, and the set obtained can be considered as a whole subassembly, which can be included as a new individual component in the model.

Using that model the task planner can determine the sequence of components that must be removed to achieve a specific sequence of tasks. If the target consists of the disassembly of a specific component, the task planner can provide the best sequence to reach the specific component [14][15]. If the target is the complete assembly of the product and found a bad component, the task planner

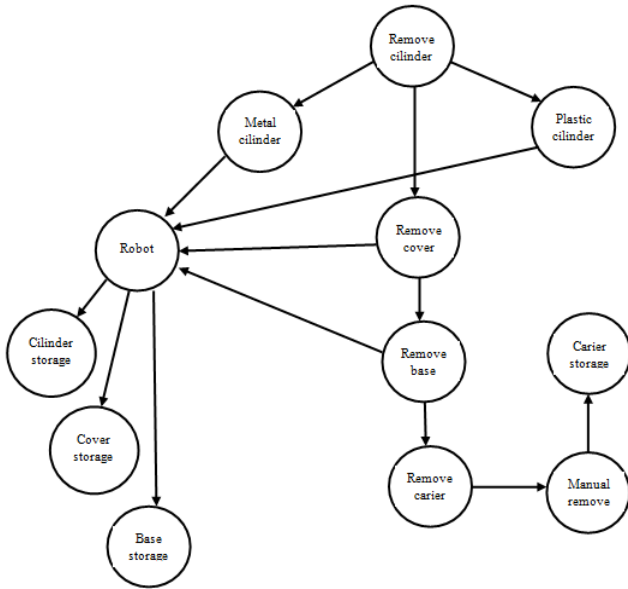


Fig 5. Disassembly task planning

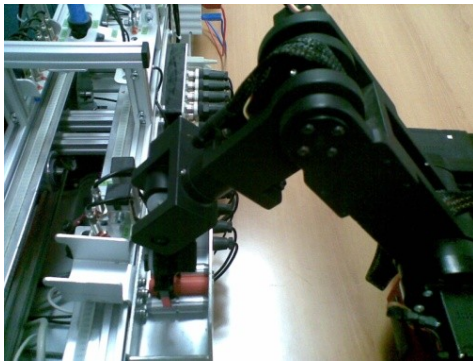


Fig 6. First disassemble operation – get cylinder

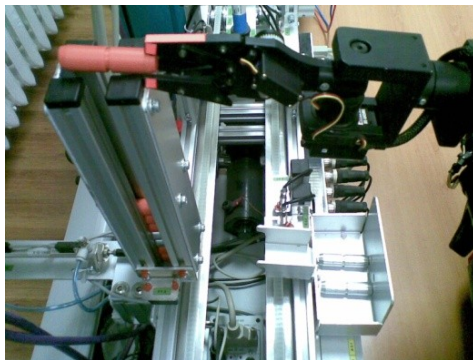


Fig 7. Second disassemble operation – introduction in storage

provides the best sequence completely to disassemble the product (see figure 3). Based on the sequence of tasks provided by the task planner, a group of rules are determined. Each task can be divided in one or more rules to disassembly a specific component. With these rules a

tree structure is created, which determines the order that must be followed to achieve the target in the disassembly planning.

Afterwards, when the task planner has determined the tree of actions, this information is used to distribute the actions among the robots that have to do the task (figure 5).

In figure 5 a tree with the tasks to be performed to completely disassemble the work part carrier with four-piece work. First of all, the work part carrier is run on conveyor belt system when the cylinder was remove by pneumatic piston. This task can be divided into the removing of the cylinder one by one or removing the plastic or metal cylinder. After removing the metal or plastic cylinder or both, the robot with manipulator is activated to get the cylinder one by one and put in to the storage.

On the other hand, after removing symbolize the components, and the set obtained can be considered as a whole subassembly, which can be included as a new individual component in the model.

V. NETS FORMALISM

Petri nets is a formalism that allows to model systems involving concurrency, resource sharing, synchronization and conflict, and allow to validate the correctness of the system by analyzing the qualitative properties of the net modelling the system.

A Petri net model of a discrete-event dynamic system is composed of a net structure with two disjoint sets of objects: places (represented as circles) that are the support of the state variables, and transitions (represented as bars) that represent the events. Places and transitions are connected by arcs following a weighted flow relation. The dynamics of the net structure is coded by a marking (an assignment of tokens to each place) and the marking evolution rule known as token game.

Formally, a Petri net is defined as a five tuple [11]:

$$N = \langle P, T, Pre, Post, A \rangle \quad (1)$$

where P is finite non-empty set of places, T is a finite non-empty set of transitions, Pre is the pre-incidence function $P \times T \rightarrow \mathbb{N}$, $Post$ is the post-incidence function $T \times P \rightarrow \mathbb{N}$, and A is the initial marking (the initial assignment of tokens to each place).

Pre and post-incidence functions can be represented by means of matrices of $|P|$ rows and $|T|$ columns. If the net has no self-loops then the net is fully characterized by a single incidence matrix $C = Post - Pre$, such that if there is an arc with weight $k \in \mathbb{N}$ from p_i to t_j then $C[i][j] = -k$, and if there is an arc with weight $k \in \mathbb{N}$ from t_j to p_i then $C[i][j] = k$. One of the method to analyze the dynamics feature of a Petri net is by means of the reachability tree, which is tree whose nodes are the marking that are

reachable by the transition to be fired to change from and marking to the next.

VI. OPTIMIZATION

Let the product to be assembled be composed of N parts, each one identified by a numeric label ranging from 1 to N . Then, the disassembly precedence matrix M is a $N \times N$ matrix that describes the precedence relationships between parts, i.e. determines which parts must be removed before being able to remove another one. The information for a given part i is codified in column i as follows:

- If parts j and k are AND precedents to part i , then $M[j][i]$ and $M[k][i]$ are set to the same positive integer value, and all the other rows are set to zero.
- If parts j and k are OR (XOR) precedents to part i , then $M[j][i]$ and $M[k][i]$ are set to different positive (negative) integer values, and all the other rows are set to zero.

This section presents an algorithm to automatically generate the precedence matrix. It makes use of a geometric tool that checks if there exists an infinitesimal motion that rigidly removes a polyhedral part from the assembled product.

Let:

- $M[i][\cdot]$ and $M[\cdot][j]$ represent, respectively row i and column j of M ;
- C be the incidence matrix of the Petri net;
- The set of transitions be composed of $N+2$ transitions, such that $t_{(N)}$ represents the disassembly of part i , and $t_{(N)}$ and $t_{(N)}$ represent the starting and the completion of the disassembly, respectively;
- The initial set of places be composed of $N+2$ places, such that $p_{(N)}$ represents the preconditions to removing part i , $p_{(N)}$ represents the beginning (i.e the assembled product) and $p_{(N)}$ represents the preconditions for the completion of the disassembly. The set of places is enlarged when modeling OR or XOR precedence relationships. The new places are labeled as $p_{(N)}$ with k starting at $(N+3)$;
- $t_{(N)}$ be label associated to transition $t_{(N)}$;
- When part j has no precedence the $M[\cdot][j]$ contains only zeros. In this case,

$$C[\cdot][t_{(N)}] = 1 \quad (2)$$

and

$$C[\cdot][t_{(N)}] = -1 \quad (3)$$

- When the removal of part i is no precedent to the removal of any part (only a precondition to the complete disassembly) then $M[i][\cdot]$ contains only zeros. In this case

$$C[p_{(N)}][t_{(N)}] = 1 \quad (4)$$

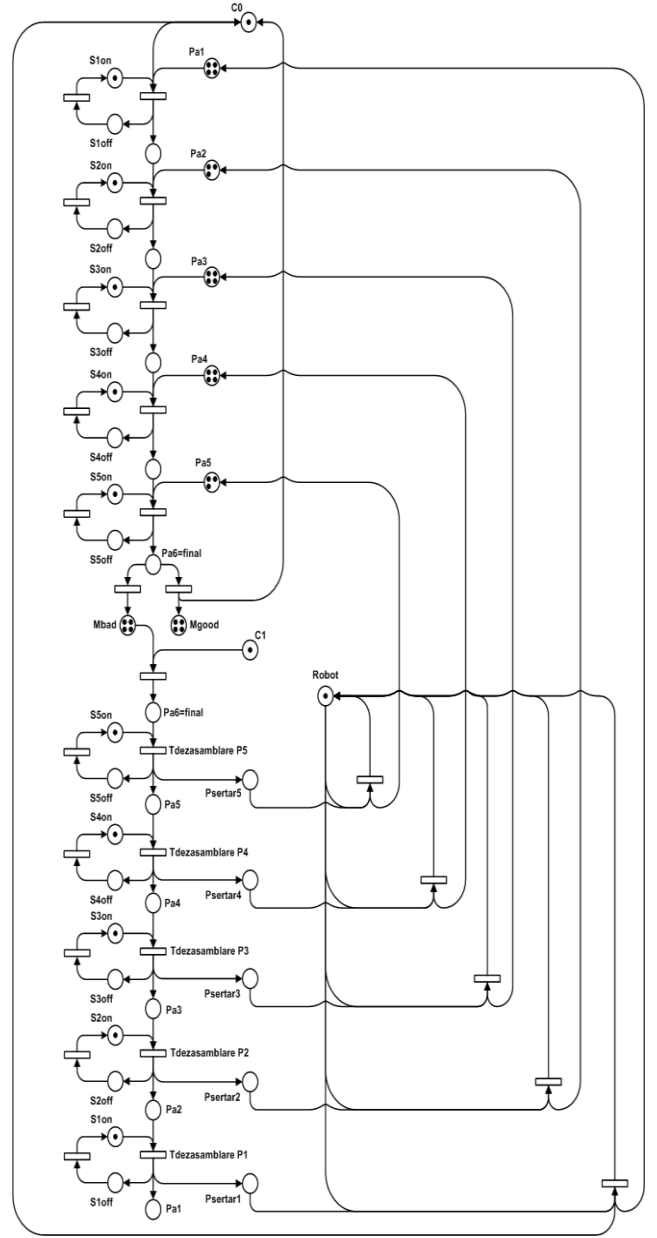


Fig 8. Petri net network of assembled/disassembled operation

- When the removal of R part with $i \in 1 \dots R$ are AND precedent to the removal of part j , then $M[\cdot][j] = L$ where $i \in 1 \dots R$ with $L = L$, i, j . In this case,

$$C[\cdot][t_{(N)}] = 1 \quad i \in 1 \dots R \quad (5)$$

and

$$C[\cdot][t_{(N)}] = R \quad (6)$$

- When the removal of any part with $i \in 1 \dots R$ is precedent to removal of part j , there is an OR precedence relationship. Then,

$$M[p][j]=L \quad i \in 1 \dots R \quad (7)$$

with

$$L \quad L \quad i \quad j \quad (8)$$

- In this case, a new place p is created,

$$C[p][j]=1 \quad (9)$$

$$C[p][t]=1 \quad (10)$$

with $i \in 1 \dots R$,

$$C[p][t_{(N)}]= -R + \quad (11)$$

and

$$C[i][j]=1 \quad (12)$$

and

$$C[i][t_{(N)}]= \quad (13)$$

- When the removal of any $i \in 1 \dots R$ (but not all of them) is precedent to the removal of part j , there is an XOR precedence relationship. Then $M[i][j]=L \quad i \in 1 \dots R$ with $L \quad k_i \quad L \quad i \quad j$. In this case, two new places p and p are created,

$$C[p][t]=1 \quad \forall \quad (14)$$

With $i \in 1 \dots R$,

$$C[p][t_{(N)}]=1 \quad (15)$$

$$C[p][t]=R1 \quad (16)$$

$$C[p][j]=1 \quad (17)$$

$$C[p][t]=1 \quad \forall \quad (18)$$

with $i \in 1 \dots R$;

$$C[p_{(N)}][t_{(N)}]=1 \quad (19)$$

$$C[p_{(N)}][t_{(N)}]=1 \quad (20)$$

and

$$C[p_{(N)}][t_{(N)}]=1 \quad (21)$$

The resulting Petri net is demonstrated to be safe, live and reversible (figure 8).

VII. CONCLUSION

The planning and operating of mainly manual disassembly systems has been a field of intensive research

for more than a decade. At the same time, highly flexible disassembly equipment has been developed, and commonly used manufacturing equipment has been applied for automated disassembly.

In comparison to manual disassembly, automated disassembly has to focus strongly on the planning of capacities and their utilization as well as on investment and operating cost, as they tend to be less flexible than workers and manual equipment. Due to the high proportion of fixed cost in automated systems and the high risk of idle capacity cost caused by low utilization or low availability, an efficient control of hybrid disassembly systems gains importance.

The control system of a hybrid disassembly system should be able to adapt to a high variety of disassembly objects, uncertain product conditions after usage and uncertain product amounts to be disassembled. In this paper, a conception for a product-driven control sequence generation for disassembly was presented, which allows coping with the disassembly object variety, different product conditions as well as product amounts.

The approach is directed to the planning level and cell control level. The conception for the generation of disassembly control sequences has three characteristics, i.e. modularity, parameterization and adaptation, in order to meet the high flexibility required for a disassembly system.

The procedure is implemented in three steps, i.e. process selection, operation planning and control sequence generation, which consider the availability of disassembly tools and the technological feasibility of disassembly processes and tools.

A modular control sequence will be generated by the programmed function unit. The conception for generation and transmission of control sequences has been verified by test bed applying the emulation approach.

The proposed approach, based on the assembly by disassembly concept, can cope with products with AND/OR/XOR relations and that require arbitrary infinitesimal motions to rigidly remove a part from the assembly.

Each transition of the Petri net represents the disassembly of a given part, and a variable cost is assigned to its firing, which depends on the previous firings performed, i.e. on the previous operations performed.

The search of a near-optimum sequence of robot tasks is done with an heuristic search of the reachability tree of the Petri net.

REFERENCES

- [1]Baldwin, D.F., Abell, T.E., Luici de Fazio, Whitney, 1991, „An integrated computer aid for generation and evaluation assembly sequences for mechanical products”. IEEE Transactions on Robotics until Automaton. 7, 78-94.
- [2]Choick, Zhaxfin and Lauws, 1998. „On the generation of product assembly sequences”. International Journal of Production Research. 36, 617-633.

- [3] Hoffman, R.L., 1991, „A Common sense approach to assembly sequence planning. In Computer-Aided Mechanical Assembly Planning”, edited by L. S. Homem de Mello and S. Lee (Boston, MA: Kluwer Academic Publishers), pp. 289-314.
- [4] Homem de Mello, L. S. and Sanderson, A.C., 1991, „A correct and complete algorithm for the generation of mechanical assembly sequences”. IEEE Transactions on Robotics and Automation., 7, 228-240.
- [5] J. Rosell, Assembly and task planning using Petri nets: „A survey and a roadmap towards autonomous robotic assembly systems,” technical report IOC-DT-P-2002-13, Univ. Politècnica de Catalunya, 2002.
- [6] M. Silva, „Practice of Petri Nets in Manufacturing”, ch. Introducing Petri nets, pp. 1-62. Chapman & Hall, 1993.
- [7] T. Cao and A. C. Sanderson, „AND/OR Net representation for robotic task sequence planning,” IEEE Trans. Syst., Man and Cybernetics - Part C: Applications and Reviews, vol. 28, pp. 104-218, May 1998.
- [8] A.J. Gambin, M.A. Piera, and D. Riera, „A Petri nets based object oriented tool for the scheduling of stochastic flexible manufacturing systems”, in Proc. of the 7th IEEE Int. Conf. on Emerging Technologies and Factory Automation, vol. 2, pp. 109–1098, 1999.
- [9] Seliger G, Grudzien W, Zaidi HN. „New methods of product data provision for a simplified disassembly”. In: Proceedings of the sixth international seminar on life cycle engineering, Kingston, Canada, June 21–23, 1999. p. 250–9.
- [10] Buchholz A, Franke C. „Assessment of standard components for extended utilization”. Proceedings colloquium e-ecological manufacturing, Berlin, March 27–28, 2003. p. 39–42.
- [11] Mikell P. Groover, “Automation, Production Systems and Computer Integrated Manufacturing”, 3rd Edition, Pearson Education, Inc., 2008.
- [12] David L. Setter, „Flexible manufacturing system (FMS): the investigative phase”, Published by Technical Communications, Kansas City Division, Allied-Signal Aerospace, 1993.
- [13] Kopacek B, Kopacek P (2001) „Semi-automatised disassembly”. Proceedings of the 10th international workshop on robotics in Alpe Adria Danube region, RAAD 01, Vienna, 16–18 May, 2001, pp 363.
- [14] Drinek M, Daichendt K, Zebedin H (2001) „A new concept for a fully automatized disassembly cell”. Proceedings of the 10th international workshop on robotics in Alpe Adria Danube region, RAAD 01, Vienna, Austria, 16–18 May 2001, pp 383– 388 370.
- [15] Kopacek B, Kopacek P (1999) „Robots for disassembly”. In: Proceedings of the 30th international symposium on robotics, Tokyo, pp 207–212.

Identifiability and identification of a class of hybrid systems

Lala H. Rajaoarisoa, Jean F. Balmat and Nacer K. M'Sirdi.

Abstract—The parameter identification of nonlinear systems where dynamics interact with discrete-event ones is a difficult problem. In this paper, we address this problem through the study of state estimation of hybrid systems. For this kind of systems, if the operating modes are detectable and observable, we demonstrate the parameters identifiability. An application is proposed and some results are given.

Index Terms—Hybrid system, state estimation, modeling, identification, observability, identifiability, detectability, persistently exciting.

I. INTRODUCTION

NO system identification algorithm can properly estimate the parameters of a system which is not identifiable [1]. For instance, in power systems [2], system-wide measurements of disturbances are frequently used in post-mortem analysis to gain a better understanding of system behaviour [3].

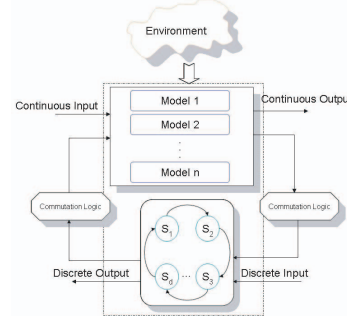
For the hybrid dynamical systems (HDS), the parameter estimation is one more complex task due to the interactions between continuous (smooth or none) and discrete dynamics [4], [5], [6]. Therefore, two alternative approaches are distinguished in the literature for HDS: either the operating conditions are fixed a priori, or they are estimated both with the sub-models. In the first case, finding a model representing the behavior of the system by observing a set of input-output data is straightforward, because data classification is easy to build, and estimation of the sub-models can be tackled by standard linear identification techniques as it shown in [8], [10], [11] and also argued in [9] and [23]. In the second case, the regions must be shaped to the clusters of data, and the strict relation among data classification, parameter estimation and regions for estimation must be verified.

In the literature we can find two subclasses of models to approximate continuous phenomena in nonlinear dynamics, namely Hinging hyperplanes ARX (HHARX) [14], [13], [7], and Piece Wise ARX (PWARX) models. These can be also solved via Mixed-Integer Linear Programming/ Mixed-Integer Quadratic Programming (MILP/MIQP) [15].

In this paper, hybrid system defined by a set of linear continuous model interconnected with discrete event atomic model are proposed and analyzed for observation and identification of nonlinear hybrid and/or switched systems (figure

1). The first model is necessary to describe the continuous dynamics of the real process and the second one is used to identify the operating modes of the global system. However, we will not present in this paper the system identification algorithm, but on the other side we will focus on the questions of identifiability and state reconstruction.

Fig. 1. Hybrid Complex System Representation



This paper is also organized as follows. Section 2 introduces the class of hybrid systems which interests us and reports several formulations of the identification problem for each model which defines the real process. Section 3 discusses on the observability and identifiability of each sub-model. Section 4 gives some experimental results for the identification of a greenhouse system. Conclusions and future work are given in Section 5.

II. HYBRID DYNAMICAL SYSTEMS MODELS

It is well-known that Hybrid Dynamical Systems (HDS) are systems composed of several interconnected sub-systems, introducing correlations, being subjected to commutations or to transitions (continuous or discrete).

Let us consider that the nominal behavior of this system is driven by multiple models orchestrated by events governing the system operation. This corresponds to some operating conditions for the main components or sub-systems in coordination of the discrete events. Thus, our system can be written as follows :

$$SS_{m_i} : \begin{cases} x(k+1) = f_{m_i}(x, u, w, k), & k \in \mathbb{R} \\ y(k) = g_{m_i}(x, k) \\ m_i = \langle Mode, I, O, \delta_{int}, \delta_{ext}, \lambda, t_a \rangle \end{cases} \quad (1)$$

where: f_{m_i} and g_{m_i} are non linear functions, $Mode$ is the set of sequential operating modes. I is the set of input event values (i.e. all the values that an input event can take). O is the set of output event values. $\delta_{int} : Mode \rightarrow Mode$ is the internal state transition function. $\delta_{ext} : Mode_- \times I \rightarrow Mode_+$ is the external state transition function. $\lambda : Mode \rightarrow O$ is the output function and t_a is the time advance function.

This work was not supported by any organization
L.H. Rajaoarisoa is with Research Institute on Embedded Systems, 76140 Saint Etienne du Rouvray, France. E-mail:lala.rajaoarisoa@esigelec.fr
J.F. Balmat is with Laboratory of Information and Systems Sciences, 83957 la Garde Cedex, France. E-mail:balmat@univ-tln.fr
N.K. M'Sirdi is with Laboratory of Information and Systems Sciences, 13397 Marseille Cedex 20, France. E-mail:nacer.msirdi@lsis.org

In linear and stationary cases, the class of system studied can be defined by three elements. The first one is a set of behavioral linear models, the second one is a transition or commutation device and the last one is a discrete event supervisor managing the commutations or transitions. This choice allows the relaxation of the partitioning number of the state-space (studied in [8]) thanks to the use of discrete event atomic models, and increases the possibility of dealing with multi-variable inputs and outputs using a piecewise affine model in state-space form. These coupled models can be described by the following relations :

$$\begin{cases} x(k+1) = A_{m_i}x(k) + B_{m_i}^e e(k) + B_{m_i}^w w(k) + v(k) \\ y(k) = C_{m_i}x(k) + D_{m_i}^e e(k) + D_{m_i}^w w(k) + \mu(k) \\ m_i = \langle Mode, I, O, \delta_{int}, \delta_{ext}, \lambda, t_a \rangle \end{cases} \quad (2)$$

where $v(k) \in \mathbb{R}^n$ and $\mu(k) \in \mathbb{R}^p$ are noise/error terms. $w(k)$ is the input which define the external environment. The real matrices/vectors $A_{m_i}, B_{m_i}^e, B_{m_i}^w, C_{m_i}, D_{m_i}^e$ and $D_{m_i}^w$, with appropriate dimensions, describe each affine dynamics. $Mode$ is the operating modes of the system (i.e. depending on the region of operation and the valid sub-model). The discrete system is in $Mode_{m_i}$ when operation is in the region Ω_{m_i} .

The events may be considered as defined by state $x(k)$ of the system and any internal $x(k)$ -dependent variable. Then, events depend on input values, noted $I(k)$, of the Mode m_i associated with the valid model [5]. As a consequence the validity domain Ω_{m_i} also depends on this input. λ is the output function which allows the activity execution. δ_{int} is the internal transition function. It ensures mode switches when no exogenous events occur before the elapsed time t_i and time advance of this mode t_a [16] : $Mode_j = \delta_{int}(Mode_i, t_i + t_a)$. δ_{ext} is the external transit function. This function is used when exogenous events come out [16] : $Mode_j = \delta_{ext}(Mode_i, t_i, I_i)$.

Here, one see that the hybrid nature of this structure requires at each time instant to determine the active local model, estimation of the time switching between two local models, estimation of the current state of the system. Thus, new forms of analysis like observability, and identifiability [17], [25], [26] is necessary because discrete changes are not handled well by continuous algorithm.

III. IDENTIFIABILITY OF HDS

This section addresses the problem of state estimation with the structure defined by the expression (2). Let us recall that for modeling phenomenon, it is important to know if the structure is well defined, in other words, all the parameters in the model can be estimated accurately with data provided. In the sequel, this observation is now formalized in terms of model identifiability.

Thus, in the following paragraph, we define identifiability for HDS. We discuss its relationship with event observations and continuous state output.

A. Discrete State Observability

To check the discrete state observability the following notions is introduced.

Definition 1: A discrete state is observable if the current state can be determined uniquely from the current state output and either the last or next event.

Thus, let us assume that:

Definition 2: The set of all operating modes $R_{I_i}^p$ where the system can evolve between an event is defined by

$$R_{I_i}^p = \{Mode_i \in Mode \mid \forall Mode_j \in Mode,$$

$$\exists I_i \in \Xi, Mode_j = \delta_{ext}(Mode_i, t_i, I_i)\}$$

Definition 3: The set of all operating modes $R_{I_j}^n$ representing all operating modes reached within an event is defined by

$$R_{I_j}^n = \{Mode_j \in Mode \mid \forall Mode_i \in Mode,$$

$$\exists I_j \in \Xi, Mode_j = \delta_{ext}(Mode_i, t_i, I_i)\}$$

so,

Proposition 1: A discrete event model is observable if:

(i) all operating modes Ξ are detectable, i.e $\Xi = Mode_0 \cup \{\emptyset\}$ and the dwell time of each operating mode $t_{a_{Mode_i}} > \varepsilon$, where ε is a positive real.

(ii) $|Mode \cap R_{I_i}^p| = 1$, i.e the previous event is available.

(iii) $|Mode \cap R_{I_i}^n| = 1$, i.e the next event is available.

(iv) $|R_{I_i}^p \cap Mode \cap R_{I_i}^n| = 1$ i.e the next and the previous event are available.

Proof: The first condition states that the unobservable events can be detected and thus validates that the current state output and the last (or the next) event are available to determine current state. If the condition (ii) (resp. (iii), (iv)) is true, there is only one state which has the current state output is reachable through the last event. Thus, the current state can be uniquely reconstructed. ■

B. Persistent Mode and Detectability

We have seen above that all operating modes can be detected (observable modes and/or unobservable modes). Now, if the sequence of discrete mode and switching times are observed and switching signal has suitable large dwell-time, then the linear switched systems can be identified by identifying each linear subsystem separately. Therefore, it is well-known that the input/output signals, classified into the mode $Mode_i$, must be changed sufficiently in order to excite the system for guaranteed that each subsystem has been identifiable. This observation can be associated with detectability notion of a persistent mode. Thus, we characterize this notion as follows for the class of system defined by the expression (2).

Definition 4: A persistent operating mode is detectable if we can determine, after a finite number of switching signal observations, the current continuous state of the system for some trajectories of the system.

However, before deriving the persistent excitation constraint (PEC) for the class of system considered, some definitions on excitation of input signals are given.

Definition 5: A scalar input signal u is strongly persistently exciting for order n if for all k there exists an integer T such that

$$\rho_1 I > \sum_{i=k}^{k+T} \begin{bmatrix} u_{i+n} \\ u_{i+n-1} \\ \vdots \\ u_{i+1} \end{bmatrix} \begin{bmatrix} u_{i+n} \\ u_{i+n-1} \\ \vdots \\ u_{i+1} \end{bmatrix}' > \rho_0 I, \quad (3)$$

where $\rho_0, \rho_1 > 0$.

Definition 6: A scalar input signal u is weakly persistently exciting for order n if

$$\rho_1 I \geq \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N \begin{bmatrix} u_{i+n} \\ u_{i+n-1} \\ \vdots \\ u_{i+1} \end{bmatrix} \begin{bmatrix} u_{i+n} \\ u_{i+n-1} \\ \vdots \\ u_{i+1} \end{bmatrix}' \geq \rho_0 I, \quad (4)$$

where $\rho_0, \rho_1 > 0$.

According to definition 5, a suitable constraint for MIMO-HDS is derived. The implementation is a priori straightforward as follows:

$$\rho_1 I > L_{mi} = \sum_{i=1}^m \sum_{j=0}^{N_{mi}} \begin{bmatrix} u_{i,j+n}^{mi} \\ u_{i,j+n-1}^{mi} \\ \vdots \\ u_{i,j+1}^{mi} \end{bmatrix} \begin{bmatrix} u_{i,j+n}^{mi} \\ u_{i,j+n-1}^{mi} \\ \vdots \\ u_{i,j+1}^{mi} \end{bmatrix}' > \rho_0 I, \quad (5)$$

where $\rho_0, \rho_1 > 0$. So if the input u is bounded, there always exists a positive scalar ρ_1 , so equation (5) is equivalent as the next one:

$$L_{mi} = \sum_{i=1}^m \sum_{j=0}^{N_{mi}} \begin{bmatrix} u_{i,j+n}^{mi} \\ u_{i,j+n-1}^{mi} \\ \vdots \\ u_{i,j+1}^{mi} \end{bmatrix} \begin{bmatrix} u_{i,j+n}^{mi} \\ u_{i,j+n-1}^{mi} \\ \vdots \\ u_{i,j+1}^{mi} \end{bmatrix}' > \rho_0 I, \quad (6)$$

and in a more compact form, we can write:

$$L_{mi} = \sum_{i=1}^m \sum_{j=0}^{N_{mi}} \Phi_{mi} \Phi_{mi}' > \rho_0 I. \quad (7)$$

In this equation, m defined the number of the input variable. Well, to tackle with extended PEC for HDS, we can formulate the following proposition:

Proposition 2: By definition, the input signal u is strongly persistently exciting for each operating mode m_i if:

- (i) the discrete event model is observable,
- (ii) there exists an integer $\rho_0 > 0$ such that

$$L_{mi} = \sum_{i=1}^m \sum_{j=0}^{N_{mi}} \Phi_{mi} \Phi_{mi}' > \rho_0 I$$

is satisfied.

Proof: This proposition relates the possibility to reconstruct each model associated to each operating mode. In fact, if first condition (i) is satisfied, then all operating modes are observable and the exact reconstruction of the continuous state is guaranteed. The second condition (ii) expresses that the experimental data contains enough information about the

dynamics of the system. Otherwise, a small-scale model can be defined to preserve the detectability and observability by removing modes not feasible. ■

C. Continuous State Observability

In this section, we consider observability regarding the continuous model. To verify observability properties, we just recall main results verifying the observability of the class of system considered [12] [19] [20].

So, a switching system is observable if:

Theorem 1: [20] $(C(k) - C(k'))A(k)$ is full rank for all $k \neq k' \in \{1, \dots, N\}$ and the dwell time $\tau_k \geq \varepsilon$ for all $k \geq 0$, then the initial state the switching times are observable if and only if for all $k \neq k' \in \{1, \dots, N\}$ we have $\text{rank}([\mathcal{O}_\varepsilon(k) \quad \mathcal{O}_\varepsilon(k')]) = 2n$.

Unfortunately Bemporad [7] show through counterexamples that observability properties cannot be easily deduced. Thus, an extension of this theorem is give in [24] to ensure the exact reconstruction of the state. This is based on the detectability and observability notion.

Theorem 2: [24] A switching system is observable if the following conditions are satisfied:

- (i) $S(q_i)$ is observable for any $q_i \in Q$
- (ii) $\forall p \in \Omega(\gamma), \forall q_i, q_j \in \gamma^{-1}(p), \exists k \in \mathbb{N} \cup \{0\} : C_i A_i^k B_i \neq C_j A_j^k B_j$

Theorem 3: [24] A switching system is detectable if the following conditions are satisfied:

- (iii) $\forall p \in \Omega(\gamma), \forall q_i, q_j \in \gamma^{-1}(p), \exists k \in \mathbb{N} \cup \{0\} : C_{i1} A_{i1}^k B_{i1} \neq C_{j1} A_{j1}^k B_{j1}$
- (iv) for all initial states $\begin{pmatrix} \xi_0 \\ q_0 \end{pmatrix} \in \mathcal{O}_i \times Q, i \in J$, and for any $\varepsilon > 0$, there exist $t > t_0$ such as $\|\xi(t, j)\| \leq \xi$ for any $t \geq t$.

With regard to the linear state-space model, the reconstruction of the continuous state puts down on the observability of the different behavioral models. Therefore, we can formulate our results like that.

Proposition 3: The continuous state is observable if:

- (vi) all linear sub-models are observable,
- (vii) the observability matrices $(\mathcal{O}^n)$ are full rank.

By combining three propositions (1), (2), (3), one gets the following results on the observability and identifiability of the global system.

For the class of considered system, with the assumption that the dwell time in operating mode is a positive number $t_{ek} \geq \varepsilon$ for all $k \geq 0$ [18]:

Proposition 4 (Globally identifiable from measurements):

A persistent model M defined by $M_i, i = 1, \dots, s$ is globally identifiable at $(\hat{A}_{mi}, \hat{B}_{mi}, \hat{C}_{mi}, \hat{D}_{mi})$ from measurements (u, y) if and only if:

- (1) Each persistent operating mode is detectable,
- (2) The switching system is observable,

Proof: The first condition states for the observability and detectability of a discrete state. It is equivalent to verify the observability of discrete event model and the detectability of a persistent operating mode. The second condition is associated to the observability of continuous state and validate that the whole operating modes are observable, detectable

and guaranteed the exact reconstruction of the continuous state. ■

IV. EXPERIMENTAL RESULT

In this section we give some results about the greenhouse modeling. The greenhouse is a nonlinear and non stationary system. It is equipped with many sensors which allow to measure the external and internal temperature, Te and Ti , (data expressed in $^{\circ}C$), the external and internal hydrometry He and Hi (data expressed in %), the wind velocity Vv (data expressed in m/s) and the global radiation Rg (data expressed in W/m^2). Moreover, many actuators allow to act on the heating Ch , the roofing OV , the moistening Br and the shutter Rd . We consider one day of March (10 March) to identify all parameters. Therefore, from expert knowledge and with a data analyse, we retain the variables Rg and Vv like descriptive variables of the environment where our system evolves. The choice of these variables have been done in a previous work [21], [22]. Moreover, these variables are necessary to design transition table (TABLE I) which is used for the discretization of the descriptive variables.

TABLE I
ENVIRONMENT SPECIFICATION DISCRETIZATION ($R_1 = 7.73m/s$,
 $R'_1 = 120W/m^2$ ET $R_2 = 250W/m^2$)

Input \ Values	Lower	Middle	Upper
Vv	$\leq R_1$	—	$> R_1$
Rg	$\leq R'_1$	$> R'_1$ and $\leq R'_2$	$> R'_2$

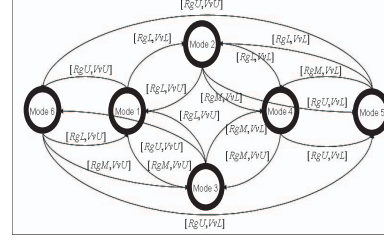
The wind speed takes two values $Vv = \{VvL, VvU\}$ which are respectively *Lower* and *Upper* values of Vv . The global radiation of the sun takes three values $Rg = \{RgL, RgM, RgU\}$ which are respectively *Lower*, *Middle* and *Upper*. Table II depicts the set of operating modes determined from the two descriptive variables.

TABLE II
DISCRETE OPERATING MODES OF THE GREENHOUSE SYSTEM.

Mode	Phase	Input values
$Mode_1$	NightWindLight	$I_1 = \{RgL, VvU\}$
$Mode_2$	NightWindStrong	$I_2 = \{RgL, VvL\}$
$Mode_3$	BreakWindLight	$I_3 = \{RgM, VvU\}$
$Mode_4$	BreakWindStrong	$I_4 = \{RgM, VvL\}$
$Mode_5$	DayWindLight	$I_5 = \{RgU, VvL\}$
$Mode_6$	DayWindStrong	$I_6 = \{RgU, VvU\}$

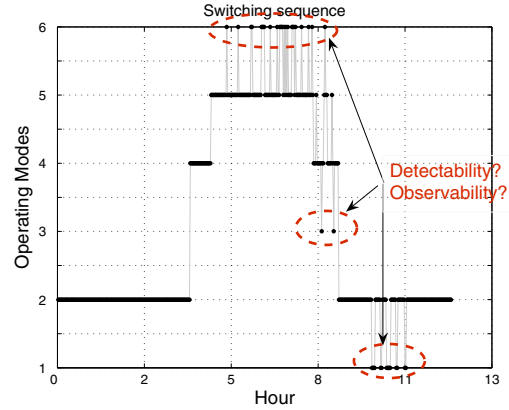
- All operating modes are defined by: $Mode = Mode_1, Mode_2, Mode_3, Mode_4, Mode_5, Mode_6$.
- Input variables are defined by the set $I = \{I_1, I_2, I_3, I_4, I_5, I_6\}$, associated respectively with the input events set $ev = \{ev_1, ev_2, ev_3, ev_4, ev_5, ev_6\}$.
- The set of corresponding output events is described by $O = \{O_1, O_2, O_3, O_4, O_5, O_6\}$.

Fig. 2. Atomic discrete events model.



The sub-models are defined in three categories: *day*, *night*, and *daybreak*. Each category contains two classes: *Cold* and *Fresh*. This leads us six sub-models for six operating modes. The corresponding atomic model is depicted in figure (2). The simulation done with the discrete events atomic model let us to have an idea on the detectability and observability of all operating modes of the system and the corresponding time range (figure 3). This result shows us a priori that our system is completely observable (resp. identifiable). To according with the proposition (4) given in the last section, our system loses its observability for several not persistently operating modes. For instance, the modes 6, 3 and 1. However, a small-scale model can be defined to preserve the observability and the identifiability by removing modes not feasible.

Fig. 3. Observability and Detectability Analysis with 6 Operating.



Therefore, the discrete events atomic model is composed of operating modes defined by $Mode^* = \{Mode_1^*, Mode_2^*, Mode_3^*\}$, input variables defined by the set $I^* = \{I_1^*, I_2^*, I_3^*\}$ associated respectively by the input events set $ev^* = \{ev_1^*, ev_2^*, ev_3^*\}$ and the set of output events is described by $O^* = \{O_1^*, O_2^*, O_3^*\}$. Thus, we have the following models for each mode:

TABLE III
NOVEL DISCRETE MODE DESIGNATION OF THE SUPERVISION DEVICE

Mode	Phase	Input variable
$Mode_1^*$	FreshNight	$I_1^* = \{RgL, VvL\}$
$Mode_2^*$	FreshDaybreak	$I_2^* = \{RgM, VvL\}$
$Mode_3^*$	ColdDay	$I_3^* = \{RgU, VvL\}$

This is illustrated in figure 4. Now, we can identify the

continuous dynamical models parameters according to the algorithm described above.

Fig. 4. Atomic discrete events model with observability and identifiability constraints.

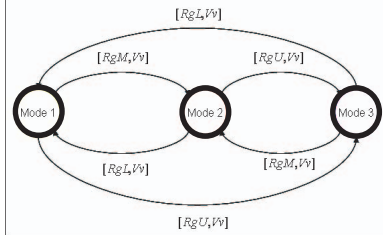
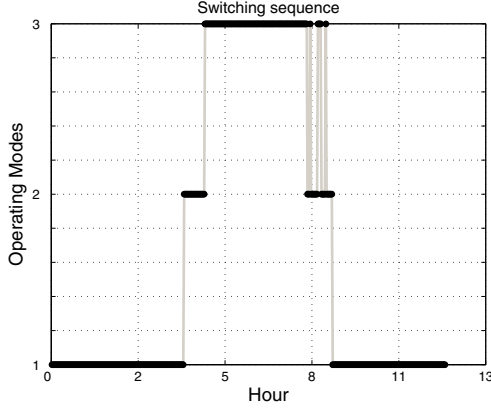


Fig. 5. Equivalent Operating Modes Commutation for the Identification Procedure.



for the mode: $Mode_1^*$

$$A_{Mode_1^*} = \begin{bmatrix} -0.2128 & 0.0819 \\ 0.1259 & -0.2162 \end{bmatrix}; C_{Mode_1^*} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix};$$

$$B_{Mode_1^*}^u = \begin{bmatrix} 0.1469 & 0 & 0 & 0 \\ -0.1480 & 0 & 0 & 0 \end{bmatrix};$$

$$B_{Mode_1^*}^v = \begin{bmatrix} 0.0848 & -0.0679 & -0.1323 & 0 \\ 0.0109 & 0.1365 & 0.1780 & 0 \end{bmatrix};$$

for the mode: $Mode_2^*$

$$A_{Mode_2^*} = \begin{bmatrix} -0.2694 & -0.3000 \\ 0.3795 & -0.4274 \end{bmatrix}; C_{Mode_2^*} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix};$$

$$B_{Mode_2^*}^u = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix};$$

$$B_{Mode_2^*}^v = \begin{bmatrix} 0.3946 & 0.3160 & -0.0630 & 0.1335 \\ 0.5616 & 0.6740 & -0.1797 & -0.9603 \end{bmatrix};$$

and for the mode: $Mode_3^*$

$$A_{Mode_3^*} = \begin{bmatrix} -0.3235 & -0.0513 \\ -0.2581 & -1.1357 \end{bmatrix}; C_{Mode_3^*} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix};$$

$$B_{Mode_3^*}^u = \begin{bmatrix} 0 & -0.1304 & -0.2003 & -0.1020 \\ 0 & 0.5993 & -0.5851 & 0.0743 \end{bmatrix};$$

$$B_{Mode_3^*}^v = \begin{bmatrix} 0.3186 & 0.1036 & -0.0076 & 0.2572 \\ 0.9125 & 0.9130 & 0.2010 & -0.1596 \end{bmatrix};$$

To validate our model we considered three days of March. The data of two days are very similar (March 11th and March 15th) while for the third day they are very different (March 12th).

The figure 6 gives us the result of simulation between the internal temperature and hydrometry of the estimated

and measured values. For the same way, we can show the effectiveness of this approach, in modeling case while considering only 3 modes where our system is observable and detectable. The criterium performance is given by the Variance-Accounted-For (VAF):

$$VAF = 1 - \frac{\text{var}(y(k) - \hat{y}(k))}{\text{var}(y(k))} * 100. \quad (8)$$

The performance between the hybrid dynamical model and a single global model are compared in table IV. We may notice that hybrid model is more efficient than global model.

TABLE IV
PERFORMANCE INDEX FOR PARAMETERS ESTIMATION

Used model	VAF-11 March	VAF-12 March	VAF-15 March
Hybrid model	80.9783	47.0203	79.6342
Single global model	63.0511	0	74.3790

The observability property is verified if $C_i * A_i^k * B_i \neq C_i * A_i^k * B_i^*$.

	$C_1 * A_1^k * B_1$	$C_2 * A_2^k * B_2$	$C_3 * A_3^k * B_3$
$k = 1$	0.0071	0	0.1466
$k = 2$	-0.003	0	-0.092
$k = 3$	0.001	0	0.0824
$k = 4$	-0.000337	0	-0.0876

To conclude about the observability analysis, let us check the rank of the observability matrix of each mode. We can remark that $\text{rank}(\mathcal{O}_{mi}^n)$ will be always equal to the sub-space dimension of the observable state $n = 2$. Thus, the system is locally observable but also globally observable between the switching sequence.

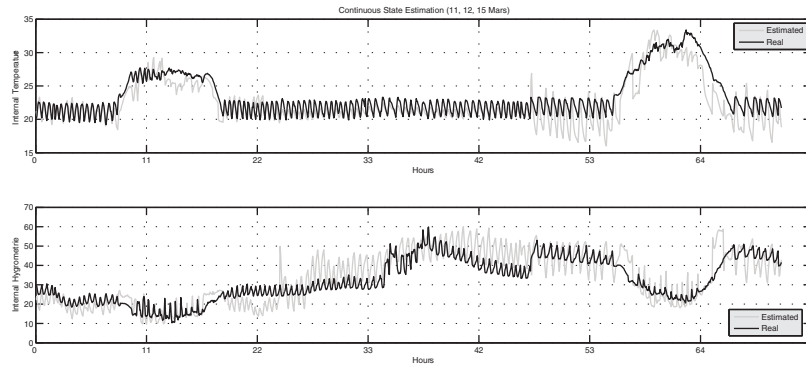
V. CONCLUSIONS

In this paper an approach to model a MIMO hybrid dynamical system is proposed and a class of systems is defined. An interconnection between a discrete atomic model and a set of state-space linear models allows to describe these systems. The atomic model allows to identify the operating modes of the system evolution. The state-space linear models allow to approximate continuous phenomena of the dynamical system.

Observability and identifiability properties are studied and some results are shown. We demonstrated that identifiability of a hybrid system depends on observability and detectability of the discrete states. Of course, continuous models must be observable. A persistent model of the experimental greenhouse presents the effectiveness of the methodology.

In future work, we will furthermore address a characterization of the set of observationally equivalent models. Finally, a study of the observability and identifiability conditions in the presence of noise will be made.

Fig. 6. Modeling and identification validation.



REFERENCES

- [1] L. Ljung, *System identification. Theory for the User*, Prentice Hall, Englewood Cliffs, NJ, 1987.
- [2] V. Donde, Development of Multiple Lyapunov Functions for a Hybrid Power System with a Tap Changer, *ECE 497DL*, Spring, 2001.
- [3] R.H. Craven, T.George, G.B. Price, P.O. Wright, I.A. Hiskens, Validation of dynamic modelling methods against power system response to small and large disturbances, in *Proc. of CIGRE General session*, Paris, August 1994.
- [4] E.D. Sontag, Nonlinear regulation: The piecewise linear approach. *IEEE Transactions on Automatic Control*, Vol. AC-26, No. 2, April 1981.
- [5] I. Baruch, F. Thomas, R. Garrido and E. Gortcheva, A Hybrid Multi-model Neural Network for Nonlinear Systems Identification, 1996.
- [6] B. P. Zeigler, *Theory of modeling and simulation*, Wiley and Sons, New York, NY, 1976.
- [7] A. Bemporad and M. Morari, Control of systems integrating logic, dynamics, and constraints, *Automatica*, Vol. 35, No. 3, pp. 407-427, 1999.
- [8] V. Verdult, M. Verhaegen, Subspace Identification of Piecewise Linear Systems. In *Proc. of the 43rd IEEE Conference on Decision and Control*. Atlantis, Paradise Island, Bahamas, December 14-17, 2004.
- [9] J. Roll, A. Bemporad, and L. Ljung, Identification of Piecewise Affine Systems via Mixed-Integer Programming'. *Automatica*, Vol. 40, No. 1, pp. 37-50, 2004.
- [10] L. H. Rajaoarisoa, N. K. M'Sirdi, E. K. Boukas and A. Naamane, Subspace Techniques for Identification of Hybrid Complex System. In *Proc. of the 16th Mediterranean Conference on Control and Automation, MED'08*, Ajaccio, Corsica, France, June 25-27, 2008.
- [11] P. Van Overschee, Subspace identification of linear systems: theory, implementation, application. *Phd Thesis*, Katholieke Universiteit Leuven, 1995.
- [12] P.J. Ramadge and W.M. Wonham, Supervisory control of a class of discrete-event processes'. *SIAM Journal of Control and Optimization*, Vol. 25, pp. 206-230, 1987.
- [13] G. Ferrari-Trecate, D. Mignone and M. Morari, Moving horizon estimation for Piecewise Affine Systems. *Proceedings of American Control Conference*, Arlington (VA), US, 2001.
- [14] L. Breiman, Hinging hyperplanes for regression, classification and function approximation, *IEEE Trans. on Information Theory*, Vol. 39, pp. 999-1013, 1993.
- [15] A. Bemporad, J. Roll and L. Ljung, Identification of Hybrid Systems via Mixed-Integer Programming'. In *Proc. of the 40th IEEE Conference on decision and control*, Orlando USA, 2001.
- [16] N. Giambiasi, N. R. Doboz and E. Ramat, Introduction à la modélisation à événements discrets', *Rapport interne*, LSIS UMR CNRS 6168, Lille Calais, No LIL-2002-02, 2002.
- [17] I. A. Hiskens, Identifiability of Hybrid System Models, *Proceedings of the 2000 IEEE International Conference on Control Applications*, Anchorage, Alaska, USA, September 25-27, 2000.
- [18] N. K. M'Sirdi, L. H. Rajaoarisoa and A. Naamane, Modélisation, Identifiabilité et Observabilité pour le diagnostic et la commande d'une classe de systèmes Complexes à Structure Variable', *Ouvrage Collectif suite à Qualita07: Diagnostic des systemes complexes*, pp.83-98, 2008.
- [19] A. Balluchi, L. Benvenetu, T. Villa, H. Wong-Toi and A. L. Sangiovanni-Vincentelli, Controller synthesis for hybrid systems with lower bounds of event separation. *Proceedings 38th Conference on decision and control*, Phoenix Az., pp.3984-3989, 1999.
- [20] S. Chaib, D. Boutat, A. Benalia and J.P. Barbot, Observability of the discrete state for dynamical piecewise hybrid systems. *Elsevier Nonlinear Analysis*, vol. 63, pp. 423-438, 2005.
- [21] L. H. Rajaoarisoa and N. K. M'Sirdi, Modeling for Control and Diagnosis for a Class of Non Linear Systems, *Sciences et Techniques de l'Automatique STA06*, Tunisie, Novembre 2006.
- [22] L. H. Rajaoarisoa, N. K. M'Sirdi, E. K. Boukas, J. F. Balmat and J. Duplaix, Greenhouse Climate Modeling: Observability and Identification for Supervision. In *Proc. of the 5th International Conference on Informatics in Control, Automation and Robotics, ICINCO 08*, Portugal, 11-15 May, 2008.
- [23] M. Verhaegen, Identification of the deterministic part of MIMO state space models, *Automatica*, vol. 30, pp. 61-74, 1994.
- [24] De Santis, E., Di Benedetto and M.D., Pola, G., On observability and detectability of continuous-time linear switching systems, in *Proc. of the 42nd IEEE Conference on Decision and Control*, Vol. 6, pp. 5777 - 5782, 2003.
- [25] P. Collins and J.H. Van Schuppen, Observability of Piecewise-Affine Hybrid Systems. In: *R. Allur and G.J. Pappas (eds.): HSCC, LNCC 2993*, pp. 265-279, 2004.
- [26] R. Vidal, A. Chiuso, S. Soatto, and S. Sastry, Observability of linear hybrid systems. In *Hybrid Systems: Computation and Control, LNCS*, pp. 526-539. Springer Verlag, 2003.

Electric Wheelchair Control for People with Locomotor Disabilities Using Eye Movements

George Constantin Rascanu and Razvan Solea, *Member, IEEE*

Abstract—The paper describes a control method of a robot, by means of an electric wheelchair, dedicated to severely disabled persons, equipped with a low-cost web camera, using only eye movements and gaze direction. The camera is positioned in front of the user's right eye so that the iris is captured as it moves. This movement is interpreted to obtain the command which is sent to the wheelchairs actuators. This command can be "FORWARD", "LEFT" or "RIGHT", when the eye (the iris) is detected on center, left and right respectively. For safety reasons, emergency stop, with the command "STOP", is caused by failure to detect the iris. This command is generated when the user encounters a close obstacle or when he wishes to shut down the system.

I. INTRODUCTION

The ability to move freely is highly valued by all people. However, it is sometimes difficult for a person with locomotor disabilities to do this, so an electric wheelchair is the only solution.

Users of wheelchairs are estimated to be about 200 million people worldwide. In the U.S.A., roughly 11000 people are affected by spinal cord injury every year, with 47 percent of those accidents leading to quadriplegia [1].

While traditional electric wheelchairs (a class of Wheeled Mobile Robot - WMR [2], [3]), build to facilitate maneuvering in indoor and outdoor environments, have helped millions of people with disabilities, many of them do not fully serve certain categories of persons with disabilities [4].

The assistive technology developed in the last years will significantly impact and improve the life of wheelchair users affected by partial or complete paralysis. Many of these systems, which are still in the prototype stage, were placed on the market for sale [5].

This paper sets out the design and development of a control system of a semi-autonomous robot, for people with severe disabilities, using only eye movements.

The system allows the processing of information coming from the user through a CMOS sensor camera, and interprets this information in order to designate the desired movement.

The camera is attached in front of the eye with using a mounting system and is equipped with a homemade Infra-Red night vision system for indoor environments and also it has an on/off button for outdoor environments.

This work was supported by CNCIS-UEFISCSU, Romania, project PNII-IDEI 506/2008

G. C. Rascanu is student at the Faculty of Automatic Control, Computers, Electric and Electronic Engineering, Dunarea de Jos University of Galati, Romania rascanu.george@yahoo.com

R. Solea is with the Faculty of Automatic Control, Computers, Electric and Electronic Engineering, Dunarea de Jos University, Galati, Romania razvan.solea@ugal.ro

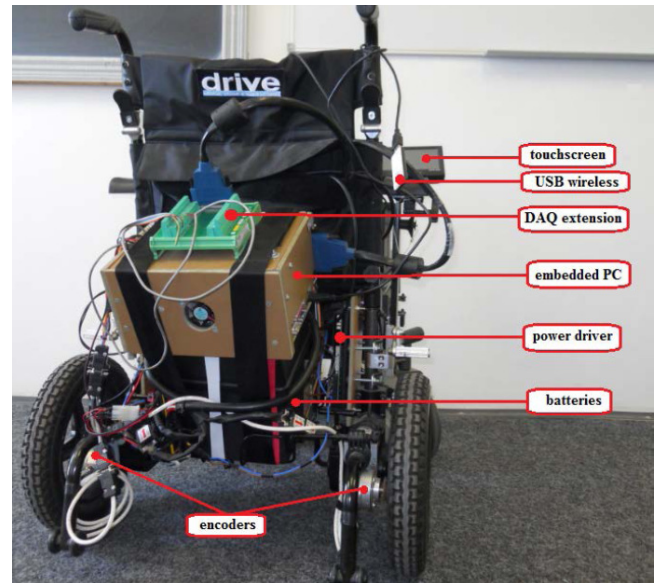


Fig. 1. Hardware system

The system interacts with the user through a graphical interface that provides information such as direction indication, both visual and audio, or displaying battery level.

II. HARDWARE AND SOFTWARE RESOURCES

For the entire system design a number of hardware resources have been used:

- electric wheelchair (Cirrus Power Wheelchair), equipped with industrial encoders with a resolution up to 10000 pulses for speed and position tracking at any time.
- EmbeddedPC, model D945GCLF2D PC Desktop Board with the necessary dimensions (L: 17 145 x W: 17.145 cm) so that it can be attached on the wheelchair without affecting the normal behavior of this kind of system. Also, the computer is equipped with PCI slot.
- PCI data acquisition board, type DAQ 6024E.
- TFT 7" display used for GUI (Graphical User Interface).
- low-cost web camera with CMOS sensors.
- a fastener device, goggles type, provided with infrared LEDs.

The Fig. 1 illustrates the system with the hardware resources presented above.

In Fig. 2 a WMR with two diametrically opposed drive wheels (radius R) and free-wheeling castors (not considered in the kinematics models) is presented. P_r is the origin of the

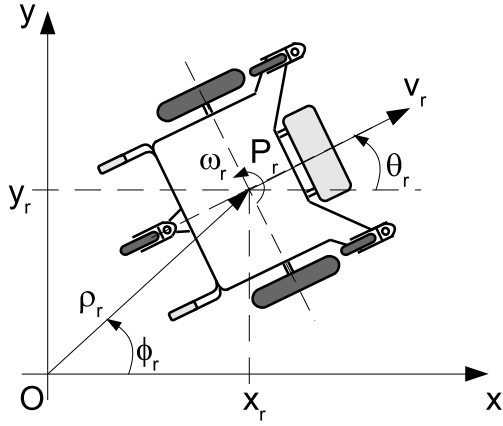


Fig. 2. Robotic wheelchair model and symbols

robot coordinates system. L is the length of the axis between the drive wheels. v_R and v_L are the velocities of the right and left wheels. Let the pose of the mobile robot be defined by the vector $P_r = [x_r, y_r, \theta_r]^T$, where $[x_r, y_r]^T$ denotes the robot position on the plane and θ_r the heading angle with respect to the x -axis. In addition, v_r denotes the linear velocity of the robot and ω_r the angular velocity around the vertical axis. For a unicycle WMR rolling on a horizontal plane without slipping, the kinematic model can be expressed by:

$$\begin{aligned} v_r &= \frac{v_R(t) + v_L(t)}{2} \\ \omega_r &= \frac{v_R(t) - v_L(t)}{L/2} \end{aligned} \quad (1)$$

The kinematics equations of the WMR can be written:

$$\begin{aligned} \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \end{bmatrix} &= \begin{bmatrix} \cos(\theta_r) & 0 \\ \sin(\theta_r) & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} = \\ &= \begin{bmatrix} 1/2 \cdot (v_R + v_L) \cdot \cos(\theta_r) \\ 1/2 \cdot (v_R + v_L) \cdot \sin(\theta_r) \\ (v_R - v_L)/(L/2) \end{bmatrix} \quad (2) \\ \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \end{bmatrix} &= \begin{bmatrix} 1/2 \cdot \cos(\theta_r) & 1/2 \cdot \cos(\theta_r) \\ 1/2 \cdot \sin(\theta_r) & 1/2 \cdot \sin(\theta_r) \\ -L/2 & L/2 \end{bmatrix} \cdot \begin{bmatrix} v_R \\ v_L \end{bmatrix} \quad (3) \end{aligned}$$

For the control of the WMR, two programs are required: MATLAB and LabVIEW. The image processing is done using MATLAB of the area of interest - the eye, by default the iris. LabVIEW provides the graphical user interface which controls the actuators of the wheelchair.

To link the two programming languages, MATLAB and LabVIEW, it is necessary to follow certain steps.

The image processing program is converted into executable using *deploytool* function available in MATLAB. The program will include the routine which inserts data into a file, acting like a buffer. The data represents the string indicating the command to be followed by WMR. The direction pointing is made by the commands "FORWARD", "LEFT", "RIGHT and for shut-down the "STOP" command.

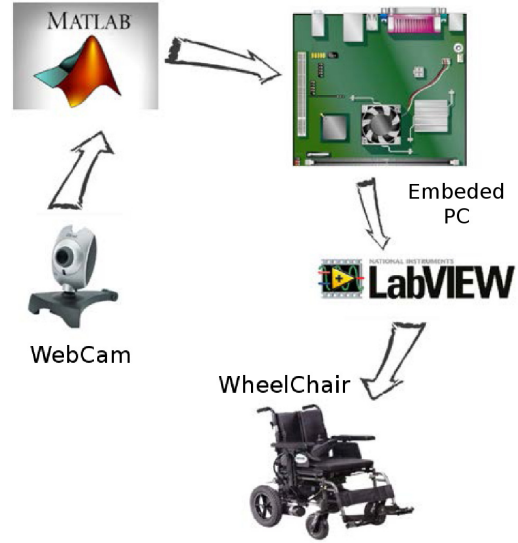


Fig. 3. General software-hardware scheme

LabVIEW, reads these strings and interprets them as commands for the two differential drive wheels of the WMR. The entire hardware-software assembly is presented in the Fig. 3.

The image processing technique uses existing functions available in the MATLAB toolbox called "Image Processing" [6], [7]. This technique, combined with elements of geometry (distance between two points, fitting a circle through 3 non-collinear points) and pixel-level filtering (removing redundant circles), identifying and fitting the iris in a circle. The center of the circle will be the direction pointer depending on the 3 areas on the image: left, right, center. The shut-down command is generated by the undetection of the iris, which by default implies the shutdown of the wheelchair actuators.

In graphic real-time processing, one significant matter is represented by the resolution of the images. A small resolution is not appropriate for the algorithm, the eye having a relatively small area. A high resolution increases the processing time of each frame and the necessity of higher computing resources (CPU, RAM). Thus, a resolution of 320x240 it was chosen as optimal.

The steps, followed for iris tracking and its center are illustrated in the Fig. 4.

In Fig. 5 the histogram equalization is presented corresponding to the images from Step 2 and Step 3. Through this adjustment, the intensities can be better distributed on the histogram. This allows for areas of lower local contrast to gain a higher contrast. Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values.

For direction tracking, using the center of the circle fitted around the iris, an algorithm was used to measure the center position along the image OX -axis ($0 \div 320$ pixels). Therefore, when the center of the circle is under the limit of 100, the direction will be left, when the limit is above 200, the direction is right and the center is given by the length of

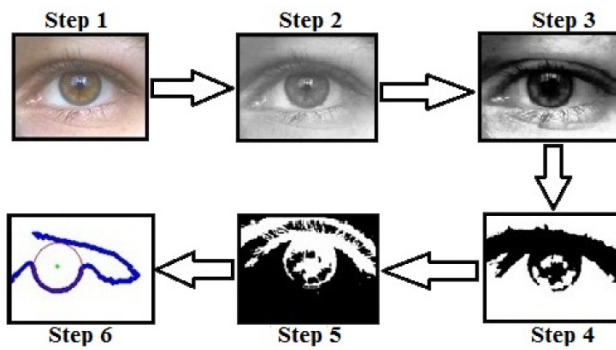


Fig. 4. Iris tracking and its center

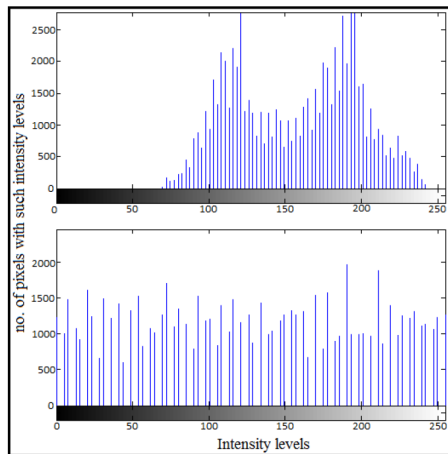


Fig. 5. Histogram equalization

intermediate values (see Fig. 6).

Two safety straps were created along the OY -axis ($0 \div 240$ pixels) - on the top in the range ($180 \div 240$) and on the bottom in the range ($0 \div 40$). These straps are designed to filter out false circles may result from involuntary eye movements. The Fig. 7 describes the implementation of the solution for direction tracking using the eye movements.

III. EXPERIMENTAL RESULTS

Graphical user interface, created with LabVIEW, is illustrated in Fig. 8. The red arrow indicates the inactive direction and the green one the direction chosen by the users eye movement. Figure 9 shows the LabVIEW block diagram corresponding to the graphical user interface.

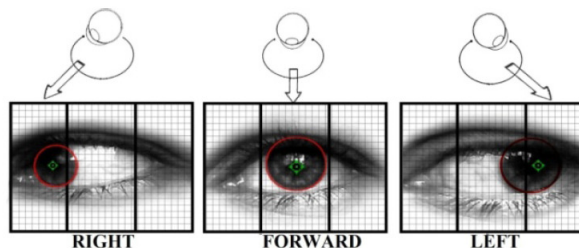


Fig. 6. Direction tracking

Presentation of experimental results are show in Figs. 10 and 11.

IV. CONCLUSIONS AND FUTURE WORKS

The paper shows the real implementation of a solution for wheeled mobile robot control using eye movements and gaze for people with severe motor disability. In this paper is presented a tracking direction algorithm using MATLAB and also a graphical user interface, built with LabVIEW.

The advantage of this implementation is a great versatility in use thanks to the LabVIEW program. The system also allows its portability on any robotic platform and its remote control.

In the future more safety features like laser sensors and distance sensors will have to be implemented and also bring an improvement to the iris tracking algorithm. These types of sensors have the purpose to shut-down the entire system in case of imminent collision, but also to avoid obstacles.

All these future implementations involve the need of more computing resources and use of CCD video cameras, both for iris tracking.

V. ACKNOWLEDGMENTS

This work was supported by CNCIS-UEFISCSU, Romania, project PNII-IDEI 506/2008.

REFERENCES

- [1] L. Fehr, W. E. Langbein, and S. B. Skaar, Adequacy of power wheelchair control interfaces for persons with severe disabilities: A clinical survey, *Journal of Rehabilitation Research and Development*, vol. 37(3), 2000, pp. 353-360.
- [2] R. Solea, U. Nunes and Filipescu A., Trajectory Planning and Sliding mode control for WMR Trajectory-Tracking and Path-Following Respecting Human Comfort Travel, in *7th Portuguese Conference on Automatic Control CONTROLO 2006*, IST-Lisboa, Portugal, 2006.
- [3] R. Solea, Sliding mode control applied in trajectory tracking of WMRs and autonomous vehicles. *PhD thesis*, Coimbra, Portugal, 2009.
- [4] H. Ghaffari Nik, Hum-Power Controller for Powered Wheelchairs, *Graduation thesis*, George Mason University, 2007.
- [5] B. Siciliano, L. Sciavicco, and G. Oriolo, *Robotics: Modelling, Planning and Control (Advanced Textbooks in Control and Signal Processing)*, Springer; 2nd Printing edition, 2011.

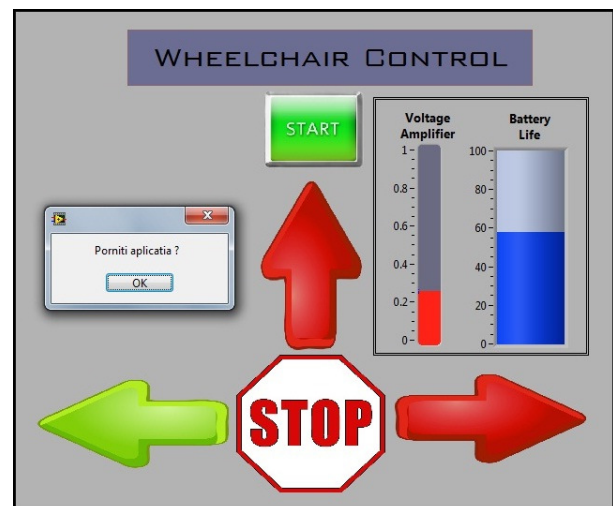


Fig. 8. Graphical user interface

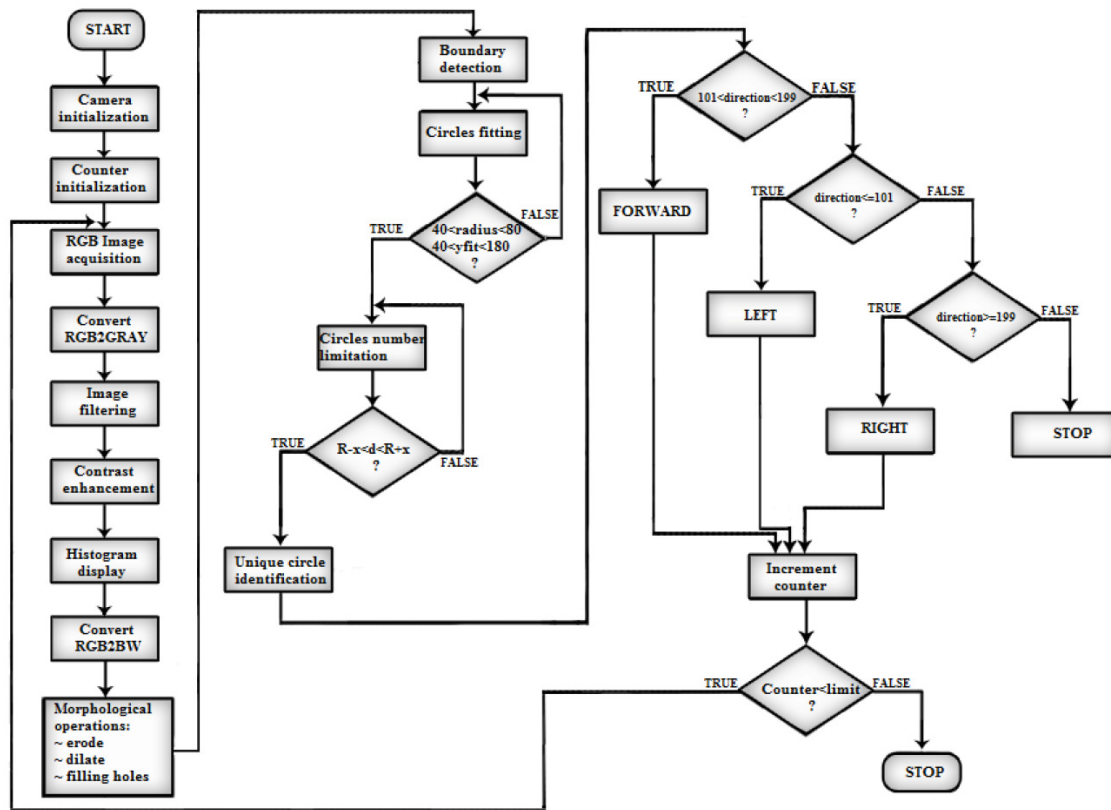


Fig. 7. Block diagram associated to the MATLAB algorithm

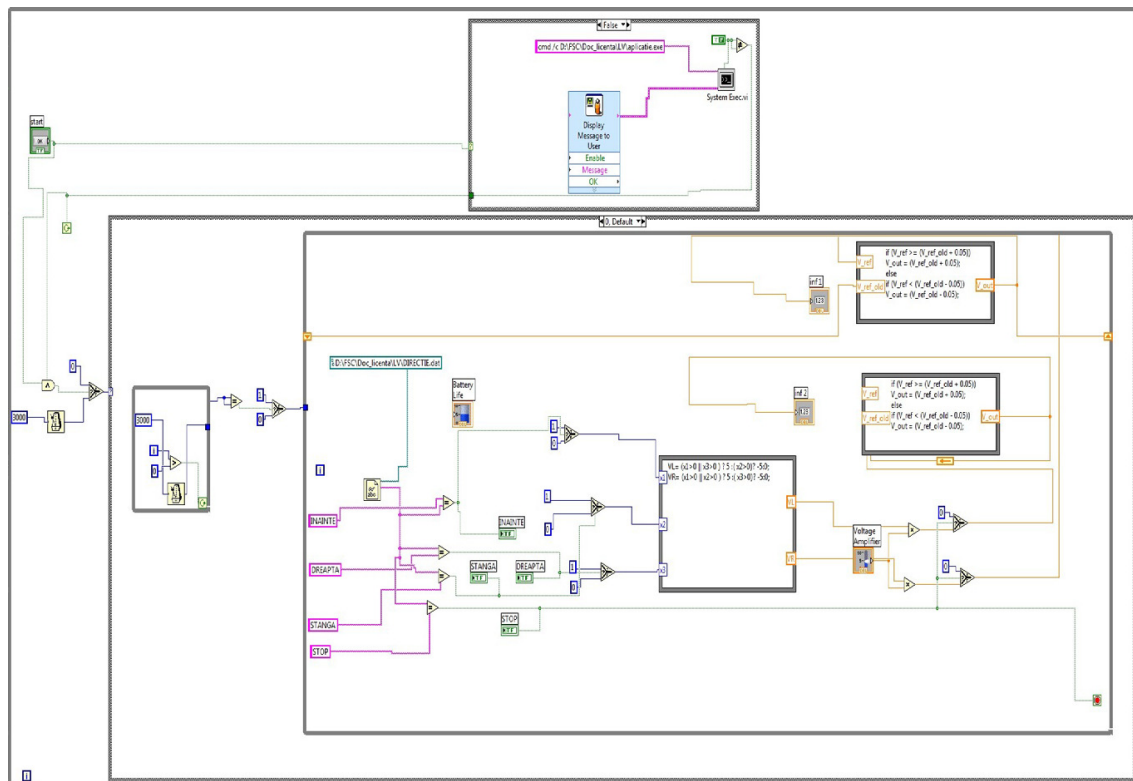


Fig. 9. LabVIEW schematics associated to the graphical user interface

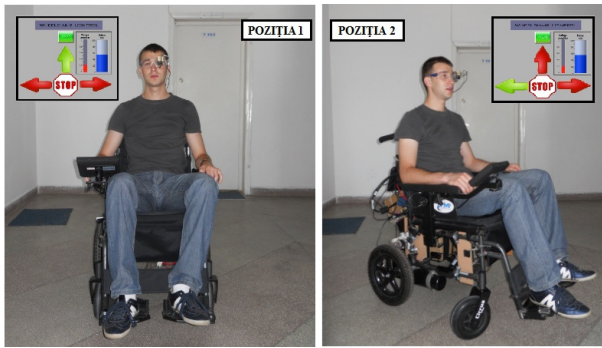


Fig. 10. Pictures from real experiments

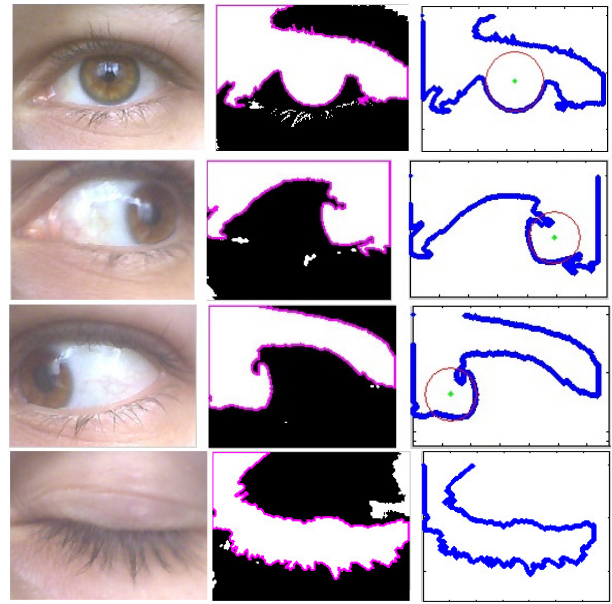


Fig. 11. Results from real experiments

- [6] R. Gonzalez, and R. Woods, *Digital Image Processing Using MATLAB*, 2nd edition, Gatesmark Publishing, 2009.
- [7] The MathWorks, inc., *Image Processing Toolbox Users Guide*, 2007.

Forced periodic oscillations and almost linear behavior

Vladimir Răsvan

Abstract—There are considered systems with sector restricted and globally Lipschitz nonlinear functions which are forced by periodic or almost periodic signals. For these systems there is discussed existence, uniqueness and exponential stability of the limit regimes which are of the same type as the forcing term. The three properties considered together define the almost linear behavior. Since partial results may be obtained using stability multipliers, there are analyzed the properties associated to such multipliers and possible extensions. The paper ends with a sketch of a future research programme.

I. INTRODUCTION AND PROBLEM STATEMENT

Almost linear behavior - an expression coined in [1] and given conceptual significance in [2] - means that some nonlinear system displays several properties that are to be met basically in linear systems: i) an equilibrium at the origin of the state space that is globally asymptotically (possibly exponentially) stable and ii) if forced by an (almost) periodic signal, the system will have a unique limit regime, defined on the entire real ("time") axis, globally asymptotically (possibly exponentially) stable, that is (almost) periodic (in fact is of the same type i.e. has the same "waveform" as the forcing signal).

A. In order to better explain the concepts and to perform the aimed extensions we shall deal mainly with systems with lumped parameters i.e. described by ordinary differential equations; however other classes of systems will be also discussed.

It appeared in [1] as well as in [2] that a most suitable class of systems to display almost linear behavior is the class of the feedback systems with sector restricted nonlinearities 1

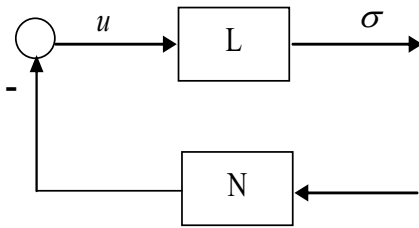


Fig. 1. Absolute stability feedback structure.

The linear block L of fig.1 accounts for a system with lumped constant parameters described by the state representation

$$\dot{x} = Ax + b\mu(t), \quad \sigma = c^*x \quad (1)$$

The author is with Department of Automatic Control, University of Craiova, A.I. Cuza, 13, Craiova, RO-200585, Romania. vrasvan@automation.ucv.ro

or by the strictly proper rational transfer function $\chi(\sigma)$; if (A, b) is controllable and (c^*, A) is observable, then $\chi(\lambda) = c^*(\lambda I - A)^{-1}b$ is irreducible; conversely, if $\chi(\lambda)$ is irreducible, strictly proper and rational, then a state representation (1) can be associated, with the above properties. If the linear block has time delays then we may have e.g. the following description

$$\begin{aligned} \dot{x} &= A_0x(t) + A_1x(t - \tau) + b\mu(t) \\ \sigma(t) &= c_0^*x(t) + c_1^*x(t - \tau) \end{aligned} \quad (2)$$

or the transfer function $\chi(\lambda)$, a strictly proper meromorphic function - the ratio of two quasi-polynomials. In general, a linear block with distributed parameters will be described by an integral equation of the form

$$\sigma(t) = \rho(t) + \int_0^t \kappa(t - \theta)\mu(\theta)d\theta \quad (3)$$

If a certain controllability property is assumed [3] then (3) may be replaced by

$$\sigma(t) = \int_{t_0}^t \kappa(t - \theta)\mu(\theta)d\theta \quad (4)$$

The transfer function of such a linear block is $\tilde{\kappa}(\lambda)$ - the Laplace transform of the kernel function $\kappa(t)$ assumed to be causal i.e. $\kappa(t) \equiv 0, t < 0$.

The nonlinear block is static and assumed to be described by a function whose only property is that it belongs to the class of fig.2

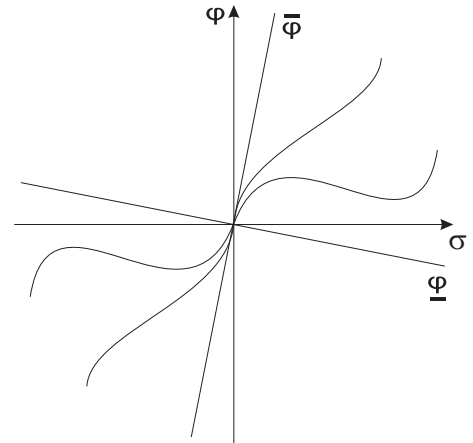


Fig. 2. Sector restricted nonlinearity.

Mathematically, these functions are characterized by the sector restrictions

$$\underline{\varphi} \leq \frac{\varphi(\sigma)}{\sigma} \leq \overline{\varphi} \quad (5)$$

sometimes combined with the slope restrictions

$$\underline{\sigma} \leq \varphi'(\sigma) \leq \bar{\sigma} \quad (6)$$

where, without any loss of generality we may assume $[\underline{\varphi}, \bar{\varphi}] \subseteq [\underline{\sigma}, \bar{\sigma}]$.

The feedback connection of fig.1 shows that the closed loop system is described by

$$\dot{x} = Ax - b\varphi(c^*x) \quad (7)$$

if the linear block is (1), by

$$\begin{aligned} \dot{x} &= A_0x(t) + A_1x(t - \tau) - b\varphi(\sigma(t)) \\ \sigma(t) &= c_0^*x(t) + c_1^*x(t - \tau) \end{aligned} \quad (8)$$

if the linear block is (2), or by

$$\sigma(t) = \rho(t) - \int_0^t \kappa(t - \theta)\varphi(\sigma(\theta))d\theta \quad (9)$$

if the linear block is (3).

The almost linear behavior is ensured in these cases by two basic properties: a) absolute stability of (7)-(9) i.e. global asymptotic stability of the zero equilibrium for all nonlinear functions belonging to the class defined above and b) existence, uniqueness and global stability of the forced oscillations that have to be of the same kind as the forcing term e.g. periodic with the same period or almost periodic.

The present paper aims to extend already existing results on *dissipativeness* - one of the tools in getting the almost linear behavior - by considering slope restricted - see (6) - and monotone nonlinearities - see (6) with $\underline{\sigma} = 0$. With respect to this it will be considered the so called *Yakubovich stability multiplier*, the oldest while not the best known non-causal (anticipative) stability multiplier. The dissipativeness (in the sense of N. Levinson i.e. ultimate boundedness) allows to obtain existence of a periodic oscillation *via* a fixed point theorem [4], [5], [6]. The extensions are made for systems with lumped parameters i.e. described by ordinary differential equations also for systems described by time delay equations.

B. We shall give now a brief account of some known results in the field of the almost linear behavior. Concerning the absolute stability, we give below an account of the frequency domain inequalities that are valid for all classes of systems (with lumped or distributed parameters).

a) If φ is also time dependent but subject to (5) the frequency domain inequality is of the circle type

$$1 + (\underline{\varphi} + \bar{\varphi})\Re\chi(i\omega) + \underline{\varphi}\bar{\varphi}|\chi(i\omega)|^2 \geq 0 \quad (10)$$

with the well known case of $\underline{\varphi} = 0$ corresponding to the Popov criterion without parameter

$$\frac{1}{\bar{\varphi}} + \Re\chi(i\omega) \geq 0 \quad (11)$$

b) For time invariant φ subject to (5) the Popov criterion with free parameter is valid

$$1 + (\underline{\varphi} + \bar{\varphi})\Re\chi(i\omega) + \underline{\varphi}\bar{\varphi}|\chi(i\omega)|^2 + \Re w\theta\chi(i\omega) \geq 0 \quad (12)$$

c) For time invariant φ subject to (5) and (6) there exist various frequency domain inequalities of the form (for $\underline{\varphi} = \underline{\sigma} = 0$)

$$\frac{1}{\bar{\varphi}} + \Re\zeta(i\omega)\chi(i\omega) \geq 0 \quad (13)$$

where $\zeta(\lambda)$ are causal or even non-causal stability multipliers. One of these multipliers is the Yakubovich multiplier [7], [8]

$$\zeta(\lambda) = 1 + \theta\lambda - \beta\lambda^2 \quad (14)$$

These frequency domain inequalities must be viewed together with the assumptions on the nonlinear function but also with the assumptions on the eigenvalues of the linear subsystems: stable, critical or even unstable.

C. Consider now the case of the forced oscillations. The strongest result concerns systems (7) or (8) forced by an oscillatory term

$$\dot{x} = Ax - b\varphi(c^*x) + f(t) \quad (15)$$

or

$$\begin{aligned} \dot{x} &= A_0x(t) + A_1x(t - \tau) - b\varphi(\sigma(t)) + f(t) \\ \sigma(t) &= c_0^*x(t) + c_1^*x(t - \tau) \end{aligned} \quad (16)$$

with $f: \mathbb{R} \mapsto \mathbb{R}^n$ being T -periodic or almost periodic and $\varphi: \mathbb{R} \mapsto \mathbb{R}$ subject to (5) with $\underline{\varphi} = 0$ and globally Lipschitz or subject to (6) with $\underline{\sigma} = 0$. If (11) holds being strict, then there exists a unique solution which is bounded on $\mathbb{R}$ and is exponentially stable; moreover, if f is T -periodic, then the global oscillation is periodic and if f is almost periodic then the global oscillation is such.

This result was established in [9] for (15) and in [10] for (16), using different approaches. The research that followed aimed for relaxation of (11), considered to be restrictive.

This paper is written following also this line of research and, as a consequence, what remains is organized as follows. There are presented the two stages of solving the forced oscillations problem: existence, based on a suitable fixed point result, and stability, based on the convergence property (asymptotics of the difference of some two solutions). It is pointed out that these properties are consequences of some inequalities satisfied by system's solutions, as consequences of the frequency domain inequalities.

As already stated, we use a frequency domain inequality with the Yakubovich multiplier (14) for the dissipativeness of (15). From dissipativeness a fixed point theorem [4], [5], [6] will give existence of the periodic solution. The paper ends with some conclusions and hints concerning almost periodic oscillations and their more "structured" version - the quasiperiodic oscillations.

II. EXISTENCE OF PERIODIC SOLUTIONS

A. Results

The object of our development will be here the system with lumped parameters (15) under the basic assumptions of [1], [9]: i) A is a Hurwitz matrix; ii) $\varphi \in \mathcal{C}^1$ and subject to the following sector and slope restrictions

$$0 \leq \sigma\varphi(\sigma) \leq \bar{\varphi}\sigma^2 ; 0 \leq \varphi'(\sigma) \leq \bar{\sigma} \quad (17)$$

for finite $\bar{\varphi}$ and $\bar{\sigma}$. The following theorem is true

Theorem 1: Consider the system (15) under the above assumptions in a relaxed form i.e. (17) hold only for “large deviations” of the form $|\sigma| \geq \text{sigma}_0$. If there exist the freely chosen parameters $\theta \in \mathbb{R}$, $\beta \in \mathbb{R}^+$ such that the following frequency domain inequality holds

$$\left(\frac{1}{\bar{\varphi}} + \frac{\beta}{\bar{\sigma}}\omega^2\right) + \Re e(1 + i\omega\theta + \beta\omega^2)\chi(i\omega) > 0, \quad \omega \in \overline{\mathbb{R}} \quad (18)$$

and the nonlinear functions of the class defined by (17) are also such that

$$\liminf_{|\sigma| \rightarrow \infty} \frac{\theta}{\sigma^2} \left[\int_0^\sigma \varphi(\eta) d\eta - \frac{1}{2} \sigma \varphi(\sigma) \right] \geq 0 \quad (19)$$

then (15) is uniformly dissipative in the sense of N. Levinson (uniformly ultimately bounded) and therefore, if f is T -periodic, (1.12) has a T -periodic solution.

Theorem 1 is the single nonlinearity version of the main result in [1]; under a slightly different form it is to be found in [9] as a dissipativeness result (in fact in both references). For this finite dimensional case existence of a periodic solution follows from the fixed point theorem of Brouwer as shown in e.g. [5].

We turn now to the same problem for time delay systems. Existence of forced oscillations (periodic solutions) may be deduced e.g. from the fixed point theorem of F. E. Browder [5], [6]. Since system (16) is also of the type occurring in absolute stability, one could be tempted to make use of the abstract positiveness (Yakubovich Kalman Popov) theory applied to time delay systems. This would require the choice of a suitable Hilbert space combined with adequate compactness criteria. For this reason the approach chosen here is to consider the analysis based on integral equations and make use of the properties of the standard Banach space $\mathcal{C}$ of the continuous functions with the topology of the uniform convergence. We thus state

Theorem 2: Consider the system (16) under the following assumptions: i) the linear autonomous system

$$\dot{x}(t) = A_0 x(t) + A_1 x(t - \tau) \quad (20)$$

is exponentially stable; ii) $\varphi : \mathbb{R} \mapsto \mathbb{R}$ is subject to the following sector restrictions

$$0 \leq \sigma \varphi(\sigma) \leq \bar{\varphi} \sigma^2 \quad (21)$$

for finite $\bar{\varphi}$ and “large deviations” $|\sigma| \geq \sigma_0$. If there exists some $\theta \in \mathbb{R}$ such that the following frequency domain inequality holds

$$\frac{1}{\bar{\varphi}} + \Re e(1 + i\omega\theta)\chi(i\omega) > 0, \quad \omega \in \overline{\mathbb{R}} \quad (22)$$

with

$$\chi(\lambda) = (c_0^* + e^{-\lambda\tau} c_1^*)(\lambda I - A_0 - A_1 e^{-\lambda\tau})^{-1} b$$

and the nonlinear functions of the class defined by (21) are such that (19) holds then (16) is uniformly bounded and uniformly ultimately bounded. If f is T -periodic then (16) has a T -periodic solution.

Under this form Theorem 2 is new and the avoidance of the Liapunov method is also new. The proof (lengthy enough not to be given here in complete form) is based on the following steps: a) to obtain an estimate for the output $\sigma(t)$ - the argument of $\varphi(\cdot)$ along system's trajectories; b) to use the above estimate in order to obtain the state estimate for $x(\cdot)$

$$|x(t + \lambda)| \leq \gamma_0 + \gamma_1 e^{-\alpha(t-\eta+\lambda)} \|x_\eta\| \quad (23)$$

where $\|x_\eta\|$ is the standard norm on $\mathcal{C}(-\tau, 0; \mathbb{R}^n)$ namely

$$\|x_\eta\| = \sup_{-\tau \leq \lambda \leq 0} |x(\eta + \lambda)|$$

From (23) it follows that

$$\|x_t\| \leq \gamma_0 + \gamma_1 e^{-\alpha(t-\eta)} \|x_\eta\| \quad (24)$$

which accounts for boundedness and also for ultimate boundedness: for $t > \vartheta$ any solution will enter the ball $\|\phi\| \leq 2\gamma_0$ of the Banach space $\mathcal{C}(-\tau, 0; \mathbb{R}^n)$.

Our goal is now to apply the already mentioned fixed point theorem of F. E. Browder that we reproduce after [6], [11]

Theorem 3: (F. E. Browder [11]) Let S and S_1 be open convex subsets of the Banach space X , S_0 - a closed convex subset of X , $S_0 \subset S_1 \subset S$ and $f : S \mapsto X$ a compact mapping. Assume that for some integer $m > 0$ f^m (the m -iterate of f) is well defined on S_1 , $\bigcup_{0 \leq j \leq m} f^j(S_0) \subset S_1$ while $f^m(S_0) \subset S_0$. Then f has a fixed point in S_0 .

We shall follow now the approach of [11], Theorem 33.1 and sketch it for the sake of completeness. Let $x(t; \eta, \phi)$ be some solution of (16) then, with the notations of J. K. Hale, let $x_t(\eta, \phi)$ be its state defined on $\mathcal{C}(-\tau, 0; \mathbb{R}^n)$ and consider $f(\phi) = x_T(0, \phi)$ - the mapping that defines the translation operator in the sense of [5]. Consider the open ball $S = \{\phi \in \mathcal{C} : \|\phi\| < \gamma\}$ for $\gamma > 0$ to be specified next; this ball is a convex subset of $\mathcal{C}(-\tau, 0; \mathbb{R}^n)$. We deduce from (24) that

$$\|x_T(0, \phi)\| \leq \gamma_0 + \gamma_1 e^{-\alpha T} \gamma = \gamma_* \quad (25)$$

Turning to (16) we find by direct computation that $|\dot{x}(t; 0, \phi)| \leq L(\gamma_*)$ for $t \in [0, T]$. This shows that S is mapped into a compact subset of $\mathcal{C}$ (in fact $x(t; 0, \phi)$ will be uniformly continuous and compactness will follow from the Arzelà-Ascoli theorem). Consequently the translation operator is compact. Let now $S_0 = \{\phi \in \mathcal{C} : \|\phi\| < 2\gamma_0\}$ where γ_0 is that of (24). This ball is a closed convex subset of $\mathcal{C}$; we shall have $S_0 \subset S$ for $\gamma > 2\gamma_0$ and, as (24) shows, all trajectories of (16) will eventually penetrate S_0 . In particular this is true for the trajectories starting from $S_1 = \{\phi \in \mathcal{C} : \|\phi\| < \beta\}$ with $\beta \in (2\gamma_0, \gamma)$ - also an open convex set (open ball) of $\mathcal{C}$; we deduce from (24) existence of an integer $m > 0$ such that for $\phi \in S_1$ we have $\|x_{mT}(0, \phi)\| \leq 2\gamma_0$ or

$$f^m(\phi) = x_{mT}(0, \phi) \in S_0 \quad (26)$$

Also if $\gamma_0 + \gamma_1 \gamma_0 = (1 + \gamma_1)\gamma_0 > 2\gamma_0$ (always achievable by possible increase of γ_1) we shall have for $\|\phi\| < 2\gamma_0 < \gamma$

$$\|x_{jT}(0, \phi)\| \leq \gamma_0 + \gamma_1 e^{-\alpha j T} \gamma_0 \leq \gamma_0 + \gamma_1 \gamma_0 < \beta$$

what gives $f^j(S_0) \subset S_1$ for any j hence $\bigcup_{0 \leq j \leq m} f^j(S_0) \subset S_1$. The assumptions of Theorem 3 being fulfilled, we deduce that the mapping defined by the translation operator has a fixed point hence a T -periodic solution exists.

B. An illustrating application

We shall consider an example from machine tool vibrations, the so called system of Johnson and Moon [12] which may display stable equilibria, self sustained periodic or almost periodic oscillations, also chaotic behavior. If the nonlinear function satisfies (5) with $\varphi = 0$ for large deviations, we shall show that this system is dissipative. The equation of the vibrating system is

$$\ddot{y}(t) + (a - c)\dot{y}(t) + c\dot{y}(t - 1) + by(t) + \varphi(y(t)) = \psi(t, y(t), \dot{y}(t)) \quad (27)$$

with $a > 0$, $b > 0$, $c > 0$. Here ψ is bounded; if $\psi(t, y(t), \dot{y}(t)) \equiv \xi(t)$ which is T -periodic then (27) may have a periodic solution.

We shall apply here Theorem 2: we check first the exponential stability of the linear subsystem whose characteristic equation is

$$\lambda^2 + (a - c)\lambda + b + c\lambda e^{-\lambda} = 0 \quad (28)$$

Equation (28) has its roots in $\mathbb{C}^-$ iff the equation

$$(4z^2 + 2az + b) \cosh z + (4z^2 + 2(a - 2c)z + b) \sinh z = 0 \quad (29)$$

has this property. We shall use here the fundamental results of [13], Chapter VII. Since $b > 0$ we check the necessary conditions of Stodola type

$$2a + b > 0, \quad 4 + \frac{b}{2} + 2(a - 2c) > 0 \quad (30)$$

which are fulfilled provided the control parameter $c > 0$ is small enough. Due to a certain inequality which is fulfilled for this system, it follows that the only additional condition required would be $c < 2a$ which ensures also (31).

Next, the transfer function of the linear part is

$$\chi(\lambda) = \frac{1}{\lambda^2 + (a - c + ce^{-\lambda})\lambda + b} \quad (31)$$

Since $b > 0$ is not limited by the linear stability conditions, having also in mind the Aizerman conjecture and problem we write down (22) for $\bar{\varphi} \rightarrow \infty$

$$\begin{aligned} \Re(1 + i\omega\theta)\chi(i\omega) &= \\ &= \frac{b - \omega^2 + c\omega \sin \omega + \omega^2\theta(a - c + c \cos \omega)}{(b - \omega^2 + c\omega \sin \omega)^2 + (a - c + c \cos \omega)^2\omega^2} > 0 \end{aligned} \quad (32)$$

For the fulfilment of the above inequality it suffices to have

$$\omega^2 \left[(a\theta - 1) - c\theta(1 - \cos \omega) + c \frac{\sin \omega}{\omega} \right] + b > 0 \quad (33)$$

We shall use the choice of $\theta > 0$ to ensure

$$(a\theta - 1) - c\theta(1 - \cos \omega) + c \frac{\sin \omega}{\omega} > 0$$

A straightforward analysis will give the choice

$$\theta > \max \left\{ \frac{1}{a - 2c}, \frac{1 + c|\cos \bar{\omega}|}{a - c(1 + c|\cos \bar{\omega}|)} \right\} \quad (34)$$

where $\bar{\omega}$ is the root of $\tan \omega = \omega$ located in the interval $(\pi/2, \pi)$. With this choice (32) holds for all real ω hence (27) is dissipative for arbitrarily large sectors.

III. SOME CONSIDERATIONS ON UNIQUENESS AND STABILITY OF THE PERIODIC REGIMES

We shall start here from a remark in [9] concerning Theorem 1 in the special case $\beta = 0$ and without the slope restriction on $\varphi(\nu)$: the periodic solution whose existence is proved is not a *limit regime* i.e. there exist solutions of (15) such that $\lim_{t \rightarrow \infty} |x(t) - \tilde{x}(t)| \neq 0$ where $\tilde{x}(t)$ is the periodic solution and $x(t)$ - a “perturbed solution” in the sense of Liapunov.

The stronger result is that of existence, uniqueness and exponential stability of the periodic/almost periodic regime. From the point of view of the frequency domain inequality, this result was obtained for $\theta = \beta = 0$ in (18) - the frequency domain inequality without parameters i.e. the correspondent of the circle criterion. In the case without delays i.e. of (15) the result was proved in [9]; the proof instrument was a quadratic Liapunov function whose existence was ensured by the frequency domain inequality. In the case of (16) the proof instrument was the standard Popov theory which allowed obtaining some inequalities [10]; these inequalities are identical to those obtained for the solutions of (15) via the Liapunov method; in fact they might have been obtained by the same approach in the case without delay also. The “fixed point” theorem used here is stronger than the theorem of Browder: it is the Kurzweil-Halanay theorem on invariant manifolds that we reproduce here again for the sake of completeness [14], [15]

Theorem 4: Suppose for a general flow in some Banach space X there exist some real positive constants $l, \theta, \alpha < 1, \kappa_1, \kappa_2$ such that

- a) $\|\tilde{c}\| \leq l \Rightarrow \|c(t; \tilde{t}, \tilde{c})\| \leq l, \quad \tilde{t} + \theta \leq \tilde{t} + 2\theta;$
- b) $\|\tilde{c}_1\| \leq l, \quad \|\tilde{c}_2\| \leq l \Rightarrow \|c(t; \tilde{t}, \tilde{c}_1) - c(t; \tilde{t}, \tilde{c}_2)\| \leq \alpha \|\tilde{c}_1 - \tilde{c}_2\|, \quad \tilde{t} + \theta \leq \tilde{t} + 2\theta;$
- c) $\|c(t; \tilde{t}, \tilde{c}_1) - c(t; \tilde{t}, \tilde{c}_2)\| \leq e^{\kappa_2(t - \tilde{t})} \|\tilde{c}_1 - \tilde{c}_2\|, \quad t \geq \tilde{t}$

Then there exists $p : \mathbb{R} \mapsto X$ such that i) $\|p(t)\| \leq l$; ii) $p(t) = c(t; \tilde{t}, p(\tilde{t}))$, $t \geq \tilde{t}, \tilde{t} \in \mathbb{R}$; iii) $\|c(t; \tilde{t}, \tilde{c}) - p(t)\| \leq \kappa_0 e^{-\nu(t - \tilde{t})} \|\tilde{c} - p(\tilde{t})\|$ for $\|\tilde{c}\| \leq l, t \geq \tilde{t}$ and some positive κ_0, ν ; iv) if the flow is T -periodic then $p(t)$ is also T -periodic; v) if the flow is almost periodic then p is also almost periodic.

Since, as we mentioned previously the results are similar for (15) and (16) even if the approaches are different, we reproduce here only the main theorem of [10]

Theorem 5: Consider the system (16) under the following assumptions i) the linear autonomous (20) is exponentially

stable; ii) $\varphi : \mathbb{R} \mapsto \mathbb{R}$ is globally Lipschitz satisfying

$$0 \leq \frac{\varphi(\sigma_1) - \varphi(\sigma_2)}{\sigma_1 - \sigma_2} \leq L, \quad \forall \sigma_1 \neq \sigma_2, \quad \varphi(0) = 0 \quad (35)$$

iii) the following frequency domain inequality holds

$$\frac{1}{L} + \Re \chi(i\omega) > 0, \quad \omega \in \overline{\mathbb{R}} \quad (36)$$

with $\chi(\sigma)$ as in Theorem 2; iv) $|f(t)| \leq M, \forall t \in \mathbb{R}$. Then there exists a bounded on the whole real axis solution of the system which is periodic, almost periodic respectively if f is periodic almost periodic respectively. Moreover this global solution is exponentially stable.

This type of result is but well known: in the years that followed it has been extended to systems of neutral functional differential equations [16], systems with periodic coefficients [17]. This seems to be the basic available result for almost linear behavior (in the Section I). Combining i) and ii) in Theorem 5 one may state that, in some broad sense, *strict real positiveness implies almost linear behavior*.

Recently some new results have been announced and/or published in connection with the almost linearity topics [18]. The starting points of these results are two: the first one goes back to [9], Theorem 2, that concerns stability of some global bounded solution of (15); the second one is the remark that (35) allows the following frequency domain stability inequality due to Zames and Falb

$$\Re \zeta(i\omega) \left(\frac{1}{L} + \Re \chi(i\omega) \right) > 0, \quad \omega \in \overline{\mathbb{R}} \quad (37)$$

where the stability multiplier is defined by

$$\zeta(i\omega) = 1 - \sum_{n=0}^{\infty} \theta_n e^{in\omega T} \quad (38)$$

with $\theta_n \geq 0$ and $\sum_{n=0}^{\infty} \theta_n < 1$. Using this multiplier it is proved that a T -periodic solution of (15) (assumed to exist) is stable in the sense of Liapunov.

On the other hand there exists at least one case when the periodic solution is not unique. Since its statement and proof are in a little circulated textbook [19] we reproduce it for the sake of completeness

Theorem 6: Consider the system

$$\dot{x} = Ax - b\varphi(c^*x, t) + f(t) \quad (39)$$

with $\varphi(\cdot, t) \in \mathfrak{E}^2$ and $\varphi(\nu, \cdot), f$ being T -periodic. Let (A, b, c) be a non-degenerate triple i.e. $\chi(\lambda) = c^*(\lambda I - A)^{-1}b$ is irreducible. Assume the following properties to be true: i) the function $\varphi : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ is such that

$$0 \leq \frac{\partial \varphi}{\partial \sigma}(\sigma, t) \leq \overline{\varphi} \leq +\infty; \quad 0 < \frac{\partial \varphi}{\partial \sigma}(0, t) \equiv \lambda_0 < \overline{\varphi} \quad (40)$$

ii) the matrix $A - \lambda_0 bc^*$ has one eigenvalue in $\mathbb{C}^+$ and all other in $\mathbb{C}^-$. If the system has a T -periodic solution and there exists some $\lambda_1 > 0$ such that $A + \lambda_1 I$ has one eigenvalue in $\mathbb{C}^+$ and all other in $\mathbb{C}^-$, then the system has at least three T -periodic solutions.

Obviously this does not account for an almost linear behavior. Since hyperbolicity of the matrices $A - \lambda_0 bc^*$ and $A + \lambda_1 I$ is essential, we may infer that almost linear behavior is not specific to hyperbolic systems.

IV. CONCLUSIONS, PERSPECTIVE, OPEN PROBLEMS

The first important conclusion of the overall view of the previous sections is that if the following conditions are met: i) exponentially stable linear part; ii) sector restricted time invariant and globally Lipschitz nonlinearities; iii) positive real transfer function of the linear part, then the almost linear behavior is ensured. There exist various relaxations of these three conditions but none of them has been yet proved to entirely preserve the property of almost linear behavior. However one may identify several research directions that represent significant progress.

The most promising appears to be the application of the Zames-Falb multiplier (38) which is causal (unlike the Popov multiplier which is non-causal i.e. anticipative) to obtain dissipativeness and, further, almost linear behavior. Our point of view is that achieving this goal heavily relies on the proof technique.

Another possible development would be extension of the above approach to systems described by functional differential equations. Since the standard Popov approach uses the integral equations, a by-product could be exactly almost linear behavior for systems described by integral equations.

These extensions, if based on dissipativeness in the sense of Levinson, gives existence and, possibly, uniqueness for periodic solutions. It is not without interest to obtain relaxed criteria for the existence of almost periodic solutions: especially the techniques of [20] may turn applicable.

To end this discussion, we mention that there exists an appealing experience in using noncausal stability multipliers, other than the Popov PD multiplier [21], [22]. A research aiming to apply them to forced oscillations might also be rewarding.

REFERENCES

- [1] I. Barbălat and A. Halanay, "Conditions de comportement "presque linéaire" dans la théorie des oscillations," *Rev. Roum. Sci. Techn.-Electrotechn. et Energ.*, vol. 19, no. 2, pp. 321-341, June 1974.
- [2] V. Răsvan, "Almost linear behaviour in systems with sector restricted nonlinearities," *Proc. Romanian Acad., Series A: Math., Phys., Techn. Sci., Inf. Sci.*, vol. 2, no. 3, pp. 127-135, September 2001.
- [3] V. M. Popov, *Hyperstability of Control Systems*. Bucharest & Berlin-Heidelberg-New-York: Editura Academiei & Springer-Verlag, 1973.
- [4] V. A. Pliss, *Nonlocal problems of oscillation theory (Russian)*. Moscow & Leningrad, USSR: Nauka, Leningrad branch, 1964.
- [5] M. A. Krasnosel'skii, *The translation operator on the trajectories of the differential equations (Russian)*. Moscow, USSR: Nauka, 1966.
- [6] T. Yoshizawa, *Stability Theory and the Existence of Periodic Solutions and Almost Periodic Solutions*, ser. Applied Mathematical Sciences. Berlin-Heidelberg-New York: Springer Verlag, 1975, no. 14.
- [7] V. A. Yakubovich, "Frequency domain conditions of absolute stability and dissipativeness of control systems with a single differentiable element (russian)," *Doklady Akad. Nauk SSSR*, vol. 160, no. 2, pp. 298-301, 1965.
- [8] —, "The method of the matrix inequalities in the the theory of stability for nonlinear controlled systems ii. absolute stability in the class of slope restricted nonlinearities (russian)," *Avtom. i telemekh.*, vol. 29, no. 4, pp. 588-590, April 1965.

- [9] —, “The method of the matrix inequalities in the theory of stability for nonlinear controlled systems i. absolute stability of forced oscillations (russian),” *Avtom. i telemekh.*, vol. 28, no. 7, pp. 1017–1029, April 1964.
- [10] A. Halanay, “Almost periodic solutions for a class of nonlinear systems with time lags,” *Rev. Roum. Math. Pures Appl.*, vol. 14, no. 9, pp. 1269–1276, 1969.
- [11] T. Yoshizawa, *Stability Theory by Liapunov's Second Method*. Tokyo Japan: The Mathem. Soc. of Japan, 1966.
- [12] T. Erneux, *Applied Delay Differential equations*, ser. Surveys and Tutorials in the Applied Mathematical Sciences. Berlin-Heidelberg-New York: Springer Verlag, 2009, vol. 3.
- [13] N. G. Chebotarev and N. N. Meiman, *The Routh-Hurwitz problem for polynomials and entire functions (Russian)*, ser. Trudy Mat.Inst. “V.A.Steklov”. Moscow: USSR Acad. Publ. House, 1949, vol. 26.
- [14] J. Kurzweil, “Invariant manifolds for flows,” in *Differential equations and dynamical systems pp. 431–468 (J. K. La Salle & J. P. La Salle eds.)*. New York: Academic Press, 1967.
- [15] A. Halanay, “Invariant manifolds for systems with time lag,” in *Differential equations and dynamical systems pp. 199–213 (J. K. La Salle & J. P. La Salle eds.)*. New York: Academic Press, 1967.
- [16] A. Halanay and V. Răşvan, “Periodic and almost periodic solutions for a class of systems described by coupled delay-differential and difference equations,” *Nonlinear Analysis. Theory, Methods & Applications*, vol. 1, pp. 197–206, 1977.
- [17] —, “Oscillations in systems with periodic coefficients and sector restricted nonlinearities,” *Operator Theory. Advances and Applications*, vol. 117, pp. 141–154, 2000.
- [18] D. A. Altshuller, Frequency domain criterion for stability of oscillations in a class of nonlinear feedback systems. Proc. 2007 Conf. on Periodic Control Systems, St. Petersburg Russia, pp. 1396–1400. [Online]. Available: <http://lib.physcon.ru/?item=1396>
- [19] G. A. Leonov, V. M. Zavarykin, and I. V. Ioffe, *Frequency methods in oscillations theory (a textbook)*. Kursk USSR: Kursk Pedagogical Institute Publ. House, 1974.
- [20] M. A. Krasnosel'skii, V. S. Burd, and Y. S. Kolesov, *Nonlinear almost periodic oscillations (Russian)*, ser. Nonlinear analysis and its applications. Moscow USSR: Nauka, 1970.
- [21] S. Gheţaru, “Hyperstability of discrete time control systems,” Ph.D. dissertation, Institute for Energy Research of Romanian Academy, 1969.
- [22] Y. V. Venkatesh, “Noncausal multipliers for nonlinear system stability,” *IEEE Trans. on Autom. Control*, vol. AC-15, no. 2, pp. 195–204, April 1970.

Control algorithms for multi-legged robots in fault conditions using fuzzy logic

C. F. Resceanu

Abstract—Fault tolerance is an important design criterion for robotic systems operating in hazardous or remote environments. This paper addresses the issue of tolerating a locked joint failure in gait planning for hexapod walking machines which have symmetric structures and legs in the form of an articulated arm with two revolute joints. A locked joint failure is one for which a joint cannot move and is locked in place. If a failed joint is locked, the workspace of the resulting leg is constrained, but hexapod walking machines have the ability to continue static walking. A fault tolerant fuzzy control algorithms using dynamics of leg is proposed. Also, problem of kinematics constraints of the failed leg in fault-tolerant gaits for a locked joint failure is presented.

I. INTRODUCTION

RESEARCH on legged locomotion has a long history. Biologists and other scientists have long studied the structure and motion of animals. In connection to the interest of the engineering community in building walking vehicles, there has been a search for more mathematical models for the study of gaits. Such models could be used in the design and control of walking vehicles.

Walking robots are being developed to provide autonomous mobile robots the ability to traverse certain types of terrain (like, flat terrain or mountain, or other rocky terrain, in indoor environments whit steps, stairways) in a more efficient and stable manner than more conventional robots, using wheels or tracks.

Another significant advantage of legged robots is their robustness to damages to legs (in specially, in static walking). Legged robots are able to continue to walk against a fault in a leg. They may maintain static stability even if a leg is broken so that is cannot support the robot body. Adaptation to a leg failure is one of the most important requirements for robust walking of legged robots, because the repair of the failed leg is almost impossible after legged robots have been launched in most applications. From their characteristics of having multi-legs, walking robots have inherent fault tolerance capability against a leg failure since a failed leg for itself may not cause catastrophic failure or instability in static walking. Among various leg failures, a locked joint failure is one of common failures that can be frequently observed in dynamics of robot manipulators, as discussed by Lewis [1].

If failed joints are supposed to be locked individually, a

single joint failure reduces the number of degrees of freedom of the robot manipulator by one and reduces its workspace to a certain limit.

In this paper, we focus our concern on the problem of kinematics constraints of the failed leg in fault-tolerant gaits for a locked joint failure and, also, on the dynamic control of the robot's leg in fault condition using fuzzy theory.

II. MACHINE MODEL

This paper assumes the general hexapod model shown in Fig. 1. The legs are placed symmetrically about the longitudinal axis. C is the center of gravity of the machine and the origin of the machine body's coordinate system $X-Y$. Dashed rectangles are working areas of legs, which represent pre-specified regions where each leg can reach at the present position of the body. Though the legs' reachable working areas can be any shape so long as they can be approximated to straight sided polygons, the conventional rectangular layout is used in this paper for convenience, as discussed by Ralpa and Pai [5], Lee and Hirose [6], Yang [7,8]. D_x and D_y are the length and width of a working area, respectively. It is assumed that the stroke pitch of a leg is the same as the length of a working area D_x , that is, the working area of a leg is adjacent with those of neighbouring ipsilateral legs, as shown in the figure.

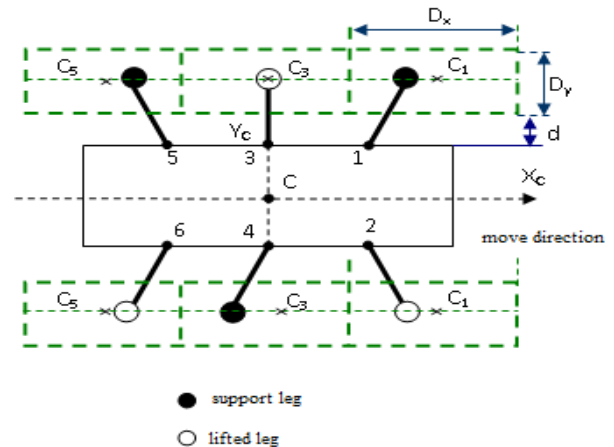


Fig. 1. A general model of a six-legged walking machine

A leg attached to the quadruped robot has the geometry of the articulated arm as showed in Fig. 2. This model has two rigid links and three revolute joints; the lower link is connected to the upper link via an active revolute joint and the upper link is connected to the body via an active revolute joint which is parallel with the body's longitudinal axis.

Manuscript received September 1, 2011.

Cristina Floriana Resceanu is with the Mechatronics Department, Faculty of Automation, Computers and Electronics, Decebal Blvd., no.107 University of Craiova, Romania (e-mail: cristina@robotics.ucv.ro).

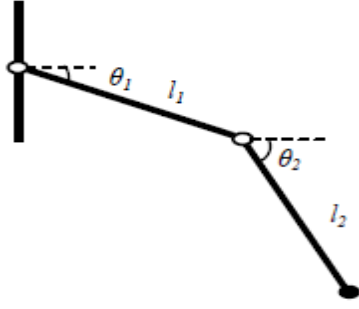


Fig. 2. Generalized coordination of quadruped leg

A. Kinematic Constraint

In this section, we show that there exists a range of kinematic constraints which the configuration of the failed leg should satisfy for guaranteeing the existence of the previous fault-tolerant gait. We assume that a locked joint failure occurs to any joint of leg 1.

Case I: Failure of Joint One

When joint one of leg is locked because failure, the leg can swing only the second link by the knee joint in the lateral direction. The resulting reachable region is thus of an

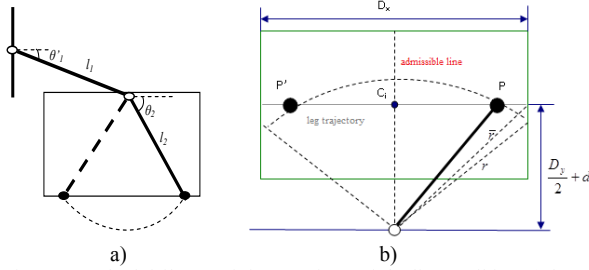


Fig. 3. Locked failure at joint one in straight-line walking a) lateral view; b) plane view

arc shape as is shown in Fig. 3.

Depending on the features of its lateral motion, the leg can be placed on the inner foothold position P' of the outer position P , as show in Figure 3.b), or cannot be placed on the ground in the worst case, as discussed by Yang and Kim [9], Pană [4].

The kinematic constraint for guaranteeing such foothold positions can be described as

$$(D_y/2) + d \leq r \leq \bar{r} \quad (1)$$

where r is the radius of the arc and $\bar{r}$ is the distance between the leg attachment point and the front (or rear) boundary of the foot trajectory projected onto the X-Y plane.

If the radius of the arc r is in the above range, there exists at least one intersection point of the arc and the foot trajectory. For describing (4) in terms of joint angles and robot parameters, let us rewrite $\bar{r}$ and r as

$$\bar{r} = \frac{1}{2} \sqrt{D_x^2 + (D_y + 2d)^2} \quad (2)$$

$$r = l_1 \cos \hat{\theta}_2 + l_2 \cos \theta_3 \quad (3)$$

where $\hat{\theta}_1$ is the locked angle of joint two. Note that r is identical to the length of the leg projection onto the working area. Since the robot body is supposed to have a constant altitude, the angle of joint three θ_2 should also remain the same in the support phase. Substituting (2) and (3) into (1) leads to

$$(D_y/2) + d \leq l_1 \cos \hat{\theta}_1 + l_2 \cos \theta_2 \leq \frac{1}{2} \sqrt{D_x^2 + (D_y + 2d)^2} \quad (4)$$

The above result prescribes the kinematic constraint of locked angle $\hat{\theta}_1$ which guarantees the existence of the fault-tolerant gait for straight-line walking proposed by Chen [2].

Case II: Failure of Joint Two

The motion of leg in the case of a locked failure at joint two is almost similarly as that of a locked failure of joint one. If joint two, or the knee joint, is locked because failure, the leg is reduced to a manipulator with one link and one revolute joint as discussed by Yang and Kim [9], Pană [4]. The reduced reachable region in the working area is an arc as in Fig. 4.

The lateral motion of the failed leg is similar, the leg is moved only by joint one and the second link is passively lifted associated with the lift-off of the first link.

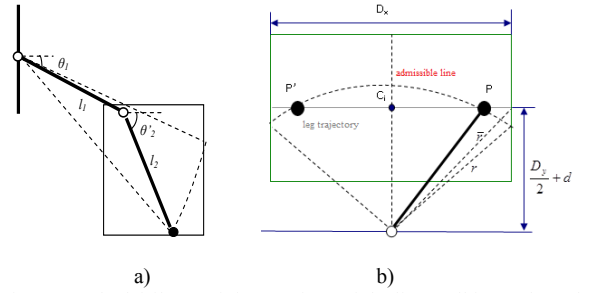


Fig. 4. Locked failure at joint two in straight-line walking a) lateral view; b) plane view

The kinematics constraint for the existence of the fault-tolerant gait is the same as the case of joint one:

$$(D_y/2) + d \leq l_1 \cos \theta_1 + l_2 \cos \hat{\theta}_2 \leq \frac{1}{2} \sqrt{D_x^2 + (D_y + 2d)^2} \quad (5)$$

where $\hat{\theta}_2$ is the locked angle of joint three.

B. Dynamic model of leg

Dynamic equations of robot's leg are the following, as discussed by Pana [3]:

$$\tau_1 = M_{11}\ddot{\theta}_1 + M_{12}\ddot{\theta}_2 - c\dot{\theta}_2^2 - 2c\dot{\theta}_1\dot{\theta}_2 + G_1 + F_{ex1} \quad (6)$$

$$\tau_2 = M_{12}\ddot{\theta}_1 + M_{22}\ddot{\theta}_2 + c\dot{\theta}_1^2 + G_2 + F_{ex2} \quad (7)$$

where:

$$\begin{aligned} M_{11} &= m_1 l_{c1}^2 + I_1 + m_2 (l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos \theta_2) + I_2 \\ M_{22} &= m_2 l_{c2}^2 + I_2 \\ M_{12} &= M_{21} = m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos \theta_2 + I_2 \\ c &= m_2 l_1 l_{c2} \sin \theta_2 \\ G_1 &= -g(m_1 l_{c1} \cos \theta_1 + m_2 [l_{c2} \cos(\theta_1 + \theta_2) + l_1 \cos \theta_1]) \\ G_2 &= -gm_2 l_{c2} \cos(\theta_1 + \theta_2) \\ F_{ex1} &= -F_z [l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2)] + \\ &+ F_x [l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2)] \\ F_{ex2} &= -F_y l_2 \cos(\theta_1 + \theta_2) + F_x l_2 \sin(\theta_1 + \theta_2) \end{aligned} \quad (8)$$

G_1 and G_2 are gravitational moments due to mass m_1 and m_2 , F_{ex1} and F_{ex2} environmental interaction of the foot with the ground. During stepping, when foot is in transfer phase, F_z the foot weight and $F_x = 0$. When the foot is in contact phase with the ground and the robot is a quadruped, then $F_z = 1/4$ of the whole weight of the robot. In this case, $F_x = fF_z$, where the coefficient of friction. In following simulations we will consider that $m_1 = m_2 = m$, $l_1 = l_2 = l$ and $l_{c1} = l_{c2} = l/2$.

III. FUZZY CONTROL OF LEG POSITION

We propose a closed loop control system to achieve a

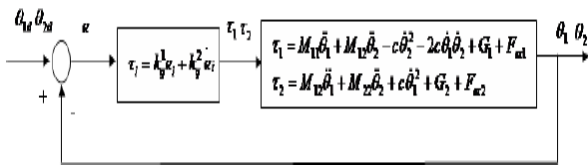


Fig. 5. Closed-loop control system to achieve a desired position of the foot

desired position using the mathematical model of the leg as Error of the control system will be defined by:

$$e_i = \theta_i - \theta_{id}, \quad i \in [1,2] \quad (9)$$

We proposes a control law of the next form:

$$\tau_i = k_q^1 e_i + k_q^2 \dot{e}_i \quad (10)$$

where k_q^1, k_q^2 are positive factors.

In order to analyze the response of the system, this test was done by fuzzy control technique using a PD-type fuzzy controller. Membership functions used to define the two inputs and one output are shown in Fig. 6, their labels are rendered in a manner so as to be easily identifiable, and the rule base is described in the table below, where N - negative, NZ - negative zero, PZ - positive zero, P - positive. Each measure is assigned four labels of triangular form. Surface caused by IF-THEN type rules used is illustrated in

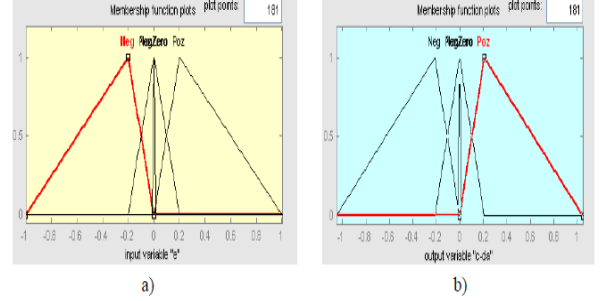


Fig. 6. Type of membership functions) for input and b) output

TABLE I
DEFINITION OF RULE BASE

$\begin{matrix} e \\ \backslash \end{matrix} \begin{matrix} \dot{e} \end{matrix}$	N	NZ	PZ	P
P	N	N	P	P
PZ	N	NZ	PZ	P
NZ	P	PZ	NZ	N
N	P	P	N	N

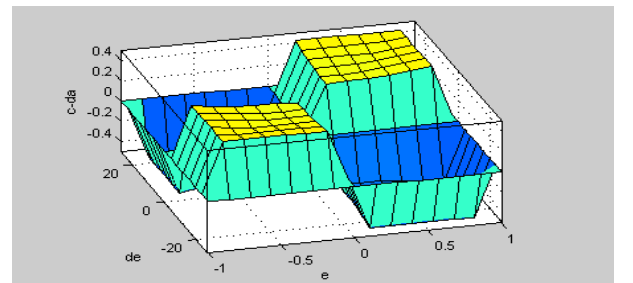


Fig. 7. Surface of rules

A. Control in normal condition

The simulation results are suggestive illustrated in Figure 8, which can track initial positions, final and intermediate positions. Mechanical parameters of the system are $m = 1400g$, length of each link is the total length of the leg is $L = 0.8m$. We consider the initial position of the foot is lifted to its position, as discussed by Pana [4].

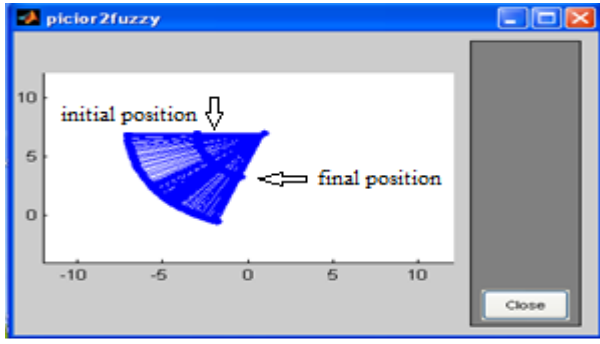


Fig. 8. Evolution of foot to the desired position

For a quantitative understanding of system evolution, as represented in Figure 9, phase portrait of the movement is represented, where we considered the overall error of control system is defined by

$$e(t) = q(t) - q_d(t) \quad (9)$$

$$\dot{e}(t) = \frac{\partial q(t)}{\partial t} \quad (10)$$

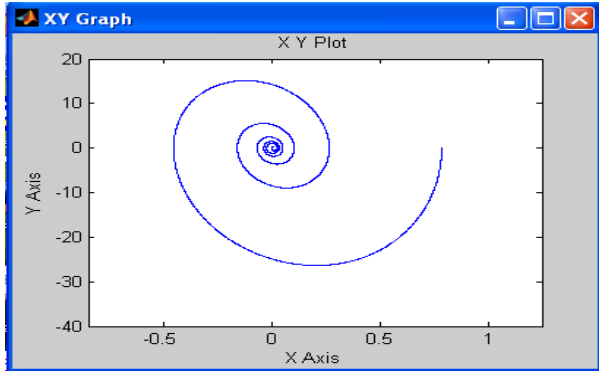


Fig. 9. Phase portrait

B. Fault Condition

We propose a closed loop control system to achieve a desired position of leg with a blocked joint, as shown following:

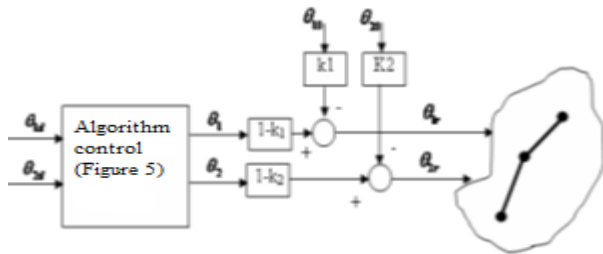


Fig. 10. Control system to achieve a desired position of the foot when blocking a joint

According to the figure, commands were sent to foot the expression:

$$\theta_{ir} = (1 - k_i)\theta_i + k_i\theta_{i0}, \quad i=1,2 \quad (11)$$

where:

$k_i=0$, when joint actuator is in good condition;

$k_i=1$, when joint actuator is locked in an arbitrary angle θ_{i0}

The vector of commands has the following expression:

$$\Theta_r = \begin{bmatrix} \theta_{1r} \\ \theta_{2r} \end{bmatrix} = (I - K_D) \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} + K_D \begin{bmatrix} \theta_{10} \\ \theta_{20} \end{bmatrix} \quad (12)$$

where:

$$K_D = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \quad (13)$$

is matrix of fault. When $K_D \equiv 0$, it can make the claim that the system works correctly. The vector, $[\Theta_0]^T = [\theta_{10} \ \theta_{20}]$ represent values of blocking angles. Respect to Figure 10 and relations (12) and (13), we study the behaviour of leg by the choosing particular structures for matrix of fault.

Case I: Failure of Joint One

The simulation results are suggestive illustrated in Figure 11, which can track initial positions, final and intermediate positions. Mechanical parameters of the system are $m = 1400g$, length of each link is the total length of the leg is $L = 0.8m$. We consider the initial position of the foot is lifted to its position and the first joint is blocked at angle $\theta_{10} = -\pi/4$

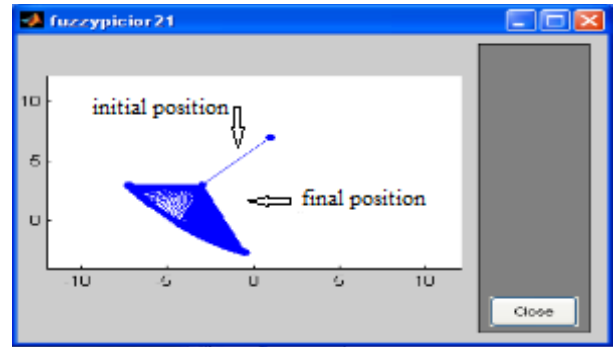


Fig. 11. Evolution of foot to the desired position

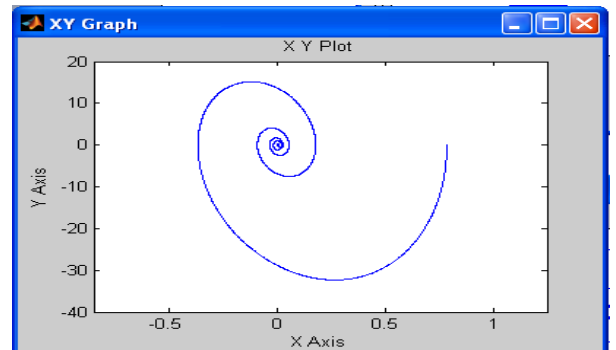


Fig. 12. Phase portrait

Case II: Failure of Joint Two

We consider the initial position of the foot is lifted to its position and the first joint is blocked at angle $\theta_{20} = \pi/2$. The simulation results are suggestive illustrated in Figure 13, which can track initial positions, final and intermediate positions.

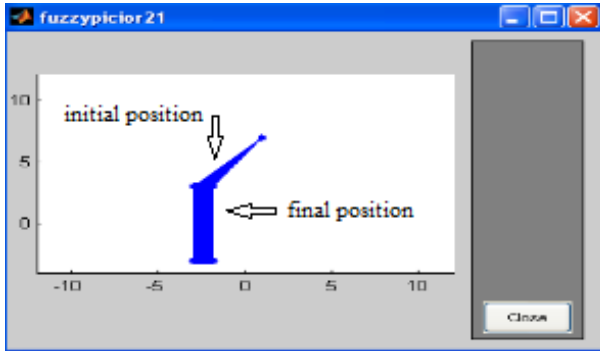


Fig. 13. Evolution of foot to the desired position

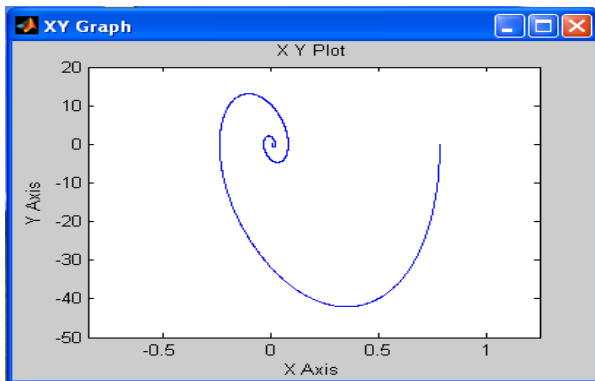


Fig. 14. Phase portrait

The simulations presented were obtained the following results:

-Figure 8 illustrates the motion-control robot right leg, which is highlighted in Figure 9 the error and its derivative tend to zero.

-Figure 11 illustrates motion-control when blocking first leg joints and in Figure 12 presents the evolution of the error and its derivative tend to zero.

-Figure 13 illustrates motion-control when blocking second leg joints and in Figure 14 presents the evolution of the error and its derivative tend to zero.

IV. CONCLUSION

In conclusion, a fault-tolerant fuzzy control is proposed for robot's leg. This control is based on the dynamics of the leg model using fuzzy algorithms. The fault-tolerant control can be applied to a generic class of actuator faults that are second-order differentiable. The effectiveness of the proposed fault-tolerant control is illustrated through experimental results.

ACKNOWLEDGMENT

This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007-2013.

REFERENCES

- [1] C.L. Lewis, A.A. Maciejewski, Fault tolerant operation of kinematically redundant manipulators for locked joint failures, *IEEE Trans. Robot. Autom.*, Vol.13, No.4, 1997, 622–629
- [2] F.T. Chen, H.L. Lee, D.E. Orin, Increasing the locomotive stability margin of multilegged vehicles, *Proc. IEEE Int. Conf. Robot. Automat.*, 1999, 1708–1714
- [3] C. Pana (Resceanu), V.I. Stoian, A Fault-Tolerant Control System for a Hexapod Robot, 4th Conference Mechatronic System and Materials 2008, MSM 2008 Bialystok, Poland, July 14-17, 2008
- [4] C. Pana (Resceanu), Control algorithms for legged robots in fault conditions, Ph.D. thesis, Dept. Mechatronics., University of Craiova., 2009
- [5] S.K. Ralphi, D.K. Pai, Fault tolerant locomotion for walking robots. In: *Proc. IEEE Int. Symp. on Comput. Intell. in Robot. Automat.*, pp. 130–137, 1997
- [6] Y.J. Lee, S. Hirose, Three-legged walking for fault-tolerant locomotion of demining quadruped robots, *Adv. Robot.* 16(5), 415–426, 2002
- [7] J.M. Yan., Fault tolerant gaits of quadruped robots for locked joint failures. *IEEE Trans. Syst. Man Cybern. C.* 32(4), 507–516, 2002
- [8] J.M. Yang, Crab walking of quadruped robots with a locked joint failure. *Adv. Robot.* 17(9), 863–878, 2003
- [9] Yang J.M., Kim J.H., Fault-tolerant locomotion of the hexapod robot. *IEEE Trans. Syst. Man Cybern. B.* 28(1), 109–116, 1998

An Integrated System for Wireless Monitoring of Chronic Patients and Elderly People

C. Rotariu, H. Costin, Gladiola Andrusac, R. Ciobotariu and F. Adochiei

Abstract — In the last years the demographic changes and ageing of population increase health care demand. Increasing number of chronic patients and elders requires close attention to their health conditions. In this paper we present the realization of a wireless monitoring system capable to measure process and transmit patient's physiologic signals (electrocardiogram, respiratory rhythm, arterial saturation of oxygen, blood pressure and body temperature) to a central medical server. The use of our system is suitable for continuous long-time monitoring, as a part of a diagnostic procedure or can achieve medical assistance of a chronic condition. We use custom developed and commercially available devices, low power microcontrollers and RF transceivers that perform the measurements and transmit them to the Personal Server. The Personal Server, in form of a PDA, that running a monitor application, receives the physiological signals from monitoring devices, activates the alarms when the monitored parameters exceed the preset limits, and communicates periodically to the central server by using WiFi or GSM/GPRS connection.

I. INTRODUCTION

IN the last years, the demographic changes in European societies became a global phenomenon, resulted from lower fertility rates and higher life expectancy [1]. All countries in Europe are experiencing an increasing health care demand because of an ageing of their population and an increasing number of people with chronic diseases. Also, chronic diseases represent the major causes of death all over the world.

Patient monitoring refers to the continuous observation of repeating events of physiologic function to guide therapy or to monitor the effectiveness of interventions. Historically, these medical instruments are designed to be used by highly trained personnel, in the intensive care units and operating rooms of hospitals, and thus such instruments are not applicable for home monitoring.

Successful trauma management requires accurate monitoring of several important physiological parameters,

so that proper action can be taken to help maintain critical functionality.

The main vital signs routinely monitored by medical professionals and healthcare providers include the following: electrocardiogram, respiratory rhythm, arterial saturation of oxygen, pulse rate, blood pressure (not considered a true vital sign, but is measured with other vital signs) and body temperature. Vital signs are useful in detecting or monitoring medical problems and can be measured at hospital, at home, at the site of a medical emergency, or elsewhere. Over time, patient monitoring has moved from invasive to non-invasive measurements, which decrease patient risk of infection.

Recent advances in integrated circuits, wireless communications and physiological sensing open the way to miniature, lightweight, low power and intelligent monitoring devices.

Wireless devices had recently a significant development being considered one of the most important technologies of the XXI century. This is due, first of all, to the fact that the mode of transmission (radio waves) completely eliminates data link cables.

Through wireless connectivity is achieved infrastructure and mobile support for real-time monitoring of the patient regardless of patient location, and tracking system for responding in case of alarm (emergency).

Sampling, processing and extracting information from biomedical signals is achieved by appropriate methods, based on performance electronic devices, including computers focused mainly to signal processing.

Wireless monitoring represents a medical practice that involves remotely monitoring patients who are not at the same location as the health care provider. Generally, a patient have a number of monitoring devices at home, and the monitoring results will be transmitted to the database server from the monitoring center. There are several categories of patients who may need continuous monitoring or intensive care. Chronic patients, recovering from serious illness, are often placed in special units, where vital signs can be watched constantly by the use of classic monitoring devices.

The computer-assisted measurements involve unwieldy wires between sensors and monitoring devices that are not very comfortable for normal activity [2]. In order to avoid this situation, we use wireless devices, based on low power microcontrollers, having radio micro-transmitters and allowing an autonomic movement of the subject.

Manuscript received June 17, 2011. This work was supported by the Romanian Ministry of Education and Research, within PNII Programme (www.cnmp.ro/Parteneriate), contract No. 11-067/2007

C. Rotariu, H. Costin, and Gladiola Andrusac are with the University and Medicine and Pharmacy "Gr. T. Popa" Iasi, Faculty of Medical Bioengineering, 700454 ROMANIA phone: 0040232213573; e-mail: cristian.rotariu@bioinginerie.ro.

R. Ciobotariu is with the "Gh. Asachi" Technical University of Iasi, Faculty of Electrical Engineering, ROMANIA

F. Adochiei is with the "Gh. Asachi" Technical University of Iasi, Faculty of Electronics, Telecommunications and Information Technology, ROMANIA.

Healthcare specialists can receive information regarding patient condition that has a longer time span than a patient's normal stay in a hospital and this information has great long-term effects on home health care, including reduced expenses for health care. Physicians also have more accessibility to the healthcare specialists, allowing the physician to obtain information on diseases and provide the best health care available. Moreover, patients can thus save time, money and comfort.

II. MATERIALS AND METHODS

We propose a system architecture made up of three main components, represented in Figure 1:

- (1) a wireless body area network of devices (WBAN), attached on elderly person or the chronic patient, based on low power microcontrollers and RF transceivers that perform the measurements of physiological signals;
- (2) a user Personal Server, in form of a Personal Digital Assistant (PDA), running a monitoring application which receives the raw signals from wireless devices, performs a local signal processing, activates the alarms when the monitored parameters are above the limits, and communicates periodically to the monitoring center through the Internet by using WiFi or GSM/GPRS connection;
- (3) a central medical server for management of database related to registered users, receives and stores the data received from the Personal Servers and is responsible for broadcasting medical records and alarms to the users (patient's physician, healthcare specialist or ambulance center).

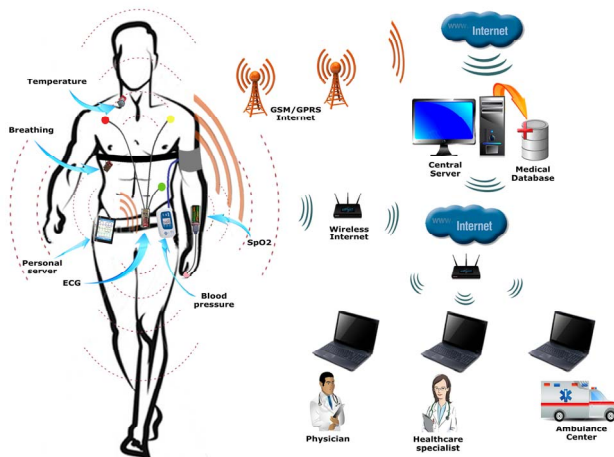


Fig. 1. Integrated system for wireless monitoring – network architecture

The WBAN is composed of the different types of devices for continuous measurement of the following physiological signals: electrocardiogram (ECG), heart rate (HR), respiratory rhythm (RR), oxygen saturation (SpO_2), arterial blood pressure (BP) and body temperature (BT).

The proposed system architecture is centered on a medical database server which is designed to service registered users and it is connected to medical personnel

and healthcare professionals.

The user must wear a WBAN of devices and a PDA attached on the body. Each wireless device samples the physiologic signal and transfers the raw data to the PDA by using a network protocol (IEEE 802.15.4). The PDA performs the WBAN configuration and management (acts as a coordinator), receives the raw signals from devices, and, after a local signal processing according to the specific monitored feature, the processed data are transmitted periodically (at a preset interval of time), via Internet, to the medical database server. If one or more of the monitored parameters are above the preset limits, an alarm is sent in short time to the monitoring center.

The medical database server stores the medical records of registered users and provides services to the physicians, healthcare specialists or ambulance centers.

The WBAN is built by using custom developed and available commercially devices for physiologic signals measurement and a low power microcontroller board (evaluation module eZ430-RF2500 from Texas Instruments) for wireless communication.

The eZ430-RF2500 uses the MSP430F2274 16 bit microcontroller, providing all the hardware and software for the MSP430F2274 microcontroller, and Chipcon CC2500 2.4GHz wireless transceiver, designed for low-power wireless applications. Operating on the 2.4 GHz unlicensed industrial, scientific and medical (ISM) bands, the CC2500 provides extensive hardware support for packet handling, data buffering, burst transmissions, authentication, clear channel assessment and link quality. The radio transceiver is also interfaced to the MSP430F2274 microcontroller by using the serial peripheral interface and has 5m/10m range indoors/outdoors.

Based on the system architecture illustrated in Figure 1 we have developed a prototype of a WBAN of devices for physiological signal measurement, represented in Figure 2.

The ECG signals were acquired by using a 3-lead custom developed ECG amplifier. The amplifier has for each channel a high gain (500), a high input impedance ($>10M\Omega$) and common mode rejection ($>90dB$), a cut-off frequency around 35Hz and is AC coupled.

For SpO_2 and blood pressure measurements we used a commercially available Micro Power Oximeter Board (from Smiths Medical) [3], and UA-767PC blood pressure monitor (from A&D). These devices communicate with the eZ430-RF2500 through asynchronous serial channel.

Respiratory rhythm measurement was performed by using a piezoelectric respiration transducer [4]. The respiration transducer used, PNEUMOTRACE (from UFI), placed around the thorax generates a high-level linear signal in response to changes in thoracic circumference, associated with respiration.

For the body temperature measurements we used the

TMP275 temperature sensor (from Texas Instruments) directly connected to the eZ430-RF2500 by using the I²C bus.

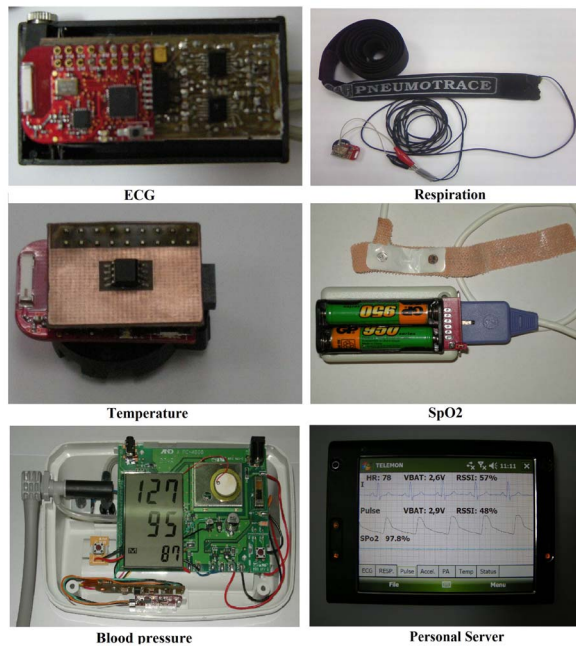


Fig. 2. The wireless body area network of devices

Personal Server was implemented by means of a PDA (HTC X7500) running Windows Mobile 5 as operating system. The software working on PDA was written by using C# from Microsoft Visual Studio 2008. The software displays temporal waveform of physiologic signals, numerical values for monitored parameters and the status of the device (the battery voltage and distance of the device from the PDA). The distance is represented in percent of 100 computed based on RSSI (received signal strength indication measured on the power present in a received radio signal).

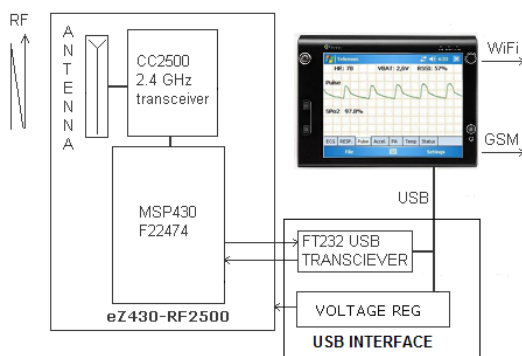


Fig. 3. The Personal Server (PDA)

The USB interface is realized by using a serial to USB transceiver (FT232BL from Future Technology Devices International) and enables the eZ430-RF2500 to remotely send and receive data through USB connection using the MSP430 Application UART. It also contains a low dropout

voltage regulator (TPS77301 from Texas Instruments) to provide 3.3V to the eZ430-RF2500.

Wireless data transmission between the devices and PDA may be performed by using Bluetooth or WiFi standard, but they are more expensive, consume more power and are useful for applications that require high bandwidth. As a limitation Bluetooth allows only a limited number of nodes to communicate. The wireless protocol used to transfer data from device to PDA was *SimpliciTi* (from Texas Instruments). *SimpliciTi* is an open-source protocol for networks that typically contain battery operated devices which require long battery life and low data rate.

The software running on the PDA receives the real-time user raw data from wireless devices and performs a local signal processing in order to detect anomalies. When an anomaly is detected in the measured physiological parameters, the software application sends an alarm to the medical database server.

Table I presents the default physiological conditions that can cause alarms [5]. These values are used by the alarm routine running on PDA. The healthcare specialist may modify these values according to user age or weight.

III. RESULTS

A. Server Side

TABLE I
DEFAULT PHYSIOLOGICAL CONDITIONS FOR ALERTS

Type of alarm	Physiological condition
low oxygen saturation	SpO2 < 90%
heart bradycardia	HR < 40bpm
heart tachycardia,	HR > 120bpm
low respiratory rate	RR < 10rpm
high respiratory rate	RR > 30rpm
blood pressure	systolic BP < 90mmHg or
hypotension	diastolic BP < 60mmHg
blood pressure	systolic BP > 160mmHg or
hypertension	diastolic BP > 100 mmHg
low body temperature	BT < 35°C
high body temperature	BT > 38°C

The whole system acts as a client/server application [6]. The medical information in form of personal information about user, history of his/her illness, results of physical examination, laboratory tests, physiological signals and medical images (x-ray, computed tomography or magnetic resonance images) needs to be distributed among all registered users.

The medical database server application was developed by using the Microsoft Visual Studio 2008. The database is based on SQL Server 2008 and contains ten relational tables. The diagram of the database structure is represented in the Figure 4.

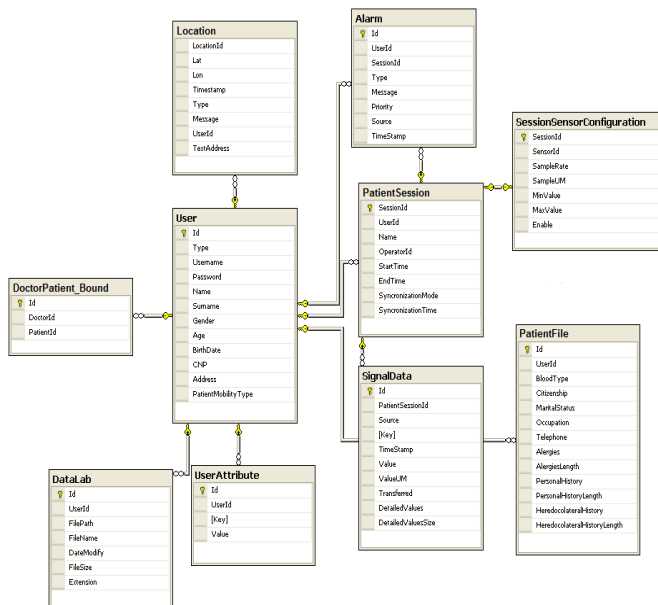


Fig. 4. The structure of the relational database

The *User* table contains the user's basic personal health information, the physiologic signals are stored in the *SignalData* table, the results from laboratory tests or medical images are stored in the *DataLab* table, the alarms received from the users are stored in the table *Alarm*, the history of user illness and allergies are stored in the table *PatientFile*, and the user location based on GPS coordinates is stored in the table *Location*.

B. Client Side

The client application (called *MedApp*), including the communication with the medical database server, was developed to run on windows based systems. While the server is running, the client can start a session from anywhere in the Internet by providing a proper log in and password on the *MedApp*. The display area has been divided into three parts: the graphics area, the user list area, and the buttons area. The graphics area is used to display temporal waveforms retrieved from medical database server.

When an alarm is received from the user, the client's software application generates an alert in the user interface. Figure 5 represents an ECG type HR alert in the client software application.

In order to track the location of users we integrated into our monitoring system a GPS sensor. The ability to track the location of the users outdoor allows the ambulance center quickly locate a specific user who needs medical assistance immediately. The ambulance center has the possibility to locate the user with the aid of TelemonMap application. This application uses a vectorial map with 16 layers, created with GIS NetSET Map Professional.

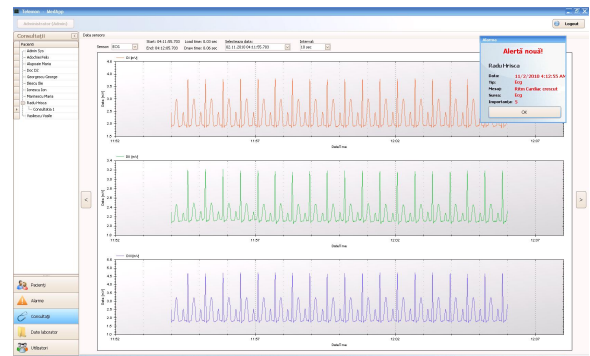


Fig. 5. MedApp – ECG signals and an HR alert

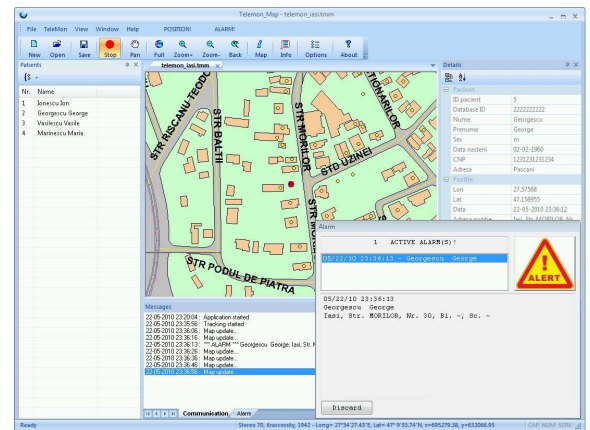


Fig. 6. TelemonMap user interface

IV. CONCLUSION

We have developed a prototype of an integrated system for wireless monitoring. The system tried to meet the diverse needs of users, minimizing cost and maximizing extensibility and reliability.

REFERENCES

- [1] R. Muenz, "Aging and Demographic Change in European Societies," *Main Trends and Alternative Policy Options*, No. 0703 in SP Discussion Paper. Social Protection Advisory Service – The World Bank, Washington, DC, USA, 2007.
- [2] A. Milenkovic, C. Otto, E. Jovanov, "Wireless sensor networks for personal health monitoring: Issues and an implementation", *Computer Communications* (Special issue: Wireless Sensor Networks: Performance, Reliability, Security, and Beyond, Vol. 29 2006, pp. 2521-2533.
- [3] F. Adochiei, C. Rotariu, R. Ciobotariu, H. Costin, "A Wireless Pulse Oximetry System for Patient", *Advanced Topics in Electrical Engineering*, Proc. of the 7th International Symposium ATEE 2011, pp.155-158.
- [4] R. Ciobotariu, F. Adochiei, C. Rotariu, H. Costin, "Wireless Breathing System for Long Term Telemonitoring of Respiratory Activity", *Advanced Topics in Electrical Engineering*, Proc. of the 7th International Symposium ATEE 2011, pp.635-638.
- [5] T. Gao, D. Greenspan et. al., "Vital Signs Monitoring and Patient Tracking Over a Wireless Network", *Proceedings of the 27th Annual International Conference of the IEEE EMBS*, Shanghai 2005.
- [6] J. Bai, B. Hu, Y. Zhang and D. Ye, "A Communication Server for Telemedicine Applications", *IEEE Transactions on Information Technology in Biomedicine*, Vol. 1, No. 3, 1997, pp. 205-209.

Scalable Distributed Agent-Based English Auction Server

Liviu Sandu, Raluca Sboru, Sorin Ilie, and Costin Badica

Abstract— This paper presents a distributed agent-based server architecture for English auctions. Bids of auction participants are handled by a hierarchical organization of agents that can be efficiently distributed on a computer network. This approach avoids the bottlenecks of bid processing that might occur during periods of heavy bidding, like for example snipping. We present experimental results that show a significant improvement of the server throughput compared with the architecture where one auction manager agent is used for coordinating the participants for each active auction that is registered with the server.

I. INTRODUCTION

IN this paper we present an approach for improving the throughput and response time of an agent-based auction server using a multi-agent hierarchical scheme that can be distributed on a computer network. The baseline of our solution consists of our initial results reported in [2] where we introduced the design and implementation of an open flexible infrastructure for service-based automated negotiations using multi-agent systems. The approach follows the idea introduced in [7] for improving the performance of on-line auction systems using special computing nodes available in active networks that are implemented as proxy agents in our system.

Our initial solution for an agent-based auction server described in [2] integrates the best features of: i) generic software framework for automated negotiations [1], [3]; ii) market architecture for auction development [5]; iii) rule-based declarative representation of auction mechanisms [3],

[4]. The main advantage of this approach is that an auction is seen as a separate service, rather than being entirely incorporated into an e-shop infrastructure. The service can be for example rented and configured by an e-shop that would like to sell its products through an auction, for example for clearing its shelves on 'sales' time.

There are many definitions of the agent concept [6]. For the purpose of this work, by software agent (agent in what follows) we understand a software entity that: (i) has its own thread of control and can decide autonomously if and when to perform a given action; (ii) communicates with other agents by asynchronous message passing; (iii) can be located on an arbitrary machine in a computer network, providing that a certain runtime environment is locally available. Each agent can be referenced using its name, also known as agent identifier [8].

Following [6], auctions provide a general solution to the problem of discrete resource allocation among selfish agents in a multi-agent system. In this paper we focus on English auctions. In this case, the auctioneer announces an initial price for the good (assuming an e-commerce application setting) and auction participants bid increasing amounts of money during a predefined time frame, usually by some minimum increment set by the auctioneer. The agent with the highest bid at the end of the bidding process is declared the winner of the English auction.

When an auction (in particular an English auction) is modeled as a multi-agent system, the agents' interaction through message exchanges is required for the bidding and price update activities. In our previous approach [2], the central agent for auction management was responsible with handling all the communication with and between the auction participants. So, if a large number of participants join the auction, this agent becomes highly stressed. In the approach introduced in [2] the central agent was a system bottleneck, and thus hindered scalability.

In this paper we propose a solution aiming to improve the architecture in [2] by relieving the central agent from part of its heavy message exchanges. Based on the idea that was firstly proposed by [7] for improving the performance of on-line auction systems using special computing nodes, we enhanced our system with the introduction of third party agents called proxies. Each proxy agent will handle a part of the communication with the auction participants.

So, while in our prior approach the auction manager would receive all the bids from the participants, in this

Manuscript received June 24, 2011. This work was partially supported by the strategic grant POSDRU/88/1.5/S/50783, Project ID50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resources Development and partly by the research project “SCIPA: Servicii software semantice de Colaborare si Interoperabilitate pentru realizarea Proceselor Adaptive de business” between Software Engineering Department, University of Craiova, Romania and National Authority for Scientific Research, Romania

L. Sandu is undergraduate student at the Software Engineering Department, University of Craiova, Bvd.Decebal 107, 200440, Craiova, Romania.

R. Sboru is undergraduate student at the Software Engineering Department, University of Craiova, Bvd.Decebal 107, 200440, Craiova, Romania.

S. Ilie is Ph.D. student at the Software Engineering Department, University of Craiova, Bvd.Decebal 107, 200440, Craiova, Romania (e-mail: sorin.ilie@software.ucv.ro).

C. Badica is professor at the Software Engineering Department, University of Craiova, Bvd.Decebal 107, 200440, Craiova, Romania (e-mail: costin.badica@software.ucv.ro).

approach the participants are split into disjoint groups, and each group is managed by a single proxy agent. The participants communicate heavily with their proxies, while proxies pass on to the auction manager agent only the relevant bids while the other bids are filtered out and processed locally, thus reducing the amount of messages handled by the auction manager.

Finally, note that communication between two agents is faster when the agents are located on the same machine than when the agents are located on separate machines. When our auction server is distributed on several computers, some agents will have to exchange messages over the network, thus increasing their communication time, as well as the overall server response time. So, with our solution we also aimed to improve the communication time between agents, when the server is distributed on several computers, by keeping, whenever possible, proxy agents on the same machine with their participant agents.

II. AGENT TYPES

In this section we introduce our English auction server architecture. The auction server is represented by a society of cooperating agents that interact between themselves and with the external environment.

The software infrastructure of the server contains the following types of agents:

- *Auction Service* manages all the active auctions on the server.
- *Auction Manager* manages a single active auction on the server (also known as auction instance in [2]).
- *Auction Directory* manages the registry of active auctions, as well as the identifiers of their Auction Manager agents.
- *Participant* represents a user with the buyer role, registered to a particular auction. The *Participant* agents of a given user are linked to the *Personal Agent* of the user.
- *IParticipant* represents the user with the seller role in a given auction. He is the initiator participant, according to the terminology introduced in [2]. Note that in an English auction the initiator participant has the role of seller.
- *Proxy* handles the bids received from a subset of the participants.
- *Personal Agent* is acting as an interface between the system and the user.
- *Computer Manager* is responsible with the management of *Participant* and *Proxy* agents.
- *Resource Manager* contains a registry of all the agents of type *Computer Manager* from the system.

The server is actually composed of three main parts: core, resource manager, and interface.

The core of the server is represented by *Auction Service*, *Auction Manager*, *Auction Directory*, and *Proxy* agents. It is responsible with the auctions' management and the

coordination of the auction participants by implementing the rules that govern the auction.

The agents *Computer Manager* and *Resource Manager* are composing the part of the server that is responsible with the management of the computational resources on which the server software is ran.

Finally, *Participant*, *IParticipant*, and *Personal Agent* agents compose the server tier that realizes the software interface of the server with the external environment.

Note that the interaction protocols with the core of auction server follow the same rules as the initial proposal of [3] that we also considered in [2]. According to these protocols, a user via his *Personal Agent* can create auctions, subscribe to active auctions, submit bids, receive replies about bid acceptance or rejection, receive notifications about the update of an auction status, and receive notifications about auction termination and auction winner.

III. AGENT INTERACTIONS

In what follows, because of the lack of space, we briefly describe each agent of the enhanced architecture, focusing on some of the details of the internal communication between agents.

The *Auction Service* is the agent that manages the active auctions registered with the server. He implements a specific type of auction – English auction in this case, but in principle an *Auction Service* can be configured to support an arbitrary auction type. The management of active auctions includes the activities of auction creation and auction termination.

The *Auction Manager* is the agent that coordinates the participants to an active auction. There is a separate *Auction Manager* agent for each active auction in the system. The *Auction Manager* is created by the *Auction Service* when a user wants to sell a product through an auction.

The *Auction Manager* has the following responsibilities:

- creates and destroys *Proxy* agents
- creates *Participant* agents and assigns them to the right *Proxy* agent
- notifies all the *Proxy* agents when the value of the highest bid was updated
- accepts or rejects the bids from *Participant* agents that are forwarded by *Proxy* agents
- manages the parameters of the auction, including: auction name, starting price, product name, starting and ending dates
- triggers the auction termination when the time has expired by informing all the *Proxy* agents that the auction has finished
- records the winner and the final price of the auction

The *Personal Agent* is the software artifact residing on the server side that connects the user with the auction server. For each user registered with the server there is

exactly one *Personal Agent* created on the server. Whenever a human user connects to the server, usually using an external program (for example a GUI or a Web browser), a *Personal Agent* assigned to the user is created. We setup our server to be used in a Web environment using the servlet technology on the server side. In this case it was the responsibility of the servlet to create the *Personal Agent*, when the user logged into the system.

The *Personal Agent* allows the user to perform the following operations:

- create a new auction
- subscribe to an existing auction
- submit bids to one of his or her subscribed auctions, via his or her *Participant agent*
- receive notifications about status updates of his or her subscribed auctions
- receive accepts and rejects for his or her submitted bids

The *Participant agent* represents a user registered to a given auction. He acts as a link between the *Auction Manager* and the *Personal Agent* of the user that is connected to an auction. The *Participant agent* forwards the bids received from the *Personal Agent* to the auction server.

Proxy agents form an intermediate layer between the *Auction Manager* and the *Participant agents*. Their main

responsibility is to take over a part of the load that is necessary for bid processing that was initially the responsibility of the *Auction Manager* [2]. Each *Proxy agent* records a local currently highest bid (that can be different from the currently highest bid of the auction recorded by the *Auction Manager*) and updates it regularly based on the notifications received from the *Auction Manager* and bids received from *Participant agents*.

The responsibilities of the *Proxy agent* are:

- filters the bids received from the *Participant agents* passing up to the *Auction Manager* only those bids that are higher than the local currently highest bid known by the *Proxy*
- notifies *Participant agents* about the acceptance or rejection of their bid
- receives notifications about the update of the currently highest bid from the *Auction Manager*, and
- notifies *Participant agents* after receiving such an update from the *Auction Manager*; the local currently highest bid of the *Proxy* is also updated.

The details of the interaction between the *Auction Manager*, the *Proxy agents*, and the *Participant agents* are shown on Fig. 1.

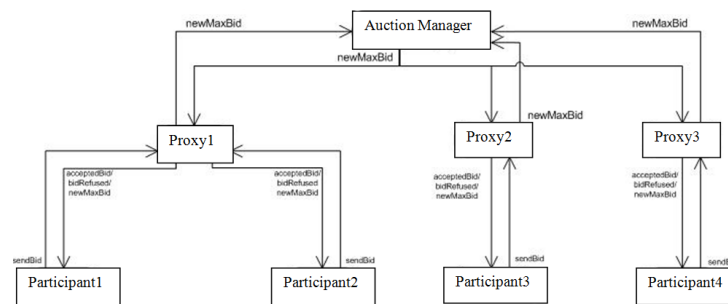


Fig. 1. The interaction between *Auction Manager*, *Proxy*, and *Participant agents*.

A new auction is created when a *Personal Agent* sends to the *Auction Service* a *createAuction* request (see Fig. 2). Then the *Auction Service* creates an *Auction Manager* and an *IParticipant agent* linked to the *Personal Agent* of the user seller. The *Auction Service* then confirms to the

Personal Agent the creation of the *IParticipant agent*. The auction description is also added to the *Auction Directory*. The *IParticipant* and *Auction Manager* agents must further interact during the auction, so the *Auction Service* informs both of them about each other identifier.

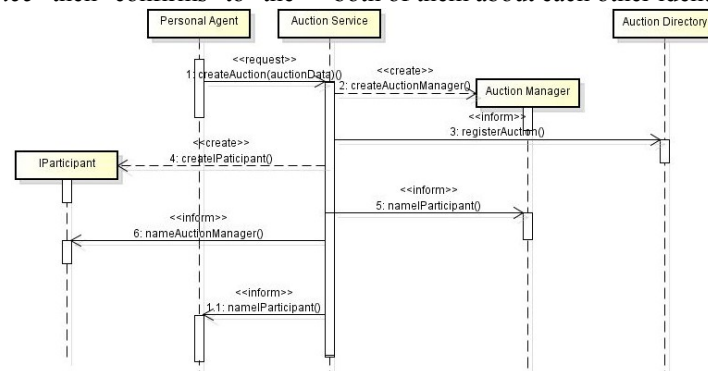


Fig. 2. Agent interactions for initiating an auction

A *Personal Agent* can query the *Auction Service* about existing auctions (see Fig. 3). This interaction happens via the *Auction Service* that has the capability of directly submitting a query to the *Auction Directory* agent. The *Personal Agent* then chooses an auction and contacts the *Auction Manager* of that auction which should create a *Participant* for it.

Let us suppose that on average a *Proxy* agent is managing k *Participant* agents and that the total number of *Proxy* agents is p . This means that the total number of participants is $n = p \times k$. According to our coordination model of an English auction [2], [3], whenever a new bid is accepted by the *Auction Manager*, all the participants must be notified accordingly, and this process obviously will require a time $O(n)$. However, with the new proxy-based hierarchical approach this time will be $O(\max(p, k))$. This value can be optimized to $O(n^{1/2})$ for n participants if we set the constraint that the number of participants managed by each *Proxy* agent will never outnumber the total number of *Proxy* agents connected to the *Auction Manager*.

It follows that *Proxy* and *Participant* agents must be linked into a balanced two-level hierarchical structure rooted at the *Auction Manager* such that the total number of *Proxy* agents equals (or at least, it is almost equal to) the number of *Participant* agents that are linked to each *Proxy* agent. Therefore the dynamic creation and destruction of *Proxy* agents will take into account this balancing constraint.

Proxies are created and destroyed by the *Auction Manager*. A *Proxy* agent is created when a new participant is created and

$$n > p^2$$

where:

- n is the total number of participants including the newly created one.
- p is the number of proxies

The message exchanges that are needed for creating *Participant* and *Proxy* agents are shown on Fig. 4.

The *Auction Manager* takes the decision to create the *Proxy* or *Participant* agents, but their creation is handled by the *Resource Manager* agent, while the actual creation is performed by the *Computer Management* agent. The *Auction Manager* requests the *Resource Manager* agent to choose the right computer where these agents will be created. The *Resource Manager* determines this computer and instructs the corresponding *Computer Manager* agent, which is unique for each computer from our network, to perform creation action.

When a participant quits an auction, if

$$n \leq (p-1)^2$$

then the *Proxy* with the smallest number of participants is deleted, where:

- n is the total number of participants (after the removal of the current participant from the server)
- p is the current number of proxies

The *Participant* agents that were linked to the *Proxy* that will be deleted will be evenly reallocated to the remaining *Proxy* agents with the smallest number of participants.

Each *Participant* agent sends his bids to their designated *Proxy* according to the protocol shown on Fig. 5. This bid is then compared by the *Proxy* to the currently best bid. Larger bids are reported to the *Auction Manager* while smaller bids are rejected with a *refuseBid* message. Then the *Auction Manager* saves the largest bid (*bestBid* state variable) and refuses submitted bids that are smaller than *bestBid*. The refusal is then returned by the *Proxy* agent back to the originating *Participant*. If a bid higher than *bestBid* is received then the *Auction Manager* responds to the *Proxy* with a *bidAccepted* message which it propagates to the originating *Participant*. If the *bestBid* value has changed then the *Auction Manager* informs all the *Proxy* agents. Then each *Proxy* will propagate the value of the new *bestBid* to its *Participant* agents. This forwarding is represented on Fig. 5 as a multicast message that is displayed as a little circle at the end of the message arrow.

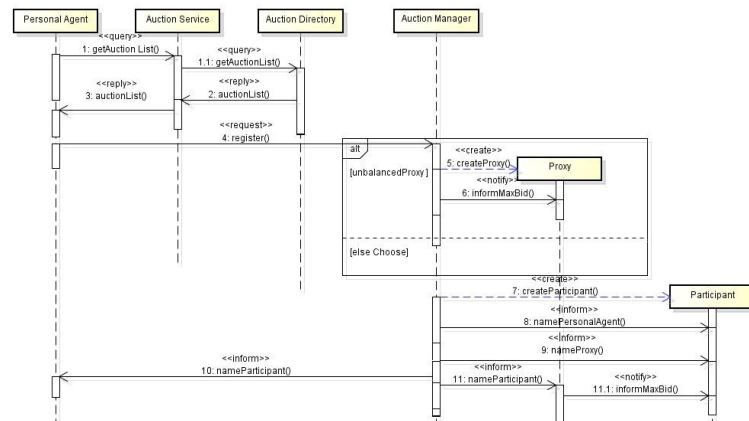


Fig. 3. Interaction protocol for joining an auction

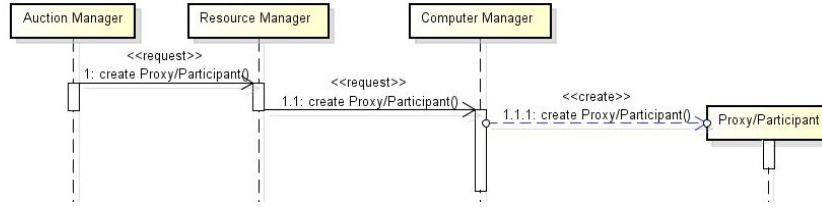


Fig. 4. Interaction protocol for automated creation of *Proxy* and *Participant* agents

Our auction server is designed to run on a network of computers. On each available computer in the network a special agent called *Computer Manager* is created. An agent called *Resource Manager* that registers all the *Computer Manager* agents is created on the same computer as the *Auction Service*. Whenever a *Participant* or a *Proxy* needs to be created, the *Resource Manager* chooses a computer and asks its *Computer Manager* to create the new agent. On each computer there is allowed the creation of a maximum number of *Proxy* agents (currently this number is the same for all computers). When a computer is full the next computer is used. If all computers are full then the computer with the smallest number of proxies or participants (in that order) is used. All *Participant* agents are created on the same computer as their *Proxy* agent

IV. EXPERIMENTS AND RESULTS

We experimentally evaluated the current architecture by comparing it with our initially proposal from [3], where no proxies were used. Then we conducted initial scalability experiments by running our system on 2 and 3 computers. For the experiments we used a network of dual core processors at 2.5 GHz and 1GB of RAM memory. These workstations were interconnected using a high-speed Myrinet interconnection network at 2Gb/s.

In this experiment the participants are allowed to bid automatically, so they were equipped with a bidding strategy to tell them if, when, and how much to bid. The bidding follows a snipping scenario, i.e. as soon as a participant receives a notification that he was outbid by another participant, he immediately submits a higher bid by adding a predefined increment to the value of the currently highest bid. During the snipping scenario the auction server is heavily loaded with bid processing activities.

The starting price of the auction was set to 0. The auction duration was set to 1.5 minutes in all the experiments. The increment value for the participants' strategy was set to 10. The agents were allowed to bid up to a very high reserve price (100000). The auction terminates when its allocated time expires. In our cases, as the reserve prices were set to very high values, the auctions end before any of the agents reaches their reserve price.

Note that whenever a *Participant Agent* bids according to this strategy, he or she must receive an answer confirming or not if the bid was successful. This is very important, as we are in a distributed environment with multiple agents bidding concurrently, and it might happen that even if an agent is choosing a right value to bid, he or she might be outbid by another agent that submits his or her bid almost simultaneously.

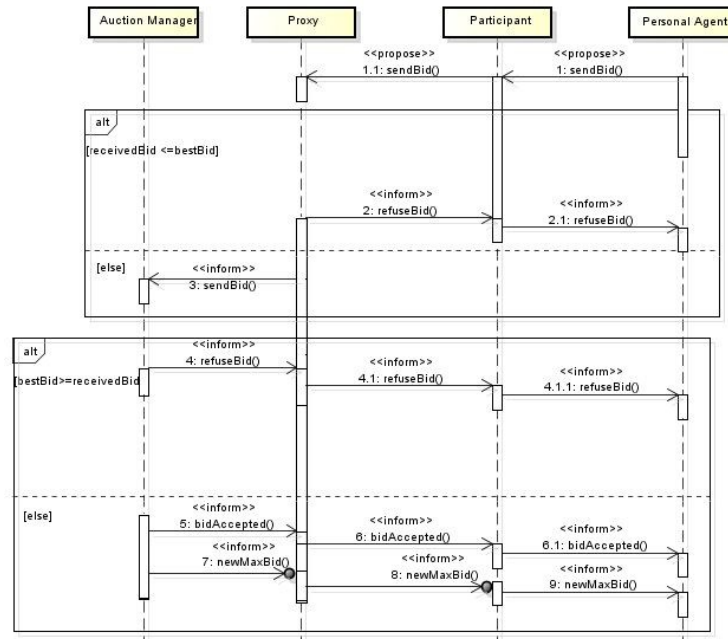


Fig. 5. The bid submission protocol

The following performance measures were recorded in our experiments:

- *Latency* = the average time it takes the system to answer a bid;
- *Throughput* = the number of bids handled per unit of time; this value is calculated by dividing the total number of answered bids by the duration of the auction;

The setup of the experiment assumes running a script that starts the JADE multi-agent platform [8] and automatically creates the *Auction Service* and *Auction Directory* agents. Then the *Personal Agents* are created for each user that participates in the auction.

The *Personal Agent* that initiates the auction was configured to start the bidding in our experiment. He is notified about the number of participants that joined the auction. When a certain given number of participants is reached, this *Personal Agent* allows the *Auction Manager* to start the bidding process.

We ran our experiments using an increasing number of *Personal Agents* and we calculated the performance measures by running the framework on one, two and three computers. In order to also compare with our prior approach we also ran a version of the program without proxies, forcing the *Auction Manager* to handle all bids. In each test we ran only one auction. The case when multiple auctions are run in parallel was left as future work.

The results of our experiments are shown in Fig. 6 and Fig. 7, as well as in Table I and Table II.

TABLE I
LATENCY [MS]

Agents	No Proxy	1 Comp	2 Comp	3 Comp
500	5087.53	2172.14	148.55	104.39
1000	20038.9	8114.66	266.85	217.01
1500	39752.5	8781.39	426.50	277.88
2000	66852.3	11691.2	575.06	452.73

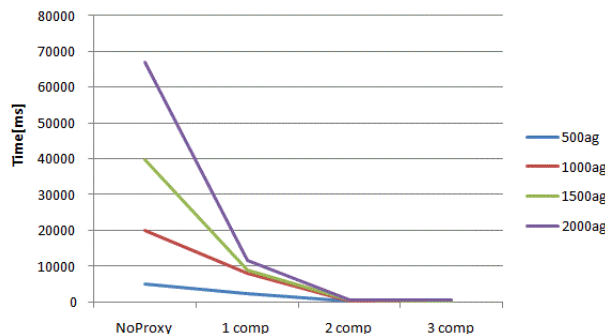


Fig. 6: Latency [ms]

Note that when we ran our system with 2000 bidding agents without proxies only 1273 agents actually got to bid at least once during the allocated time of 1.5 minutes.

TABLE II
THROUGHPUT [NO. OF BIDS/MS]

Agents	No Proxy	1 Comp	2 Comp	3 Comp
500	0.141	1.115	1.527	1.801
1000	0.065	0.949	1.714	1.744
1500	0.035	0.851	1.43	1.86
2000	0.025	0.779	1.502	1.672

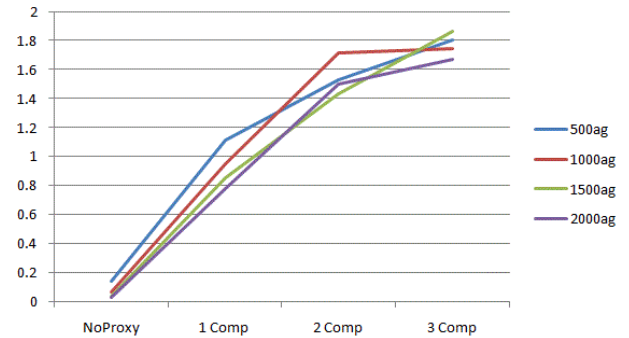


Fig. 7. Throughput [no. of bids/ms]

V. CONCLUSIONS

In this paper we introduced an improved agent-based architecture for an English auction server. The initial experimental results show that our hierarchical scheme of structuring the server using *Proxy* agents and a simple balancing scheme is effective and has good scalability results when the server is distributed on multiple machines. As future work we plan to: (i) strengthen the results by performing more experiments on larger networks; (ii) extend the architecture to other types of auctions.

REFERENCES

- [1] Badica, C., Ganzha, M., and Paprzycki, M. (2007) 'Implementing Rule-Based Automated Price Negotiation in an Agent System', *Journal of Universal Computer Science*, Vol. 13, No. 2, pp.244-266.
- [2] Dobriceanu, A., Biscu, L., Badica, A., Badica, C. (2009) 'The design and implementation of an agent-based auction service', *International Journal of Agent-Oriented Software Engineering (IAOSE)* Vol.3, No.(2/3), pp.116-134, Inderscience Publishers.
- [3] Bartolini, C., Preist, C., and Jennings, N.R. (2005) 'A Software Framework for Automated Negotiation', *Lecture Notes in Computer Science*, Vol. 3390, pp.213-235, Springer.
- [4] Badica, C., Giurca, A., and Wagner, G. (2007) 'Using Rules and R2ML for Modeling Negotiation Mechanisms in E-commerce Agent Systems', *Lecture Notes in Computer Science*, Vol. 4473, pp.84-99, Springer.
- [5] Rolli, D., Luckner, S., Gimpel, H., and Weinhardt, C. (2005) 'A descriptive auction language', *Electronic Markets - The International Journal*, Vol. 16, No. 1, pp.51-62.
- [6] Shoham, Y. and Leyton-Brown, K. (2009) 'Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations', MIT Press.
- [7] Hillston, J. and Kloul, L. (2001) 'Performance investigation of an on-line auction system', *Concurrency and Computation: Practice and Experience*, Vol. 13, No.1, pp.23-41, John Wiley & Sons.
- [8] Bellifemine, F. L., Caire, G., and Greenwood, D. (2007) 'Developing Multi-Agent Systems with JADE', John Wiley & Sons.

An Optimal Feedback Approach to Particle Source Identification in Tokamaks

Enrique SANTIAGO^{*†}, Emmanuel WITRANT^{*}, Marc GONICHE[†] and Frédéric CLAIRET[†]

^{*}GIPSA-lab, Université Joseph Fourier / CNRS, BP 46, 38 402 Saint Martin d'Hères, France

[†]CEA, IRFM, F-13108 Saint Paul-lez-Durance, France

Corresponding author: emmanuel.witrant@ujf-grenoble.fr

Abstract—A new identification technique is proposed to study the plasma phenomena taking place in the so-called *scrape-off layer* and their correlation with the *lower hybrid* radio frequency antenna in the experimental nuclear fusion tokamak reactor *Tore Supra*. A deeper knowledge of the plasma behavior in this region would contribute to the achievement of steady-state controlled thermonuclear fusion for power generation. The proposed approach relies on the design of a feedback/feedforward optimized architecture to solve a blind identification problem. While our paper is mostly focused on experimental results for the studied application, it can provide valuable insights on input estimation and model validation for transport phenomena described by partial differential equations.

Keywords: distributed parameter systems, blind identification, process control.

INTRODUCTION

Recent developments in control theory and controlled thermonuclear fusion research are naturally leading to research topics of common interest that are particularly challenging for both scientific communities. For example, new modeling and identification tools are needed for the understanding and analysis of complex physical phenomena. The representation (qualitative and quantitative) of particles transport at the plasma edge is an example of such topics.

Tokamak experiments, such as *Tore Supra* or *JET*, are equipped with Lower Hybrid (LH) antennas to heat the plasma and drive a non-inductive current. The LH waves are recognized as the most efficient non-inductive current drive sources and their use is forecasted for ITER experiment. A LHCD system has been implemented on *Tore Supra* since more than 20 years to fulfill the quasi-steady-state discharges objective [1] and has been proposed for ITER aiming at extending the pulse duration to 3000 s [2]. The efficiency of such antennas is strongly related to our ability to ensure an appropriate coupling between the waves on the plasma, which directly depends on the electron density in a region called the *scrape-off layer* (SOL), located between the last closed magnetic surface (the *separatrix*) and the wall. This region, as well as the key elements discussed in this work, are depicted in Figure 1. The importance of local density control in the SOL is emphasized in [3], [4].

The electron density in the SOL is directly influenced by the LH input power P_{LH} [3]. Indeed, a small but significant part of this power is absorbed in the plasma edge during the wave propagation to the core. A possible effect of this absorption

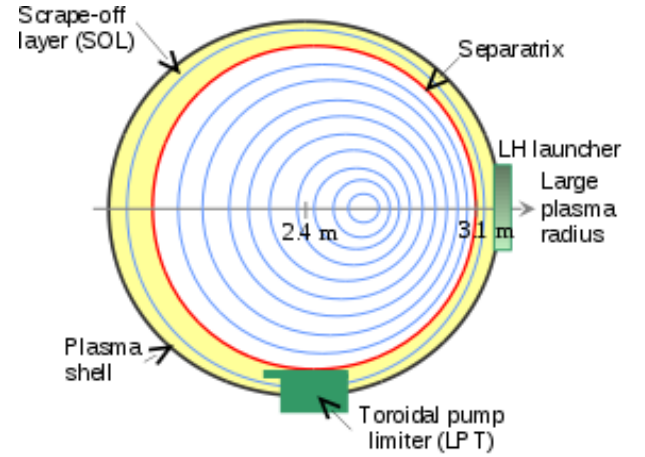


Fig. 1: Tore Supra cross section. Figure courtesy of [5].

is the gas ionization, which results in an increased electron density. This suggests that P_{LH} is a key parameter for the local control of the electron density, and consequently for the coupling efficiency. The development of new models based on experimental measurements that consider the impact of P_{LH} are consequently of prime interest. On *Tore Supra*, the electron density is measured by using a microwave reflectometer that has both a good spatial (≈ 1 cm) and temporal resolution (≈ 2 ms for the data considered).

The aim of this work is to propose an identification method for determining the *source term*, i.e. the number of electrons created per unit time and volume, when the high frequency heating is switched on. To achieve this, a simple particle transport model for the area of non-confined plasma (SOL) and a quasi-steady-state approach was proposed in [5]. Our aim is to develop an appropriate parametric identification method for distributed systems involving the transients, based on the reflectometer measurements. The resulting algorithm then provides an efficient tool for analyzing the coupling phenomena associated with the electron density behavior in the SOL, even if the detailed physical relationships are unknown.

This paper is organized as follows. First, the physical model and identification problem formulation are introduced in Section I. An optimal input identification procedure is then proposed in Section III, along with experimental results. The inclusion of shape constraints on the admissible inputs is

considered in Section IV. Finally, the physical transport model is revised with a multizone approach in Section V.

I. PHYSICAL MODEL AND PROBLEM FORMULATION

Using specific hypotheses associated with the physical properties of the SOL, the following transport model was established in [5]:

$$\frac{\partial \tilde{n}_e}{\partial t} = D_{\perp}(t) \frac{\partial^2 \tilde{n}_e}{\partial r^2} - \frac{\tilde{c}_s(r,t)}{2\tilde{L}_c(r,t)} \tilde{n}_e + \tilde{s}(r,t) \quad (1)$$

where $n_e(r,t)$ is the electron density, r is the small plasma radius, t is the time, $D_{\perp}$ is the cross-field diffusion coefficient (typically $\sim 1 \text{ m}^2 \text{ s}^{-1}$, which is the value used for the presented simulations), c_s the sound speed, L_c the connecting length along the flux tube to the flow stagnation point and $\tilde{s} = S_{LPT} + S_{LH}$ reflects the particle source induced by the pumped toroidal limiter S_{LPT} and the LH antenna S_{LH} . The tilde superscript denotes the expression of the variables in terms of the tokamak small radius and will be dropped in the remaining of the paper after the radius normalization. The sound speed can be approximated as $c_s \approx \sqrt{(T_e + T_i)/m_i}$, where T_e and T_i are the electron and ion temperatures, and m_i is the ion mass. The ratio T_i/T_e is obtained thanks to the experimental measurements described in [6]. In the case of a toroidal limiter, $\tilde{L}_c$ is deduced from the safety factor q as $\tilde{L}_c \approx 2\pi R_0 q$, where R_0 is the plasma major radius. When no other objects (chamber wall, secondary limiters...) intercept the field lines in the SOL, this connection length has small variations and will be considered as constant.

A Neumann boundary condition $\partial \tilde{n}_e(0,t)/\partial r = 0$ is set at the center while a Dirichlet one $\tilde{n}_e(L,t) = n_{e,L}(t)$, where $n_{e,L}(t)$ is given by the measurements, governs the plasma edge. The data set considered in this paper is characterized by constant and repeated LH impulses that highlight the impact of LH antenna (no other radio frequency source).

As a first approach, this paper is focused on determining the variations of the source term $\tilde{s}(r,t)$ in (1) from the measurements of $\tilde{n}_e(r,t)$. The transport parameters $D_{\perp}$, $\tilde{c}_s$ and $\tilde{L}_c$ are obtained from existing models and measurements. They are assumed to be constant according to a supposed steady state behavior of the plasma transport properties (i.e. these parameters vary slowly in comparison with the state dynamics).

Using the subscripts t and z to denote the time and space derivatives, respectively, where $z = r/L \in [0, 1]$ is the normalized radius, the class of systems considered can be described as:

$$\begin{cases} n_{e,t}(z,t) = \alpha n_{e,zz}(z,t) - \nu n_e(z,t) + s(z,t), \\ n_{e,z}(0,t) = n_{e,z0}(t), \quad n_e(1,t) = n_{e,L}(t), \\ n_e(z,0) = n_{e,t0}(z), \end{cases} \quad (2)$$

where $\alpha = D_{\perp}/L^2$ is the diffusion and $\nu = c_s/(2L_c) > 0$ is the sink term. The existence and unicity of a solution can be proved with the results proposed by [7].

The dynamics (2) can be approximated by ordinary differential equations by discretizing with respect to space (see [8]

or similar textbooks for a detailed analysis of the discretization step). For example, the finite differences method with a central discretization of the diffusion term:

$$\frac{\partial^2 n_e}{\partial z^2} \approx \frac{n_{e,i+1} - 2n_{e,i} + n_{e,i-1}}{\Delta z^2}$$

where $n_{e,i}(t)$ denotes the value of $n_e(z,t)$ at the location $z = z_i$ ($i = 1, \dots, N$, where N is the number of measurements), and the use of the boundary conditions as:

$$n_{e,z}(0,t) \approx \frac{n_{e,1}(t) - n_{e,0}(t)}{\Delta x} = n_{e,z0}(t), \quad n_{e,N+1} = n_{e,L}(t),$$

provides a N -dimensional state-space model of the density distribution. It writes in the standard state-space form:

$$\dot{x} = Ax + BS + w, \quad x(0) = x_0$$

where $x = [n_{e,1} \ n_{e,2} \ \dots \ n_{e,N}]^T$, $S = [s_1 \ s_2 \ \dots \ s_N]^T$, $w = [a_1 n_{e,z0} \ 0 \ \dots \ 0 \ a_2 n_{e,L}]^T$, $a_{1,2}$ are coefficients determined by the spacial discretization scheme, $A \in \mathbb{R}^{N \times N}$ (see [9] for a complete description of the matrix elements and $a_{1,2}$) and $B = I \in \mathbb{R}^{N \times N}$.

II. A FEEDBACK APPROACH TO BLIND IDENTIFICATION

Classical identification methods are available to determine the system behavior (or specific parameters) based on input/output data. Our problem is slightly different as the term that has to be identified is the input variable itself (the so-called source term). This relates to specific research issues, typically considered from the signal processing point of view and referred to as *blind identification problems* (see for example [10], [11] and references therein). This designation alludes to the fact that we don't have information about the input that is being introduced in the system.

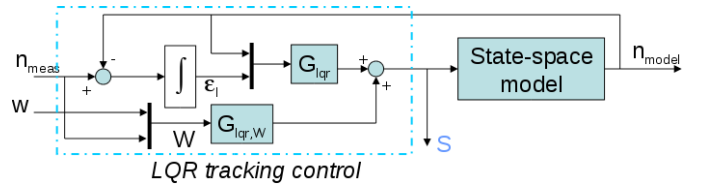


Fig. 2: Optimal tracking for blind identification.

An alternative to the signal processing approach is to formulate an optimal tracking problem based on the model and the measurements [12]. In this case, the model is the system which dynamics has to be controlled, $S(t)$ acts as the controlled input and the measurement $n_{e,meas}(t)$ is the tracked reference. The control objective is thus to minimize the modeling error $\epsilon(t) = n_{e,meas}(t) - n_{e,model}(t)$ over the experiment time horizon. It can be achieved for example using a LQR tracking controller with integral action. This specific architecture is presented in Figure 2. The related control law is detailed in the next section.

III. OPTIMAL SOURCE IDENTIFICATION

The aim of this section is to design an optimal approach to the source identification, considering a given transport model and a density measurements data set. The proposed optimization method is inferred from the variational approach to optimal control problems, such as the one described in [13].

A. Extended state and cost function

An integral action is first introduced in the feedback architecture by extending the state with a new variable corresponding to the integral of the estimation error $\varepsilon(t)$. The estimation error is the difference between the reference signal r (e.g. the measured plasma density) and the model output $y = x$ (e.g. the modeled plasma density). The extended state writes as:

$$\underbrace{\frac{d}{dt} \begin{bmatrix} x \\ \varepsilon_I \end{bmatrix}}_{\dot{X}} = \underbrace{\begin{bmatrix} A & 0 \\ -I & 0 \end{bmatrix}}_{A_e} \underbrace{\begin{bmatrix} x \\ \varepsilon_I \end{bmatrix}}_X + \underbrace{\begin{bmatrix} B \\ 0 \end{bmatrix}}_{B_e} S + \underbrace{\begin{bmatrix} w \\ r \end{bmatrix}}_W \quad (3)$$

Our aim is to optimize the estimated source such that ε should be close to zero with a minimum weighted input effort. The cost function that is to be minimized is consequently:

$$J = \frac{1}{2} \int_0^{t_f} \begin{bmatrix} x \\ \varepsilon_I \end{bmatrix}^T \begin{bmatrix} 0 & 0 \\ 0 & Q_r \end{bmatrix} \begin{bmatrix} x \\ \varepsilon_I \end{bmatrix} dt + S^T R S dt \quad (4)$$

where $Q_r = Q_r^T \geq 0$ weights the integrated error and $R = R^T > 0$ the input usage.

B. Optimal tracking with exogenous inputs

The Hamiltonian associated with (3)-(4) writes as:

$$H = \frac{1}{2} (\varepsilon_I^T Q_r \varepsilon_I + S^T R S) + \lambda^T (A_e X + B_e S + W) \quad (5)$$

where λ denotes the Lagrange multipliers (adjoint state). The optimal input is given by the following necessary condition

$$\frac{\partial H}{\partial S} = RS + B_e^T \lambda = 0 \Leftrightarrow S = -R^{-1} B_e^T \lambda \quad (6)$$

The adjoint vector $\lambda(t)$ is solution of the differential equation:

$$\dot{\lambda} = -\frac{\partial H}{\partial X} = -A_e^T \lambda(t) - Q_r X \quad (7)$$

It can be shown (considering the extended system composed of X and λ) that a general solution to the previous equation writes as:

$$\lambda(t) = P(t)X(t) - \gamma(t) \quad (8)$$

where $P(t)$ and $\gamma(t)$ are real variables of appropriate dimensions. Expressing (7) in terms of the previous solution implies:

$$\dot{\lambda}(t) = (-A_e^T P - Q_r)X + A_e^T \gamma \quad (9)$$

Introducing (6) and (8) into (3) gives the optimal state trajectory:

$$\begin{aligned} \dot{X} &= A_e X - B_e R^{-1} B_e^T \lambda + W \\ &= A_e X - B_e R^{-1} B_e^T P X + B_e R^{-1} B_e^T \gamma + W \end{aligned} \quad (10)$$

The differential equation (7) can now be written as

$$\begin{aligned} \dot{\lambda}(t) &= \dot{P}(t)X(t) + P(t)\dot{X}(t) - \dot{\gamma}(t) \\ &= (\dot{P} + PA_e - PB_e R^{-1} B_e^T P)X + (PB_e R^{-1} B_e^T) \gamma \\ &\quad + PW - \dot{\gamma} \end{aligned} \quad (11)$$

Equating (9) and (11) provides:

$$\begin{aligned} \dot{P} &= PB_e R^{-1} B_e^T P + Q_r - PA_e - A_e^T P \\ \dot{\gamma} &= PB_e R^{-1} B_e^T \gamma + PW - A_e^T \gamma \end{aligned} \quad (12)$$

Which allows for the computation of the optimal source given by (6). The boundary of the previous dynamics are provided by the terminal constraints $P(t_f) = 0$ and $\gamma(t_f) = 0$ (as no terminal cost is considered in J).

Supposing that γ and P reach equilibrium with limited variations (infinite horizon solution), a quasi-steady state solution can be used for simplified computations with:

$$\begin{aligned} Q_r &= PA_e + A_e^T P - PB_e R^{-1} B_e^T P \\ \gamma(t) &= -[PB_e R^{-1} B_e^T - A_e^T]PW(t) \end{aligned} \quad (13)$$

The optimal state trajectory is then given by:

$$\dot{X}(t) = A_e X - B_e R^{-1} B_e^T P X + B_e R^{-1} B_e^T \gamma(t) \quad (14)$$

and the corresponding optimal input is:

$$u(t) = -R^{-1} B_e^T [PX(t) - \gamma(t)] \quad (15)$$

with P and γ provided either by (12) or (13). Simulations tests comparing the use of the dynamic or the quasi-steady-state solutions for P and γ have shown that both were equivalent. The quasi-steady-state solution is preferred and kept in the final design, as it is simpler to implement. The optimal input thus has a feedback (Riccati equation) plus feedforward (to include time-variations of the boundary conditions) architecture. The controller gains presented on Figure 2 are thus:

$$G_{lqr} = -R^{-1} B_e^T P, \quad G_{lqr,W} = -R^{-1} B_e^T [PB_e R^{-1} B_e^T - A_e^T]P$$

C. Structure verification and Q_r and R tuning

To test the performance of the structure, a verification test was carried out with physical values for the transport parameters. Inserting an hypothetical Gaussian particle source $S(t)$ in the model, a theoretical density distribution $x(t)$ was obtained. This distribution was introduced in the identification structure and the Gaussian source was retrieved. The identification efficiency is evaluated thanks to the normalized accumulated tracking error:

$$\bar{e}(t) = \Delta z \sum_{i=0}^N e_i(t), \text{ with } e(t) = \left| \frac{x_{measured}(t) - x_{model}(t)}{x_{model}(t)} \right|$$

where e_i is the i^{th} component of e . When the weighting matrices for the LQR are chosen as $Q_r = R = I$, it was noticed that the estimation error, although small, was always persistent. Since an integral action had been added to the system, the steady state error is expected to disappear relatively quickly. As the plasma/wave coupling phenomenon is quasi-instantaneous (in comparison with the density time constant), the estimation settling time needs to be much faster than the

system dynamics, which was achieved by setting $Q_r = 10^7 \times I$ and $R = 10^{-7} \times I$. The associated simulation results can be found in [9].

D. Application to real data shot TS38953

The adequacy of the proposed method to model experimental results is investigated on Tore Supra density measurements from shot TS38953. In this shot, the LH antenna was being constantly switched on and off. The boundary conditions are first supposed to be constant with $n_{ez0}(t) = n_{e,L}(t) \equiv 0$ (this hypothesis will be released in the next section). Figure 3 presents the measured density and the accumulated tracking error. The identified source term can be observed in Figure 4a.

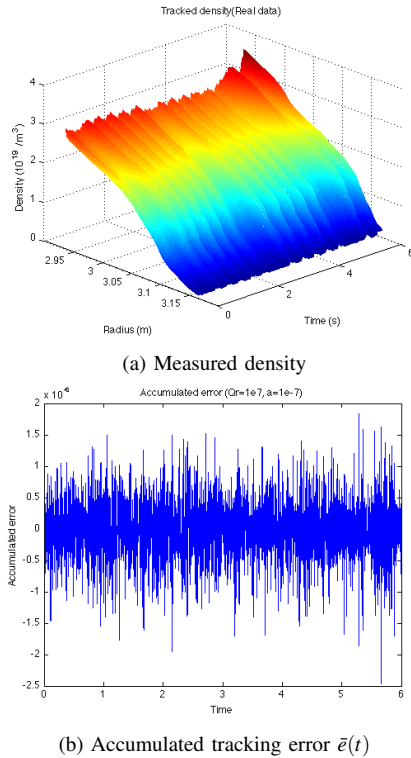


Fig. 3: Measured density and accumulated tracking error from TS38953.

As expected, two sources appear in the SOL, one due to the limiter (LPT) and the other one due to the LH antenna. The source term can thus be written as:

$$S(z, t) = S_{LPT}(z, t) + S_{LH}(z, t)$$

where $S_{LPT}(z, t)$ is larger than $S_{LH}(z, t)$. Also, a plasma displacement when the LH power is on can be observed (non-uniform normalized plasma radius with respect to the sensors frame).

While the low accumulated error is satisfactory, three main problems appear:

- 1) the source term reaches negative values, which cannot happen in the real source term;

- 2) positive values reached by the source are much higher than it was expected (increased by 10^2);
- 3) edge distortion appears in the source due to the constant boundary conditions that had been chosen in the initial design.

Specific strategies (such as anti-windup or saturations) can be used to solve the positivity problem. Nevertheless a revision of the transport model is preferred, in order to provide a feedback on the physical phenomena understanding. The three main problems are discussed in the following sections.

E. Time varying boundary conditions

To solve the edge source distortion problem, time varying boundary conditions were added and taken into account with the feedforward term $\gamma(t)$. The new identified source term is presented on Figure 4b: the edges are no longer distorted. On the other hand, the inner part of the source distribution doesn't suffer substantial changes and the order of magnitude for the density accumulated tracking error is maintained as 10^{-6} .

F. Application to real data shot TS45525

Due to the abnormal amplitude obtained with data shot TS38953, we decided to try the structure with another data shot obtained from Tore Supra, the shot TS45525. This shot, unlike the first one, was obtained using constant LH heating power. This allows for decoupling the electron density transport phenomena from the complex plasma-wave interactions and the plasma shape variations. The related identified source profile is presented on Figure 5.

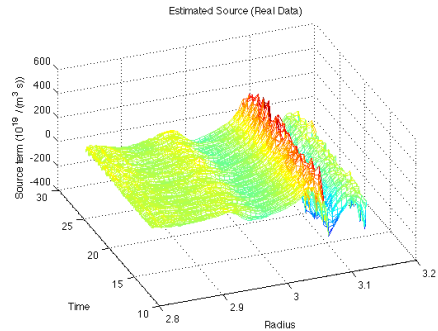


Fig. 5: Unconstrained identified source for TS45525.

It can be observed that, even if negative values keep appearing in the source term, the order of magnitude of the sources is around the expected one. Further studies will consequently be needed to identify the cause of the high source term obtained when the LH power changes abruptly, eventually including a refined model to depict the transients associated with the plasma-wave interactions.

In the new source term, a third peak can be observed between the LPT and LH sources. This third source can be caused by the antenna protection limiter (LPA) that was not taken off during TS45525 (unlike in TS38953, where it was removed especially for that experience). In fact, the

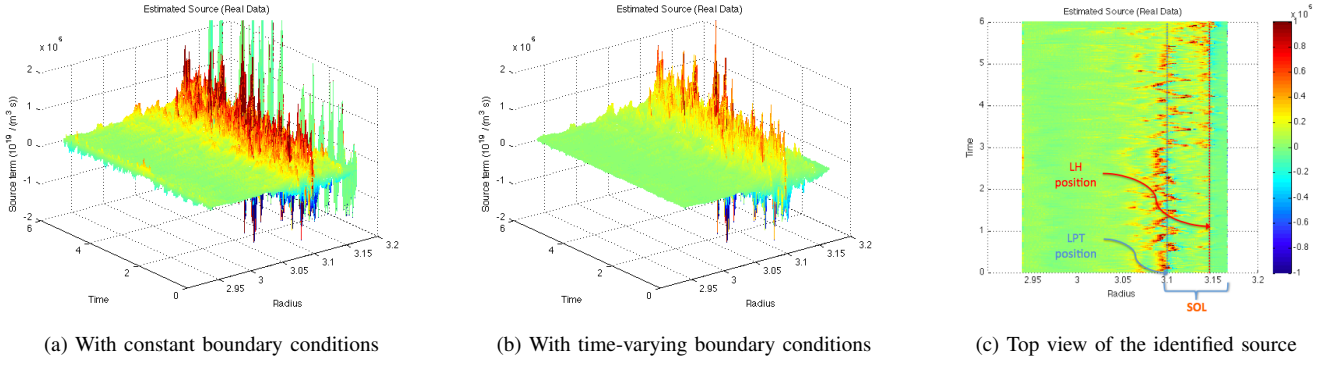


Fig. 4: Identified source term for TS38953.

position of the plasma facing components during this shot were: LPT at $R = 3.086\text{m}$, LPA at $R = 3.116\text{m}$ and LH at $R = 3.153\text{m}$. These values correspond approximately with the source peak positions observed on the identified term. We can thus conclude that the new obtained source is more coherent with the expected values. Nevertheless, the source term in the confined plasma always presents positive values (though small), while it should be zero in this region. This problem will be solved by a multi-zone approach in Section V.

IV. CONSTRAINED OPTIMAL IDENTIFICATION

With the aim of avoiding negative values in the source term, a constrained identification structure was designed so that a determined shape could be imposed to the input source term. In this section, the deduced constrained system and the constrained equilibrium source are first introduced. Then, the designed constrained identification structure and the numerical issues observed during its application to the real system are exposed.

A. Constrained system and equilibrium Gaussian source term

From [5], the assumption of a SOL source term composed by two Gaussian distributions (one due to the LPT and the other one due to the LH antenna) was taken. The main idea consists in changing the arbitrary distributed input source for a Gaussian shaped distributed source parameterized by the amplitude $\theta(t)$, the mean $\mu(t)$ and the standard deviation $\sigma(t)$ of the two Gaussian distributions:

$$S(z, t) = \theta_{LPT} e^{-(z - \mu_{LPT})^2 / (2\sigma_{LPT}^2)} + \theta_{LH} e^{-(z - \mu_{LH})^2 / (2\sigma_{LH}^2)}$$

The associated constrained discretized input is consequently set with:

$$S_i(t) = \sum_{j=\{LPT, LH\}} \theta_j(t) e^{-(z_i - \mu_j(t))^2 / (2\sigma_j(t)^2)}$$

The unconstrained source input is thus replaced by a nonlinear (Gaussian) function of the parameters:

$$\vartheta(t) = [\theta_{LPT} \ \mu_{LPT} \ \sigma_{LPT} \ \theta_{LH} \ \mu_{LH} \ \sigma_{LH}]^T$$

In order to include the nonlinearity in ϑ in the proposed optimization framework, it has to be linearized (first order

approximation) around an equilibrium point (denoted with the subscript eq).

The parameters ϑ_{eq} for the Gaussian equilibrium source are computed such that the averaged error between the obtained density and the measured density is minimized. The corresponding data set $\{x_{eq}, w_{eq}\}$ is selected to correspond to time instants when LH is on and the profiles are stabilized, and the source parameters are obtained by solving the optimization problem:

$$\vartheta_{eq} = \arg \min_{\vartheta} \{Ax_{eq} + \nabla_{\vartheta} S|_{\vartheta_{eq}} \vartheta + w_{eq}\}$$

where $\nabla_{\vartheta} S|_{\vartheta_{eq}}$ denotes the partial derivative of S with respect to ϑ (Jacobian operator), evaluated at ϑ_{eq} . The obtained equilibrium source and the equilibrium density are presented in Figure 6. This source represents the optimal constrained (Gaussian like) input for the system in order to obtain the equilibrium density, taken as the mean density.

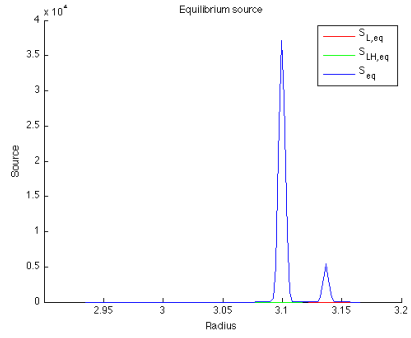
B. Numerical difficulties

Once the system was linearized around the equilibrium source, a constrained identification structure was designed to obtain the Gaussian-like source term varying in time. This structure would retrieve the incremental Gaussian parameters (the input to the linearized constrained system) and calculate the corresponding Gaussian shaped source.

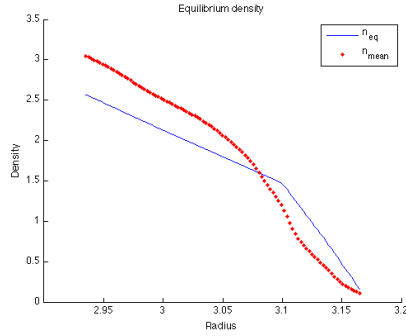
Unfortunately, although the performance of this identification structure was verified with more simple systems, when the system under study (the constrained linearized input system) was implemented, numerical problems took place and the solution for the LQR optimal controller could not be found. After studying this situation, we arrived to the conclusion that the difficulty lied in solving the Riccati equation. To solve this sort of equations, we used the *Matlab*[®] function *care* (the one used to obtain the optimal controller in the unconstrained structure), which computes the eigenvalues of the Hamiltonian matrix:

$$\mathcal{H} = \begin{pmatrix} A_e & -B_e R^{-1} B_e^T \\ -Q_r & -A_e^T \end{pmatrix}$$

For the constrained case, *Matlab*[®] justifies that the eigenvalues of this matrix are too near to the imaginary axis, and



(a) Equilibrium Gaussian source



(b) Corresponding equilibrium density and mean density

Fig. 6: Optimal equilibrium Gaussian source and density.

thus, it cannot find the solution to the Riccati equation. This problem mainly comes from the fact that A_e is not full rank and can be corrected, for example, by adding a forgetting factor on the integrated error (negative term in the lower right part of A_e). This has been successfully implemented in [14].

V. A MULTIZONE APPROACH

The previous sections have been focused in achieving the most accurate SOL identification structure in order to retrieve a source term with an expected shape. This section is focused on revising the SOL model with radial anisotropy. A new SOL model based on three different regions is thus proposed.

A. Compartmental description with a 3-zones model

The limitations highlighted in the previous sections (higher source values during transients and negative identified terms) motivate the need for a more complex and accurate transport model of the SOL. Indeed, [15] suggests that an advection model may be more appropriate for the last part of the SOL. In addition, for the plasma placed before the separatrix (e.g. confined plasma), there would be no sink term since the magnetic field lines in confined plasma are closed on themselves. The edge plasma region is thus split into three different subregions (see Figure 7): the confined plasma, first part of SOL and final part of SOL. We consequently have the following transport models:

- Diffusion model (in the confined plasma region):

$$n_{e,t} = D_{\perp} n_{e,zz} + S(z,t) \quad (16)$$

- Diffusion model with sink term (in the first part of non confined plasma region or SOL):

$$n_{e,t} = D_{\perp} n_{e,zz} - v n_e(z,t) + S(z,t) \quad (17)$$

- Advection model with sink term (in the second part of the SOL):

$$n_{e,t} + V_c n_{e,z} = -v n_e(z,t) + S(z,t) \quad (18)$$

with continuity conditions (Neumann boundaries) between each zone. The specific computation of the transport parameters in each zone can be found in [9].

To integrate these three models in the same state-space representation, the following generic description is used:

$$\begin{cases} n_{e,t} = D(z) n_{e,zz} + V(z) n_{e,z} + R(z) n_e + S(z,t), \\ n_{e,z}(0,t) = n_{e,z0}(t), \quad n_e(1,t) = n_{e,L}(t), \\ n_e(z,0) = n_{e,t0}(z), \end{cases} \quad (19)$$

where D , V and R are evaluated at each location z_i and set according to the descriptions (16)-(18). To delimit the different regions, two points are defined:

- $\mathbf{n}_a$ marks the end of confined plasma and thus the beginning of the SOL. It is chosen around the separatrix or LCMS (the position of the LCMS can be retrieved from the TS database for each shot);
- $\mathbf{n}_b$ marks the beginning of the advection model in the SOL (around 4 cm after the LCMS).

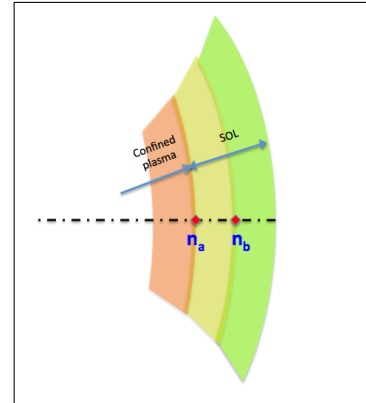


Fig. 7: Subregions in edge plasma.

B. Parameters computation and obtained source term

The space parameters are set according to the zones definition as depicted in Figures 8a-8b for the data shot TS45525. The related algorithm was conceived to allow space-varying parameters for the models, which is a more realistic situation than the constant-parameter case. Figure 8c presents the obtained source term (almost constant as the LH power input was constant for TS45525).

Comparing these results with the ones obtained with the first estimation structure (see Figure 5), it can be observed that the negative values have been reduced and the values for the three sources that keep appearing are more consistent with

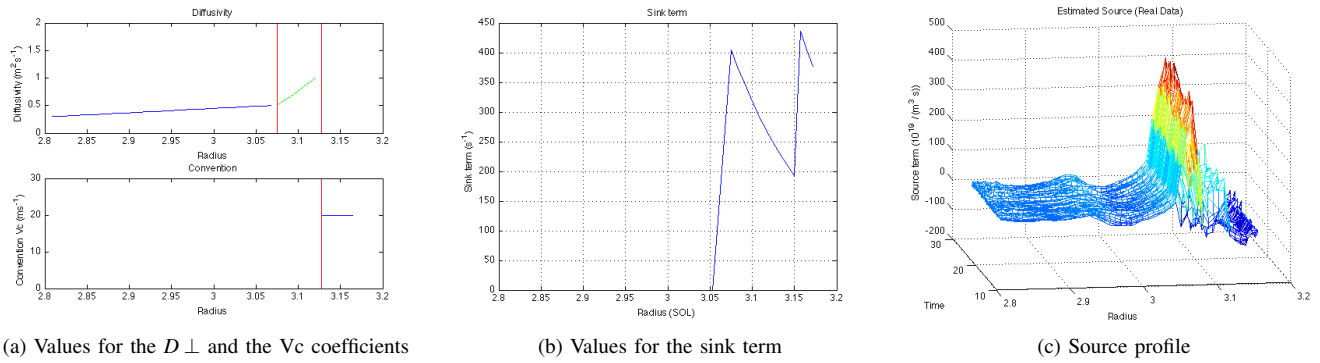


Fig. 8: Calculated parameters and resulting source term for TS45525 using the 3 zones model.

the expected values. Also, it can be seen that the source term in confined plasma is almost zero. The new model thus appears to be more accurate in terms of the qualitative representation of the expected physical phenomena and of the expected source amplitude.

It is important to notice that the obtained source term is very sensitive to variations in the model parameters, which hints towards future research focused on the estimation robustness and the estimation of time-varying transport parameters.

CONCLUSIONS

In this paper, we have developed an optimal feedback approach to infer a time-varying input for a system model when the output is measured. While this structure has been developed for the SOL model in tokamaks, it can easily be extrapolated to other systems characterized by (possibly anisotropic) transport phenomena. This identification method makes possible the use of constraints on the input or models with multiple interconnected regions. Experimental results have shown the effectiveness and limitations of the proposed approach.

Future research will focus on the specificity associated with the identification of sources with fast varying parameters and combined source/transport parameters estimation.

ACKNOWLEDGMENTS

The authors are grateful to the anonymous reviewers, who provided numerous insights for the paper improvement and to motivate further research on this topic. This work, supported by the European Communities under the contract of Association between EURATOM and CEA, was carried out within the framework of the European Fusion Development Agreement. The views and opinions expressed herein do not necessarily reflect those of the European Commission. The Tore Supra LHCD team is acknowledged for operating the LHCD system.

REFERENCES

- [1] A. Ekedahl, M. Goniche, D. Guilhem, F. Kazarian, Y. Peysson, and Tore Supra Team, "Lower hybrid current drive in tore supra," *Fusion Science and Technology*, vol. 56, no. 3, pp. 1150–1172, Oct. 2009.
- [2] G. Hoang *et al.*, "A lower hybrid current drive system for iter," *Nuclear Fusion*, vol. 49, no. 7-075001, 2009.

- [3] J. Mailloux *et al.*, "Long distance coupling of the lower hybrid waves on Tore Supra," in *Proc. of the 25th Eur. conf. on Cont. Fus. and Plasma heating*, 1998.
- [4] A. Ekedahl *et al.*, "Long distance coupling of lower hybrid waves in JET plasmas with edge and core transport barriers," *Nucl. Fusion*, 2005.
- [5] E. Witrant, M. Goniche, and Equipe TORE SUPRA, "A qss approach for particle source identification in tore supra tokamak," in *Proc. of 48th IEEE Conference on Decision and Control*, Shanghai, Dec. 2009.
- [6] M. Kočan, J. Gunn, J.-Y. Pascal, G. Bonhomme, C. Fenzi, E. Gauthier, and J.-L. Segui, "Edge ion-to-electron temperature ratio in the Tore Supra tokamak," *Plasma Phys. Control. Fusion*, vol. 50, pp. 125 009–125 019, 2008.
- [7] A. Pazy, *Semigroups of Linear Operators and Applications to Partial Differential Equations*, ser. Applied mathematical sciences. New York: Springer Verlag, 1983, vol. 44.
- [8] G. Allaire, *Numerical Analysis and Optimization: An Introduction to Mathematical Modelling and Numerical Simulation*, 9th ed. New York: Oxford Univ. Press, 2007.
- [9] E. Santiago, "Particle source identification in the plasma edge in tore supra tokamak," Master's thesis, Université de Grenoble, 2010.
- [10] S. Achard, D. Pham, and C. Jutten, "Criteria based on mutual information minimization for blind source separation in post nonlinear mixtures," *Signal Processing*, vol. 85, no. 5, pp. 965–974, 2005, special issue on Information Theoretic Signal Processing.
- [11] Y. Yu and A. Petropulu, "Parafac-based blind estimation of possibly underdetermined convolutive mimo systems," *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 111–124, 2008.
- [12] J. Weydert, "Source determination in non-homogeneous dissipative media," Master's thesis, University Joseph Fourier, 2009.
- [13] D. Kirk, *Optimal Control Theory: An Introduction*, ser. Electrical engineering. Englewood Cliffs, New Jersey: Prentice-Hall, 1970.
- [14] F. Bribiesca Argomedo, E. Witrant, C. Prieur, D. Georges, and S. Brémond, "Model-based control of the magnetic flux profile in a tokamak plasma," in *Proc. of 49th IEEE Conference on Decision and Control*, Atlanta, USA, Dec. 2010.
- [15] X. Garbet, "Plasma edge physics," Association Euratom-CEA, CEA/DSM/DRFC, CEA-Cadarache, France, lecture notes for the master Sciences de la Fusion, Oct. 2008.

Connectivity properties of the Apache Ant class collaboration network

Miloš Savić, Mirjana Ivanović, and Miloš Radovanović

Abstract—Properties of networks whose nodes represent classes of an object-oriented (OO) system, and links denote dependencies between them, are fundamental to understand, describe and characterize software complexity at the structural level. In this paper, degree distributions of the class collaboration network that represents class dependencies of Apache Ant software were examined. In contrast to similar studies where complementary cumulative degree distributions were tested only against a power law, here we also consider the exponential distribution. Our analysis revealed scale-free phenomena in in-degree and total degree distributions. However, Ant's out-degree sequence is better modeled with the exponential distribution than a power law. Based on this result, implications relevant to software engineering are discussed in the aspect of the absence of a characteristic scale of class reuse and the presence of a characteristic scale of class aggregation.

I. INTRODUCTION

In the past decade, a huge body of research has investigated statistical properties of networks that represent various complex biological, social, technological, and conceptual systems. Those analyses had revealed some common properties of real-world networks such as the existence of nodes with extremely high degree compared to the average (hubs), heterogeneous connectivity structure characterized by a power law (the scale-free property), small distance between two randomly chosen nodes (the small-world property), and highly dense sub-graphs induced by a randomly chosen node and adjacent nodes (high level of clustering) [1–3]. Recent research [4–10] suggests that networks formed from large and widely used software systems share characteristics common to many analyzed real-world networks.

A large program written in an object-oriented (OO) programming language is typically divided into a set of classes. Classes defined in an OO software system can be viewed as nodes, and dependencies between them as links in a *class collaboration network* (CCN). The links in a CCN are directional, two classes A and B are connected by the directed link $A \rightarrow B$ if class A references class B . Class collaboration networks can be viewed as simplified class diagrams, a notion extensively used in object-oriented design, because they preserve only relations between classes and discard other types of information about nodes (classes) and links (relations) presented in class diagrams.

Modern OO software systems are collections of many hundreds or even thousands of classes. Traditionally, the complexity of OO software systems is viewed as the internal complexity of classes defined in some OO program and estimated using software metrics such as the lines of code (LOC) metric or cyclomatic complexity. However, there is another level of complexity in OO systems – structural complexity of dependencies between classes. To illustrate the differences between these two layers of complexity, we will give two simple examples. Let us suppose that we need to implement an OO system which has to realize some complex requirements specification. Such an OO system can be implemented by only one class, and the internal complexity of that class will be enormously high, but the complexity of the appropriate dependency structure will be the simplest possible network (a network that contains just one node and no links), which means that the structural complexity of the system is minimized. On the other side, good decomposition of functionality can lead to a set of classes whose internal complexity is minimized, but the appropriate network which reflects the structure of the system will be enormously complex (comparing to the previous case, when the structural complexity is minimized), and not easy to comprehend.

The following simple example shows that reducing complexity at one level increases complexity at the other level. Let us suppose that we have an OO system with three defined classes A , B and C , where class A references classes B and C , and we will suppose that classes B and C have a relatively large portion of identical code. In such a situation, a refactoring of the system can be done: duplicated code can be isolated into a new class D which will be referenced by classes B and C . After the refactoring internal complexity of the classes is reduced (both average and total LOC are smaller), but the structure of dependencies becomes more complex, because the refactoring process introduced one more node, two more links and the change in class collaboration network topology (before the process, the class collaboration network was a tree, and after it is a network which contains an undirected loop).

Statistical properties of class collaboration networks can help us to understand and quantify the organizational structure of corresponding software. The degree distribution is a widely used measure of network complexity [1–3]. The degree of a node is the number of links connected to it, and degree distribution $P(k)$ indicates the fraction of nodes that have exactly k links. One of the main advantages of the degree distribution is its discriminative power. Static random networks (Erdős-Rényi random graphs [12] and

Manuscript received April 10, 2011.

M. Savić is with the Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia (corresponding author; phone: +381-21-458888; fax: +381-21-6350458; e-mail: milsav@gmail.com)

M. Ivanović and M. Radovanović are with the Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg D. Obradovića 4, 21000 Novi Sad, Serbia (e-mail: {mira, radacha}@dmi.rs).

Watts-Strogatz small-world networks [13]) have bell-curve shaped degree distributions, evolving random networks with the uniform attachment principle are characterized by the exponential distribution ($P(k) \sim \exp(-\beta k)$), while degree distributions of evolving random networks with the preferential attachment principle follow the power law ($P(k) \sim k^{-\gamma}$) [14]. For static random networks, the set of nodes is definitely formed before any connection is established. In contrast, evolving random networks grow over time by adding new nodes in each evolutionary step. The attachment principle determines how newly created nodes are integrated into the network. In the case of the uniform attachment principle, each node has equal probability to establish a connection with a newly added node, while the preferential attachment principle is based on the idea that the connection probability depends on node degree (nodes with higher degree have a higher probability to be connected with the newly created node).

In a directed network, the degree of a node is the sum of its in-degree (number of links entering the node) and out-degree (number of links exiting the node), thus there are three degree distributions that describe the connectivity of a network: total degree distribution (or just degree distribution) $P(k)$, in-degree distribution $P_{in}(k)$, and out-degree distribution $P_{out}(k)$, where $P_{in}(k)$ and $P_{out}(k)$ indicate fractions of nodes that have exactly k in-coming and out-going links, respectively.

In this work, degree distributions of the CCN associated with the Apache Ant software system are examined. The network was extracted from the source code of Ant version 1.7.0 using the Yaccne tool (detailed description of this software can be found in [10]). In Section II, motivation for this study and its contributions are highlighted. A representative selection of related works, which view software systems as complex networks is reviewed in Section III. The following section describes the results obtained in the analysis of the degree distribution of Ant's CCN. In Section V, the results are discussed from the software engineering perspective. Finally, in Section VI we give the conclusion.

II. MOTIVATION AND CONTRIBUTIONS

In our previous study [10], we investigated complementary cumulative degree distributions of five class collaboration networks formed from software systems written in Java, among them Ant. Compared to [10], in this study two improvements are made. Firstly, degree distributions are calculated for the largest weakly connected component. Small isolated clusters of trivial complexity which increase $P(k)$ for small values of k are omitted from analysis. In this case, observed degree distributions describe the connectivity of the largest fraction of connected nodes, which is the substructure that carries complexity. Secondly, the exponential distribution, characteristic of evolving random networks that employ the uniform attachment principle, is considered as a candidate to model in, out and total degree sequences. In related work [4–11], which examined degree distributions of various types of collaboration networks formed

from software systems, degree distributions were tested only against a power law. Software systems evolve over time and it is natural to suppose that software networks have a topology which can be modeled by evolving random networks. However, a power law cannot characterize all types of evolving random networks, only those based on the preferential attachment (or similar) principle. In this work, as the main result which provides completely new insights related to software engineering, we will show that Ant's complementary cumulative out-degree distribution cannot be considered scale-free. In more detail, this means the following:

- 1) Ant's complementary cumulative out-degree distribution has a power-law scaling region which does not cover the whole range of out-degree values (as was already reported in [10]),
- 2) The complementary cumulative distribution function of the exponential distribution better models the observed out-degree sequence through the whole range of out-degree values than a power law over a power-law scaling region.

In [10], CCNs extracted from ten successive versions of Ant (from version 1.5.2 to version 1.7.0) were compared, and it was shown that the preferential attachment principle can explain changes through Ant software evolution at the class collaboration level for in-coming links. This result is significant because it suggests that the preferential attachment principle is relevant to software evolution (and this relevance was also shown in [11] using the Linux kernel as a case study). This result also explains how Ant's complementary cumulative in-degree distribution evolved into a scale-free state. Although in [10] it was shown that Ant's complementary cumulative out-degree distribution reveals a region that follows a power law, we were unable to obtain evidence of preferential attachment through evolutionary changes of out degrees. Exponential scaling of Ant's complementary cumulative out-degree distribution explains why: this type of scaling does not characterize the preferential attachment, but it does signify the uniform attachment principle.

III. RELATED WORK

Valverde et al. [4] analyzed the JDK class collaboration network extracted from the class diagram. They found that two largest connected components in the JDK network are scale-free (with gamma exponents 2.5 and 2.65, respectively). However, in their analysis link directionality was ignored.

The same authors [6] analyzed the JDK class collaboration network, this time represented as a directed graph, again extracted from the class diagrams. They obtained a power-law in unnormalized complementary cumulative in-degree and out-degree distributions for the largest weakly connected component and the second largest weakly connected component. They also analyzed giant weakly connected components contained in collaboration networks extracted from the source code of 18 C/C++ applications. For all networks they obtained that in/out complementary cumulative degree

distributions follow power-law with γ_{in} in range 1.94 to 2.55 and γ_{out} in range 2.41 to 3.39.

Myers [5] examined collaboration networks associated with three open source object-oriented (OO) systems written in C++ (VTK, DM, AbiWord) and three written in C (Linux, MySQL, XMMS). He computed the unnormalized complementary cumulative degree distributions for in-coming and out-going links and found that these distributions reveal a power-law scaling region. Values of γ_{in} are in range [1.9, 2.5], of γ_{out} in range [2.4, 3.33] and $\gamma_{out} \geq \gamma_{in}$ for each analyzed network. Myers also developed a simple model of software evolution based on two refactoring techniques (decomposition of large functions into a set of smaller ones and removal of duplicated code).

De Moura et al. [7] investigated collaboration networks of four open source software projects (Linux, XFree86, Mozilla and Gimp) extracted from source code by parsing C/C++ header files. They omitted link directionality and analyzed collaboration networks as undirected graphs. They computed unnormalized degree distributions and found scale-free, small-world and larger than random graph clustering coefficients.

Potantin et al. [8] analyzed run-time object graphs (dynamic, run-time analogues of static collaboration networks) of several OO systems (Java ArgoUML, Java BlueJ, Java Forte, Java Jinsight, Java Satin, C++ GCC, Self and Smalltalk). Their research confirmed power laws in the in-degree and out-degree distributions.

Hyland-Wood et al. [9] analyzed class collaboration networks of two open source software projects written in Java for a fifteen-month period of development and produced collaboration graphs at package, class and method levels. The collaboration graphs were found to form networks which exhibited approximately scale-free properties at all three levels during each analyzed period.

Common to all mentioned works is that degree distributions were tested only against a power-law.

IV. RESULTS

Connectivity properties of a directed complex network can be summarized by in, out and total degree distributions, denoted as $P_{in}(k)$, $P_{out}(k)$ and $P(k)$, respectively. Degree distributions of random growing networks have some nice properties which make them easy to identify. If $P(k)$ (the same holds for $P_{in}(k)$ and $P_{out}(k)$) follows the power law $P(k) \sim k^{-\gamma}$ then the plot of $P(k)$ on logarithmic scales appears as a straight line with slope $-\gamma$ ($\log P(k) \sim -\gamma \log k$). This gives us a simple way to determine if a degree distribution follows the power law. However, it is better to examine the complementary cumulative distribution (CCD) instead of the degree distribution, because the CCD reduces noise that may appear in the tail of empirically observed degree distributions, and gives more accurate estimations of the scaling exponent γ [15]. Complementary cumulative degree distribution $CCD(k)$ is the probability of finding a node with degree greater than or equal to k , that is, $CCD(k) = \sum_{i=k}^{\infty} P(i)$.

If $P(k) \sim k^{-\gamma}$ then $CCD(k) \sim k^{-(\gamma-1)}$ [2], therefore the plot of $CCD(k)$ on log-log scales also appears as a straight line with slope $-(\gamma-1)$. In the case that $P(k)$ is exponentially distributed, $P(k) \sim \exp(-\beta k)$, then $CCD(k)$ is exponential with the same exponent β , $CCD(k) \sim \exp(-\beta k)$ [2]. When plotted on semi-logarithmic scales both functions appear as straight lines with slope $-\beta$ (if $f(k) \sim \exp(-\beta k)$ then $\log f(k) \sim -\beta k$). This makes power-law and exponential degree distributions particularly easy to spot by plotting the corresponding complementary cumulative degree distributions on logarithmic scales (for power laws) or semi-logarithmic scales (for exponentials). The described plotting techniques are used in the analysis of observed complementary cumulative degree distributions associated with Ant CCN.

We computed complementary cumulative in, out and total degree distributions for the largest weakly connected component of Ant CCN (97.81% of nodes) and tested them against the power law of the form ax^γ and complementary cumulative distribution function (CCDF) of the exponential distribution (CCDF of the exponential $\text{Exp}(\beta)$ distribution is of the form $e^{-\beta x}$). Coefficient of determination (R^2) was used as a goodness of fit measure. Figures 1, 2, 3 show in, out and total CCD for the Ant class collaboration network with fitted power law and CCDF of the exponential distribution. It can be seen that the power law nearly ideally fits in and total CCDs. Estimated values for scaling exponents are $\gamma_{in} = 2.05 \pm 0.01$ and $\gamma_{tot} = 2.51 \pm 0.02$ (slopes of the fitted lines are -1.05 and -1.51 , respectively). In contrast, CCDF of the exponential distribution better fits out-CCD than a power law. Although out-CCD in the tail (for out-degree > 5) has a power-law scaling region followed by faster decay (for out-degree > 21), the fit of exponential CCDF, which covers the whole range of out-degree values, has a greater R^2 value ($R^2(\text{power-law fit}) = 0.99181$, $R^2(\text{exponential fit}) = 0.99407$). Also, it can be visually seen that on semi-log scales out-CCD appears as a straight line with small fluctuations (Figure 4).

V. IMPLICATIONS RELEVANT TO SOFTWARE ENGINEERING

Software engineering practice and simple software metrics such as LOC can give us evidence that modern software systems are complex artifacts. Traditional software metrics used for estimating software complexity are mainly oriented to software entities (functions, classes, methods). These metrics view software as a collection of isolated entities and derive overall software complexity as an average or cumulative complexity of the entities. Also, they can be used to identify complex entities in order to refactor them into smaller, less complex entities. Although this process reduces complexity of entities, it increases complexity at a higher, structural level: new entities and new dependencies between entities appear. Recent progress in the theory of complex networks enables us to measure and characterize software complexity at the structural level, where software

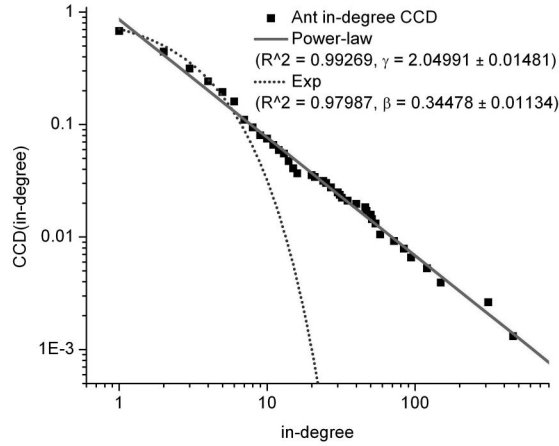


Fig. 1. In-CCD for the Ant CCN with fitted power law of the form ax^γ (line) and exponential of the form $e^{-\beta x}$ (dots)

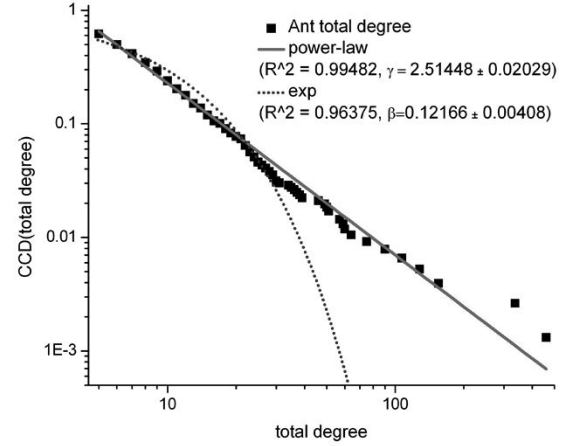


Fig. 3. Total CCD for the Ant CCN with fitted power law of the form ax^γ (line) and exponential of the form $e^{-\beta x}$ (dots)

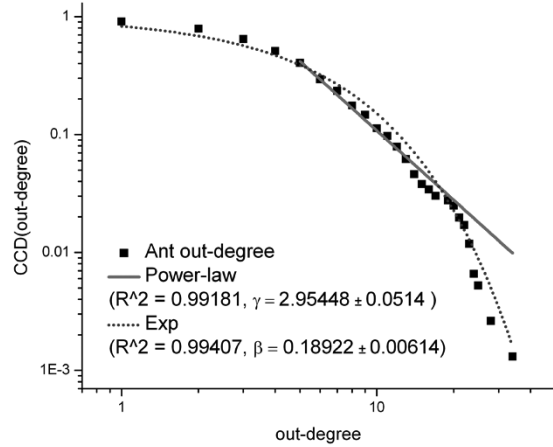


Fig. 2. Out-CCD for the Ant CCN plotted on logarithmic scales with fitted power law of the form ax^γ (line) and exponential of the form $e^{-\beta x}$ (dots)

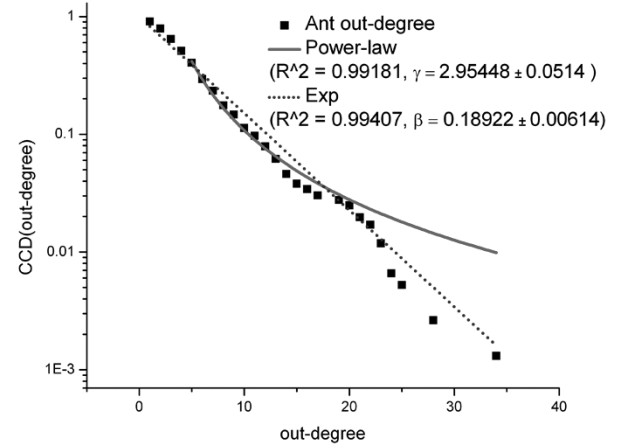


Fig. 4. Out-CCD plotted on semi-logarithmic scales with fitted power law of the form ax^γ (line) and exponential of the form $e^{-\beta x}$ (dots)

is not seen as a set of isolated, but rather of interconnected software entities.

In-coming degree of a node in a class collaboration network reflects the degree of reuse of a class (or interface) represented by the node. The existence of a power law in-degree distribution implies a broad spectrum of code reuse: most of the classes present in Apache Ant (65.59%) are not reused, but there is a statistically significant number of highly reused classes (2.23%), whose degree of reuse is far above average. For example, two most reused classes “BuildException” and “Project” have 459 and 313 in-coming links, respectively, and average in-degree is 4.768. Preferential attachment, a generating mechanism for power-law degree distributions and empirically verified in Ant for incoming links [10], suggests the tendency of increasing reuse for already highly reused classes as the class collaboration network grows through software development. Software en-

gineering practice encourages code reuse and this tendency may seem very desirable. However, it is very difficult to change highly reused classes because of their importance to the stability of the software system. Changes in a highly reused class may affect a large number of classes which directly or indirectly depend on it. Thus, identifying highly reused software entities, especially ones which do not realize simple functionalities, may help to prevent the following conflicted situation: the presence of highly reused, hard-to-maintain classes with the tendency of increasing reuse which makes them even more difficult to maintain.

Our findings obtained though the conducted experiments suggest that out-degree complementary cumulative distribution of Apache Ant is better modeled with exponential complementary cumulative distribution functions than a power law. Power law for out-going links was reported in studies which analyzed software networks without ignoring

link directionalities [5, 6, 9, 10], but where complementary cumulative distributions were tested only against a power law. Out-going degree of a node in a CCN reflects the degree of aggregation of the class represented by the node. As noted by Myers [5], classes that have high out-degrees are intra-complex because they aggregate behavior from many other classes. To confirm this thesis we investigated how internal complexity of Ant classes estimated using traditional software metrics is correlated with out-degrees of appropriate nodes in the Ant class collaboration network. Figure 5 shows average WMC (weighted method per class, which is the sum of cyclomatic complexity of methods contained in a class) as a function of out-degree for Ant classes. A trend of increasing intra-complexity (estimated using WMC) can be noticed as out-degree increases. The same trend appears when the intra-complexity of classes is estimated using the LOC (lines of code) metric (see Figure 6). In contrast to the in-degree distribution, the absence of a heavy-tailed out-degree distribution implies the absence of a broad spectrum of intra-complexities. This means that software systems exhibit characteristic intra-complexity (average out-degree) which can describe well the overall intra-complexity of the software system.

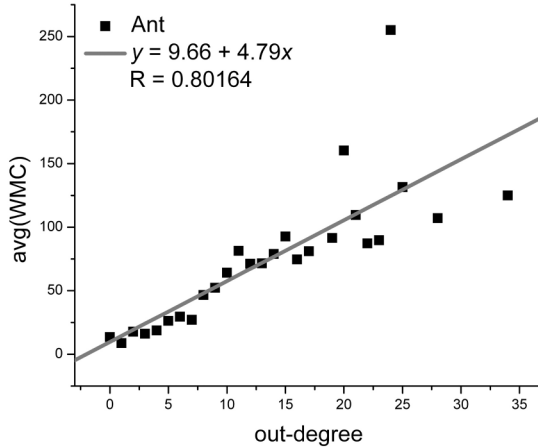


Fig. 5. Average WMC as s function of out-degree for Ant classes

Tables I and II show top ten highest in-degree and out-degree nodes present in Ant class collaboration network, respectively. As can be seen class “Project” appears in both tables, which means that this class exhibits a high degree of reuse and aggregation at the same time. It has both significant internal complexity (due to aggregating the behavior of several other classes) and significant external responsibility (because it is used in a lot of other classes defined in the Ant software project). It is interesting to observe that in-degree and out-degree measures are very similar to the information-flow metrics defined by Henry and Kafura [16]: fan-in (the number of functions calling a given function) and fan-out (the number of functions being called from a given function). Actually, in-degree and out-degree

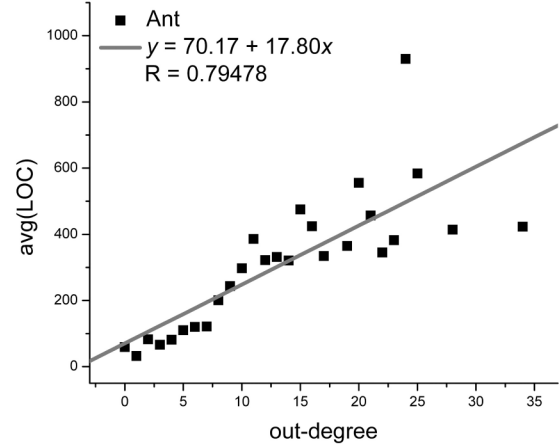


Fig. 6. Average LOC as function of out-degree for Ant classes

TABLE I
TOP TEN HIGHEST IN-DEGREE NODES IN ANT

CLASS	IN	OUT	LOC	C_p
BuildException	459	1	69	14536989
Project	312	24	930	52145233920
Task	149	6	175	139866300
FileUtils	120	8	618	569548800
Path	94	13	374	558488216
Commandline	84	6	298	75696768
Resource	72	3	186	8678016
Reference	58	2	44	592064
FileSet	54	3	34	892296
DirectoryScanner	54	10	758	221032800

are fan-in and fan-out analogues at the class collaboration level. Henry and Kafura [16] defined a complexity metric C_p calculated as $C_p = LOC * (fan-in * fan-out)^2$, where LOC represents the number of lines of code in a function. According to Zimmermann et al. [17], functions with a large value of Henry-Kafura complexity may indicate poor design and such entities are candidates for refactoring. If Henry-Kafura complexity for classes is defined as $C_p = LOC * (in-degree * out-degree)^2$, then class “Project” has the highest Henry-Kafura complexity and its C_p value is at least two orders of magnitude higher than others (C_p values for the top ten in-degree/out-degree nodes are shown in Tables I and II).

Power laws in in-degree distributions suggest a trend of increasing reuse of highly reused nodes, and therefore good predictions can be made about classes that will be reused by classes added as software evolves. In contrast, the uniform attachment principle, a generating mechanism for the exponential degree distribution observed in out-going links, suggests that prediction about aggregation cannot be made: statistically speaking, each class has the same apriori probability to aggregate newly added classes. In other words, based only on the topology of a CCN, reuse of classes can be predicted, but aggregation can not.

TABLE II
TOP TEN HIGHEST OUT-DEGREE NODES IN ANT

CLASS	IN	OUT	LOC	C_p
Delete	3	34	423	4400892
AbstractFileSet	6	28	414	11684736
MatchingTask	25	25	171	66796875
Zip	1	25	996	622500
Project	312	24	930	52145233920
ResourceUtils	6	23	351	6684444
BaseSelectorContainer	5	23	116	1534100
AbstractSelectorContainer	2	23	106	224296
JUnitTask	1	23	956	505724
ConditionBase	7	22	100	2371600

VI. CONCLUSION

This paper investigated complementary cumulative degree distributions of the largest weakly connected component of Ant's class collaboration network. Distributions were tested against a power law, characteristic of scale-free networks, and the complementary cumulative distribution function of the exponential distribution, characteristic of evolving random networks with the uniform attachment principle. It was shown that Ant's network exhibited the scale-free property for in and total degrees, but out-degree sequence can be better modeled with the exponential distribution. On the basis of this, implications relevant to software engineering were discussed in the following aspects: importance of determining nodes with large in-degree that correspond to highly reused classes, characteristic scale of intra-complexities among classes, predictability of class reuse, and unpredictability of class aggregation.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support of this work by the Serbian Ministry of Education and Science through project *Intelligent Techniques and Their Integration into Wide-Spectrum Decision Support*, no. OI174023.

REFERENCES

- [1] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks", *Rev. Mod. Phys.*, vol. 74, no. 1, pp. 47–97, 2002.
- [2] M.E.J. Newman, "The structure and function of complex networks", *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.
- [3] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez and D.-U. Hwang, "Complex networks: Structure and dynamics", *Phys. Rep.*, vol. 424, pp. 175–308, 2006.
- [4] S. Valverde, R. Ferrer Cancho and V. Solé, "Scale-free networks from optimal design", *Europhys. Lett.*, vol. 60, pp. 512–517, 2002.
- [5] C.-R. Myers, "Software systems as complex networks: Structure, function and evolvability of software collaboration graphs", *Phys. Rev. E*, vol. 68, no. 4, pp. 046116, 2003.
- [6] S. Valverde and V. Solé, "Hierarchical small-worlds in software architecture", *Dyn. Contin. Discret. Impuls. Syst. Ser. B: Appl. Algorithms*, 14(S6), pp. 1–11, 2007.
- [7] A.P. de Moura, Y.C. Lai and A.E. Motter, "Signatures of small-world and scale-free properties in large computer programs", *Phys. Rev. E*, vol. 68, no. 1, pp. 017102, 2003.
- [8] A. Potanin, J. Noble, M. Frean and R. Biddle, "Scale-free geometry in object-oriented programs", *Commun. ACM*, vol. 48, no. 5, pp. 99–103, 2005.
- [9] D. Hyland-Wood, D. Carrington and S. Kaplan, "Scale-free nature of Java software package, class and method collaboration graphs", *Technical report no. TR-MS1286, MIND Laboratory, University of Maryland College Park*, 2006.

- [10] M. Savić, M. Ivanović and M. Radovanović, "Characteristics of class collaboration networks in large Java Software Projects", *Inf. Technol. Control*, vol. 40, no. 1, pp. 48–58, 2011.
- [11] L. Wang, Z. Wang, C. Yang, L. Zhang and Q. Ye, "Linux kernels as complex networks: A novel method to study evolution", *Proc. 25th IEEE Int. Conf. on Software Maintenance (ICSM)*, Edmonton, AB, Canada, September 20–26, 2009.
- [12] P. Erdős and A. Rényi, "On random graphs I", *Publ. Math. Debrecen*, vol. 5, pp. 290–297, 1959.
- [13] D. Watts and S. Strogatz, "Collective dynamics of 'small-world' networks", *Nature*, vol. 393, pp. 440–442, 1998.
- [14] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks", *Science*, vol. 286, pp. 509–512, 1999.
- [15] M.E.J. Newman, "Power-laws, Pareto distribution and Zipf's law", *Contemp. Phys.*, vol. 46, no. 5, pp. 323–351, 2005.
- [16] S.M. Henry and D.G. Kafura, "Software structure metrics based on information flow", *IEEE T. Software Eng.*, vol. SE-7, no. 5, pp. 510–518, 1981.
- [17] T. Zimmermann, N. Nagappan, A. Zeller, "Predicting bugs from history", In T. Mens, S. Demeyer (Eds.), *Software Evolution*, pp. 69–88, Springer, 2008.

Model Predictive Control of a Biotechnological Process

Dorin Sendrescu, *Member, IEEE*, Eugen Bobasu, *Member, IEEE*

Abstract— This paper deals with the design of a nonlinear model predictive control (NMPC) scheme for the regulation of the acetate concentration in a biomethanation process – wastewater biodegradation with production of methane gas that takes place inside a Continuous Stirred Tank Bioreactor. The NMPC control structure is based on the nonlinear model of the process whose parameters are identified using the distributions method. Minimization of the cost function is realized using the Levenberg–Marquardt numerical optimization method. Some simulation results are given to illustrate the efficiency of the proposed control strategy.

I. INTRODUCTION

DURING the last years, the control of biotechnological processes has been an important problem attracting wide attention. The main engineering motivation in applying control methods to such living processes is to improve operational stability and production efficiency. But the use of modern control for these bioprocesses is still low. Nonlinear model predictive control (NMPC) is needed especially for nonlinear, unsteady batch processes where a trajectory needs to be followed from the prediction of a nonlinear model. It is especially useful for processes operating at or near singular points that cannot be captured by linear controllers and where higher order information is needed. NMPC uses the nonlinear dynamic model to predict the effect of sequences of control steps on the controlled variables.

The dynamics of the wastewater biodegradation process are described by a set of nonlinear differential equations obtained from the reaction scheme. For the estimation of unknown parameters of the process, the distribution approach is used. The parameter identification of deterministic nonlinear continuous-time systems (NCTS), modeled by polynomial type differential equation has been considered by numerous authors [1], [2]. In this paper, we use an identification method for a class of NCTS considering that the unknown parameters can appear in rational relations with measured variables. Using

techniques utilized in distribution approach, the measurable functions and their derivatives are represented by functionals on a fundamental space of testing functions. Such systems are common in biotechnology [3]. The main idea is to use a hierarchical structure of identification. First, some state equations are utilized to obtain a set of linear equations in some parameters. The results of this first stage of identification are utilized for expressing other parameters by linear equations. This process is repeated until all parameters are identified.

The paper is organized as follows. Section II is devoted to description and modeling of an anaerobic digestion bioprocess. The nonlinear model predictive control strategy is presented in Section III. Simulations results presented in Section IV illustrate the performance of the proposed control algorithms and, finally, Section V concludes the paper.

II. PROCESS MODELING

A. Analytical approach of process modeling

In the conventional MPC controller, a linear predictive model is used because the theory of the identification of a linear system has well been established. The nonlinear part of the system response is treated as disturbance. But, a linear model, no matter how well has it been structured and tuned, may be acceptable only in the case where the system is working around the operating point. If the system is highly nonlinear, such as biotechnological processes, control based on the prediction from a linear model may result in unacceptable response. In some cases, remarkable static errors exist, and in other cases, oscillation or even instability may occur. Therefore, some kinds of non-linear models should be used to describe the behaviour of a highly non-linear system.

In this paper one considers a biomethanation process – wastewater biodegradation with production of methane gas that takes place inside a Continuous Stirred Tank Bioreactor whose reduced model is presented in [4]. It is a two phase process. In the first phase, the glucose from the wastewater is decomposed in fat volatile acids (acetates, propionic acid), hydrogen and inorganic carbon under action of the acidogenic bacteria. In the second phase, the ionised hydrogen decomposes the propionic acid $\text{CH}_3\text{CH}_2\text{COOH}$ in acetates, H_2 and carbon dioxide CO_2 . In the first methanogenic phase, the acetate is transformed into methane and CO_2 , and finally in the second methanogenic

Manuscript received June 18, 2011. This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project 1D61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007—2013.

D. Sendrescu is with the Automatic Control Department, University of Craiova, A.I. Cuza, 200585, Romania (e-mail: dorins@automation.ucv.ro).

E. Bobasu is with the Automatic Control Department, University of Craiova, A.I. Cuza, 200585, Romania (e-mail: ebobasu@automation.ucv.ro).

phase, the methane gas CH_4 is obtained from H_2 and CO_2 , [5]. The following simplified reaction scheme is considered,



where: S_1 represents the glucose substrate, S_2 the acetate substrate, X_1 is the acidogenic bacteria, X_2 the acetoclastic methanogenic bacteria and P_1 represents the product, i.e. the methane gas. The reaction rates are denoted by Φ_1, Φ_2 . The corresponding dynamical model is:

$$\frac{d}{dt} \begin{bmatrix} X_1 \\ S_1 \\ X_2 \\ S_2 \\ P_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -k_1 & 0 \\ 0 & 1 \\ k_2 & -k_3 \\ 0 & k_4 \end{bmatrix} \begin{bmatrix} \Phi_1 \\ \Phi_2 \end{bmatrix} - D \begin{bmatrix} X_1 \\ S_1 \\ X_2 \\ S_2 \\ P_1 \end{bmatrix} + \begin{bmatrix} 0 \\ DS_{in} \\ 0 \\ 0 \\ -Q_1 \end{bmatrix} \quad (2)$$

where S_{in} is the influent substrate, Q_1 the methane gas outflow rate, D the dilution rate and the state vector of the model is:

$$\xi = [X_1 \ S_1 \ X_2 \ S_2 \ P_1]^T = [\xi_1 \ \xi_2 \ \xi_3 \ \xi_4 \ \xi_5]^T \quad (3)$$

whose components are concentrations in (g/l). The reaction rates are nonlinear functions of the state components, expressed as

$$\Phi = \Phi(\xi) = \begin{bmatrix} \Phi_1(\xi) \\ \Phi_2(\xi) \end{bmatrix} \quad (4)$$

The reaction rates for this process are given by the Monod law

$$\Phi_1(\xi) = \mu_1 \frac{S_1 \cdot X_1}{K_{M_1} + S_1} \quad (5)$$

and the Haldane kinetic model:

$$\Phi_2(\xi) = \mu_2 \frac{S_2 \cdot X_2}{K_{M_2} + S_2 + S_2^2 / K_i} \quad (6)$$

where K_{M_1}, K_{M_2} are Michaelis-Menten constants, μ_1, μ_2 represent specific growth rates coefficients and K_i is the inhibition constant.

For simplicity, shall we denote the plant parameters by the vector:

$$\theta = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4 \ \theta_5 \ \theta_6 \ \theta_7 \ \theta_8 \ \theta_9]^T \quad (7)$$

where:

$$\begin{aligned} \theta_1 &= k_1; \ \theta_2 = k_2; \ \theta_3 = k_3; \ \theta_4 = k_4; \ \theta_5 = \mu_1; \\ \theta_6 &= \mu_2; \ \theta_7 = K_{M_1}; \ \theta_8 = K_{M_2}; \ \theta_9 = K_i \end{aligned} \quad (8)$$

Because the dilution rate D can be externally modified, it

will be considered the third component of the input vector

$$u = [u_1 \ u_2 \ u_3]^T \quad (9)$$

The other two components of u are the concentration S_{in} and the methane gas outflow rate Q_1 so,

$$u_1 = S_{in}; u_2 = Q_1; u_3 = D \quad (10)$$

Usually Q_1 depends on state variables, $Q_1 = \psi(\xi)$, determining a feedback to the input u_2 . Written explicitly by components, the state equation (2), within the above notations, takes the form:

$$\dot{\xi}_1 = \Phi_1 - u_3 \cdot \xi_1 \quad (11)$$

$$\Phi_1 = \theta_5 \cdot \frac{\xi_1 \cdot \xi_2}{\theta_7 + \xi_2} \quad (12)$$

$$\dot{\xi}_2 = -\theta_1 \cdot \Phi_1 - u_3 \cdot \xi_2 + u_1 \cdot u_3 \quad (13)$$

$$\dot{\xi}_3 = \Phi_2 - u_3 \cdot \xi_3 \quad (14)$$

$$\Phi_2 = \theta_6 \cdot \frac{\xi_3 \cdot \xi_4}{\theta_8 + \xi_4 + \theta_9 \cdot \xi_4^2}, \theta_9' = \frac{1}{\theta_9} \quad (15)$$

$$\dot{\xi}_4 = \theta_2 \cdot \Phi_1 - \theta_3 \cdot \Phi_2 - u_3 \cdot \xi_4 \quad (16)$$

$$\dot{\xi}_5 = -u_3 \cdot \xi_5 + \theta_4 \cdot \Phi_2 - u_2 \quad (17)$$

B. Parameters estimation

For parameters estimation the distribution based method was used. In this approach the set of nonlinear differential equations describing the state evolution is mapped into a set of linear algebraic equations respect to the model parameters. Using techniques utilized in distribution approach, the measurable functions and their derivatives are represented by functionals on a fundamental space of testing functions. The main advantages of this method are that a set of algebraic equation with real coefficients results and the formulations are free from boundary conditions.

Let Φ_n be the fundamental space from the distribution theory of the real functions $\varphi: R \rightarrow R, t \rightarrow \varphi(t)$ and $q: R \rightarrow R, t \rightarrow q(t)$ a function which admits a Riemann integral on any compact interval T from $\mathbf{R}$. Using this function, a unique distribution

$$F_q: \Phi_n \rightarrow R, \varphi \rightarrow F_q(\varphi) \in R \quad (18)$$

can be built by the relation:

$$F_q(\varphi) = \int_R q(t) \varphi(t) dt, \forall \varphi \in \Phi_n \quad (19)$$

In distribution theory, the notion of k-order derivative is introduced. If $F_q \in \Phi_n$, then its k-order derivative is a new

distribution $F_q^{(k)} \in \Phi_n$ uniquely defined by the relations:

$$F_q^{(k)}(\varphi) = (-1)^k F_q(\varphi^{(k)}), \forall \varphi \in \Phi_n \quad (20)$$

$$\varphi \rightarrow F_q^{(k)}(\varphi) = (-1)^k \int_R q(t) \varphi^{(k)}(t) dt \in R \quad (21)$$

where

$$\varphi^{(k)} : R \rightarrow R, t \rightarrow \varphi^{(k)}(t) = \frac{d^k \varphi(t)}{dt^k} \quad (22)$$

is the k-order time derivative of the testing function.

When $q \in C^k(R)$, then

$$F_q^{(k)}(\varphi) = \int_R q^{(k)}(t) \varphi(t) dt = (-1)^k \int_R q(t) \varphi^{(k)}(t) dt, \quad (23)$$

that means the k-order derivative of a distribution generated by a function $q \in C^k(R)$ equals to the distribution generated by the k-order time derivative of the function q .

So, in place of the states and their time derivatives of a system one utilizes the corresponding distributions and, in some particular cases, it is possible to obtain a system of equations linear in parameters. If the system is compatible then all the model parameters are structurally identifiable.

Consider all state variables accessible for measurements. The dynamical system (11)-(17) contains rational dependencies between parameters and measured variables. To obtain linear equations in unknown parameters, the identification problem is split in several simpler problems. Based on the specific structure of this system, it is possible to group the state equations, in such way to determine five interconnected relations. They are organized in a hierarchical structure. In the first stage some state equations are utilized to obtain a set of linear equations in some parameters. If these parameters are identified then they can be used as known parameters in the following stages. This process is repeated in the other stages until all the parameters are identified.

III. NONLINEAR MODEL PREDICTIVE CONTROL

Consider the following discrete-time, time-invariant nonlinear system:

$$\xi_{k+1} = f(\xi_k, u_k) \quad (24)$$

with ξ_k the state vector, u_k the control signal, corresponding in our case with the discretisation of system (2). The objective is to regulate the state vector (or a particular output signal) to a specified setpoint value while guaranteeing that certain input and state constraints:

$$\begin{aligned} \xi_{\min} &\leq \xi_k \leq \xi_{\max} \\ u_{\min} &\leq u_k \leq u_{\max} \end{aligned} \quad (25)$$

Nonlinear model predictive control treats such a constrained control problem by repeatedly solving the

following optimization problem:

$$\min \sum_{i=0}^{N-1} (\xi_{ref} - \xi_{k+i})^T Q (\xi_{ref} - \xi_{k+i}) + u_{k+i}^T R u_{k+i} \quad (26)$$

$$s.t. \begin{cases} \xi_{k+1} = f(\xi_k, u_k) \\ \xi_{\min} \leq \xi_k \leq \xi_{\max} \\ u_{\min} \leq u_k \leq u_{\max} \end{cases} \quad (27)$$

Among the whole sequence resulting from the on-line optimization, only the first optimal control is applied as input to the system [6]. At the next sampling time, the current state is obtained (measured or estimated) and the optimization problem (26), (27) is solved again with this new initial state value, according to the well-known receding horizon principle.

The block diagram of NMPC based control system is presented in Fig.1.

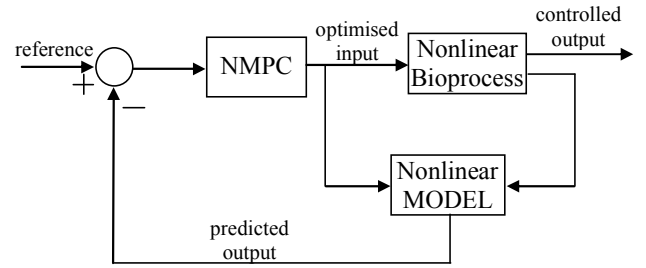


Fig. 1. Block diagram of NMPC based control system

The previous general NMPC formulation is applied to the biomethanation process, in order to regulate acetate concentration S_2 to a reference value S_{2ref} by manipulating the dilution rate D . Based on the idea of the NMPC cost function, without any particular terminal constraints and considering output feedback control, the formulation of the optimization problem is given by:

$$\begin{aligned} \min \sum_{i=1}^N (S_{2ref} - S_{2k+i})^2 + \lambda \sum_{i=1}^N D_{k+i}^2 \\ s.t. D, \xi_k \geq 0. \end{aligned} \quad (28)$$

The λ value determines the contribution that the sum of the squares of the control increments has on the performance index. To solve this optimization problem the Levenberg-Marquardt (LM) algorithm was used. LM algorithm is an iterative technique that locates the minimum of a multivariate function that is expressed as the sum of squares of non-linear real-valued functions. It has become a standard technique for non-linear least-squares problems, widely adopted in a broad spectrum of disciplines. LM can be thought of as a combination of steepest descent and the Gauss-Newton method [7].

When the current solution is far from the correct one, the algorithm behaves like a steepest descent method: slow, but

guaranteed to converge. When the current solution is close to the correct solution, it becomes a Gauss-Newton method [8], [9]. The basis of the LM algorithm is a linear approximation of f in the neighbourhood of parameter vector p . A Taylor series expansion leads to the approximation:

$$f(p + \delta_p) \approx f(p) + J\delta_p \quad (29)$$

where J is the Jacobian matrix $\frac{\partial f(p)}{\partial p}$. Like all non-linear optimization methods, LM is iterative: initiated at the starting point p_0 , the method produces a series of vectors $p_1, p_2, \dots$ that converge towards a local minimizer p_{min} for f . Hence, at each step, it is required to find the δ_p that minimizes the quantity $\|x - f(p + \delta_p)\| \approx \|x - f(p) - J\delta_p\| = \|\varepsilon - J\delta_p\|$. The parameter δ_p is thus the solution to a linear least-square problem: the minimum is attained when $J\delta_p - \varepsilon$ is orthogonal to the column space of J . This leads to $J^T(J\delta_p - \varepsilon) = 0$ which yields δ_p as the solution of the so-called normal equations:

$$J^T J \delta_p = J^T \varepsilon. \quad (30)$$

The matrix $J^T J$ is the approximate Hessian. The LM algorithm terminates when the magnitude of the gradient drops below a threshold or a maximum number of iterations is completed.

IV. SIMULATION RESULTS

Simulation was performed for the following bioprocess parameters:

$$\theta = [5.4 \quad 1 \quad 14.7 \quad 10 \quad 0.2 \quad 0.6 \quad 0.75 \quad 1 \quad 21]^T$$

where:

$$\begin{aligned} \theta_1 &= k_1; \quad \theta_2 = k_2; \quad \theta_3 = k_3; \quad \theta_4 = k_4; \quad \theta_5 = \mu_1; \\ \theta_6 &= \mu_2; \theta_7 = K_{M_1}; \theta_8 = K_{M_2}; \theta_9 = K_i \end{aligned}$$

Considering all these parameters unknown, they were identified using the procedure described in Section 2. Then the NMPC strategy from Section 3 was applied in order to regulate the acetate concentration to a desired reference trajectory. Saturation constraints in the manipulated variable are imposed to take into account that all concentrations must be positive.

The simulation results obtained using a sample period $T_s = 6$ minutes are presented in Fig. 2.

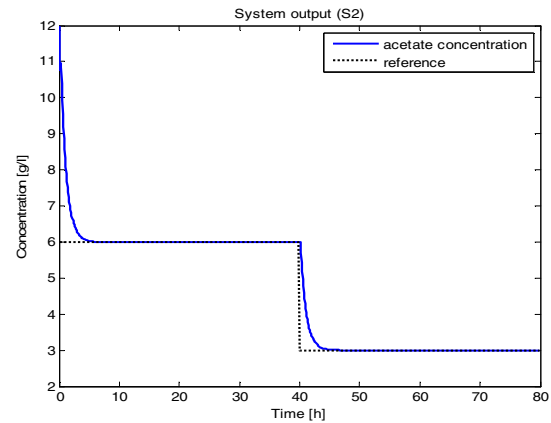


Fig. 2. Evolution of acetate concentration

V. CONCLUSIONS

In this paper a nonlinear model predictive control for a depollution bioprocess was presented. The nonlinear model was obtained using the analytical description of the chemical reactions. The unknown parameters of the model were estimated using the distribution approach. This method transforms the differential system of equations into an algebraic system linear irrespective to the unknown parameters. The nonlinear model states are used to calculate the optimal control signal applied to the system. The optimisation problem is solved using the iterative Levenberg-Marquardt algorithm. The obtained results are quite encouraging from a simulation viewpoint and show the robustness of the controller and good tracking precision. The computer simulations show that the NMPC strategy ensures a low level of the water pollution.

REFERENCES

- [1] A. Patra, H. Unbehauen, 1995. Identification of a class of nonlinear continuous time systems using Hartley modulating functions. *Int.J.Contr.* Vol.62, No.6,1431-1452.
- [2] A. Pearson, F. Lee, 1985. On the identification of Polynomial input-output differential systems. *IEEE Trans. Aut. Contr.*, Vol. AC30, No8,778-782.
- [3] G. Bastin, D. Dochain, *On-line Estimation and Adaptive Control of Bioreactors*, Elsevier, Amsterdam, 1990.
- [4] D. Dochain, P. Vanrolleghem, *Dynamical Modelling and Estimation in Wastewater Treatment Processes*. IWA Publishing, 2001.
- [5] E. Petre, D. Selisteanu, *Modelling and Identification of Depollution Bioprocesses*, Universitaria, Craiova, 2005.
- [6] B. Kouvaritakis and M. Cannon, *Nonlinear predictive control: Theory and practice*. IEE, 2001.
- [7] E.F Camacho and C. Bordons, *Model Predictive Control*, second edition, Springer, 2004.
- [8] J.W. Eaton and J.R. Rawlings, "Feedback Control of Nonlinear Processes Using Online Optimization Techniques," *Computers and Chemical Engineering*, vol. 14, pp. 469-479, 1990.
- [9] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer, New York, 1999.

Fluidization and Fluid views of Discrete Event Systems

Manuel Silva and Cristian Mahulea

Abstract—Fluidization is an efficient relaxation technique to tackle classical state explosion problems in discrete event systems (DES), which consists in approximating the discrete states by some continuous or hybrid ones. This is not a technical work in the most classical sense. More epistemological and methodological, the purpose is to overview fluidization in several well-known modeling paradigms for DES. As a central modeling paradigm for parallel and synchronized DES, Petri nets are considered first and the most used firing semantics for timed fluid PNs are briefly discussed. Fluid queuing networks and process algebras are related to fluid Petri nets. In the last part, three modeling formalisms that by definition use continuous states are recalled with a special interest on Forrester diagrams and their connection with fluid Petri nets. We assume from the reader a basic knowledge of Petri nets [1], [2].

I. INTRODUCTION

A mathematical optimization problem in which the variables are restricted to integer numbers is called an *integer programming problem*. Computationally, this is a NP-hard problem that can be solved iteratively by using cutting plane methods and/or branch and bound algorithms, for example [3]. If the objective function and the constraints are linear, the relaxation of variables into reals leads to *linear programming problems* of polynomial time complexity. This is a classical example of *fluidization* where the feasible set, initially countable, is approximated by a polyhedron. The application of this idea of relaxation to the state or fundamental equation of Petri nets was used for the first time in [4].

A dynamical system in which the *state* changes in accordance with the occurrence of events is called a *discrete event system*. All possible states of a DES compose its *state space* being a countable set, even if in many cases not bounded. Therefore, most of the “exact” methods for checking properties are “close” to integer programming problems. Moreover, DES may suffer from the well known *state explosion problem* especially when the system is highly populated. Transformation [5] and structural techniques [6], [7] (where the initial conditions play the role of a parameter) may be of interest, but they do not offer complete solutions for all cases. In order to tackle the problem in these last cases, exactly as in integer programming, the standard approximation technique called *fluidization* can be considered.

Fluidization techniques try to obtain semi-decision or approximate solutions for qualitative and quantitative analysis

of complex models, instead of exact solutions of simplified views of the original system [8], [9]. Fortunately, the quality of the approximation usually improves with the size of the population in the system being considered, while at the same time the computational savings with respect to the discrete underlying model becomes much more significant. Another advantage of the fluid approximation is that it allows some useful insights on the potential behavior of the real system under consideration.

Even if the system is not highly populated, some tractable approximated solutions can be obtained from the fluidized model. Anyhow it is particularly relevant to point out that, in essence, any model is just an approximation of a certain reality. Obviously, part of the price paid by fluidization is that certain properties cannot be considered in a fluid framework, e.g., mutual exclusion.

This paper provides an overview of fluidization applied to DES. The central formalism is Petri nets (PNs) for which we give some technical details on the analysis and designing methods presented in literature (see [10], [11], [12] for more technical details). Queuing networks and process algebras, other two modeling formalisms for DES, are considered at conceptual level. It is described how the fluidization has been applied in these cases and (possible) connections with fluid PNs. Sometime, the systems can be abstracted from the beginning with continuous (fluid) states. This happens in the case of compartmental systems, stochastic fluid systems and Forrester diagrams. Some elements of the comparison among them and the fluid PNs are given (more details on the comparison can be found in [12], in which a much broader perspective is provided).

Here, fluid or continuous Petri nets are not seen as isolated formalisms, but as part of a broad modeling paradigm for DES, the *Petri nets paradigm* [13], [14]. Based on the expression of *concurrency* and *synchronization*, locality on states (places) and actions (transitions) is a basic issue on Petri nets. One of its main consequences is the possibility to approach the design, analysis and implementation of parallel and synchronized, eventually distributed, systems using *bottom-up* (by composition of lower level modules) and *top-down* (by refinement of very abstract or upper level descriptions) approaches, in arbitrary interleaved manners. After that, PN models can be presented as *flat* or *structured* descriptions, in the last case just by keeping track of the way they are constructed, a degree of freedom of the modelers and users. Anyhow, structuring is an essential issue in order to deal with modeling and analysis of complex system.

The paper is organized as follow: Section II provides an introduction to fluid Petri nets while in Section III, Queuing

This work has been partially supported by the European Community's Seventh Framework Programme under project DISC (Grant Agreement n. INFOS-ICT-224498), by CICYT - FEDER grant DPI2010-20413 and by Fundación Aragón I+D.

The authors are with the Aragón Institute of Engineering Research (I3A), University of Zaragoza, Maria de Luna 1, 50018, Zaragoza, Spain {silva,cmahulea@unizar.es}.

Networks (QNs) and Process Algebras (PAs) are considered. In Section IV, other three formalisms are discussed. Finally, some concluding remarks are given in Section V.

II. FLUID PETRI NETS

Petri Nets have been introduced by C. A. Petri at the beginning of the 1960s [15] as a fully *non-deterministic* (untimed) conceptual framework to logically model and analyze *concurrency* and *synchronization* in DES. Notions of time in order to compute performance or dependability were added to PN's around two decades later, at the beginning of the 1980s. In the PN's paradigm time has been introduced for different purposes (for example, *time intervals* in order to deal with some qualitative or quantitative real-time properties [16]; or in a possibilistic fashion to handle uncertainty, or preferences, using *fuzzy sets* [17]). For performance evaluation and optimization, many different semantics have been also defined, providing the probability density functions (pdfs) for service times and defining some probabilistic routing (even under some fairness constraints). The most current practice is to assume random policies for queues (places), services (most frequently associated to transitions that become stations, where servers operate) and routing (at conflicts) [18]. For more elaborate disciplines, see [19].

The basic PN formalism contains only two simple and somehow orthogonal primitives: *transitions* and *places*. In some sense it can be said that PN theory has been developed in a *top-down* way, giving first especial attention to logic properties of fully non-deterministic models.

The state explosion problem appears frequently in DES, making the exhaustive state exploration methods, i.e., those based on the enumeration of all reachable states, inapplicable in many practical cases. Like in the evolution from *integer programming problems* to *linear programming problems*, one of the possible solutions to overcome it is to relax the description of the system by removing the integrality constraints. Applying this idea to discrete PN's, the firing of transitions is not limited to natural numbers but to the non-negative reals, leading to continuous PN (CPN) [1]. CPN have been used in many applications, e.g., manufacturing systems [20], [21], [22], traffic systems [23], supply chains [24], etc.

Fluidization of Petri nets offers the advantage of avoiding the state explosion problem, giving the chance of using computationally tractable algorithms. For technical details defining CPN's and an evolution of the approach, see [10], [11], as already mentioned. The set of reachable (and reachable at the limit, i.e., with infinitely long sequences, lim-reachable) markings in CPN's satisfies some properties that do not hold for discrete nets, e.g., it is a convex set [25]. Characterizations of the reachability space of an untimed CPN are given in [25], [26] while liveness and deadlock-freeness are considered in [25], [27].

A simple and interesting way to introduce time in discrete PN's is to assume that all the transitions are timed with random firing delay having exponential probability distribution

Clients	Servers	Semantics of the transition
Many	Many	infinite server semantics
Many	Few	finite server semantics
Few	Few	discrete transitions
Few	Many	discrete transitions

TABLE I

ON THE FLUIDIZATION OF A PN TRANSITION [11].

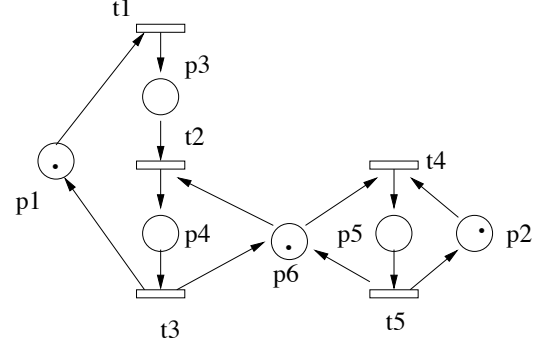


Fig. 1. CPN system used to illustrate the evolutions under infinite and finite server semantics. $\mathcal{N} = \langle P, T, \text{Pre}, \text{Post} \rangle$ is a CPN where P is the set of places, T is the set of transitions and Pre and Post are the incidence matrices defining the weights of arcs from places to transitions and from transitions to places, respectively. $\Sigma = \langle \mathcal{N}, \mathbf{m}_0 \rangle$ is a CPN system.

functions. Because this pdf has the Markov property, i.e., it is memory less, the state of the timed PN system is still given *only* by its marking (it is not necessary to keep any time information) [18]. For the timing interpretation of CPN's we will use a first order (or deterministic) approximation of the discrete case, assuming that the delays associated to the firing of transitions can be approximated by their mean values.

In general, transitions are interpreted as *stations* (in queuing network terminology), where *servers* and *clients* meet. Thus, the more appropriate firing relaxation depends on the relative number of servers and clients in the discrete model we want to approximate. Assuming, qualitatively speaking, that there may be “many” or “few” of each of them, fluidization can be considered for clients, for servers or for both. Table I represents the four theoretically possible cases. If the number of clients is “small” (Few-Few and Few-Many in Table I), the system is not really loaded, the transitions “should” remain discrete and the fluidization may be unsuitable. If there are many clients and a few servers (Many-Few) the relaxation is only at the level of clients, and *finite server semantics* [1] can provide a good approximation. On the other hand, in the case of many clients and many servers (Many-Many), a continuous model with *infinite server semantics* [11] seems a priori reasonable, since there are so many servers that there is no need to make them explicit.

It is important to notice that at conceptual level finite server semantics corresponds to a *hybrid* behavior: fluidization is applied only to clients, while servers are kept as discrete, counted as a finite number (used to define the bound of firing speed of the transition). On the other hand, infinite server semantics really relaxes clients and servers,

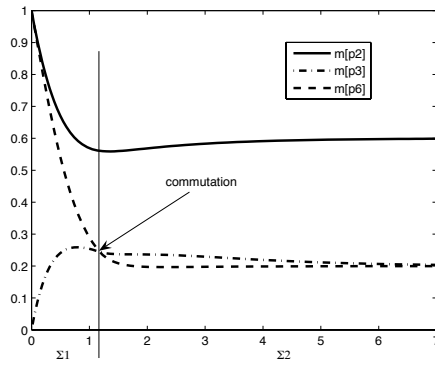


Fig. 2. Evolution of CPN in Fig. 1 with $\lambda = [1, 2, 1, 1, 0.5]^T$ under infinite server semantics.

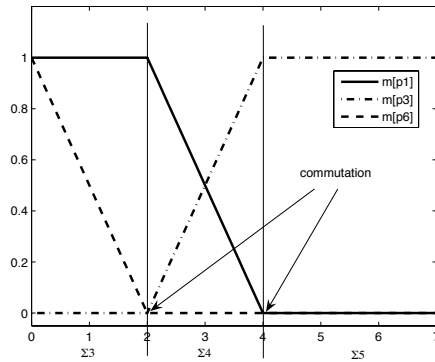


Fig. 3. Evolution of CPN in Fig. 1 with $\lambda = [1, 2, 1, 1, 0.5]^T$ under finite server semantics.

being the firing speed driven by the enabling degree of the transition, as for stochastic-markovian (discrete) PNs. The *enabling degree* of a transition t_j at a marking $\mathbf{m}$ is $enab(t_j, \mathbf{m}) = \min_{p_i \in \bullet t_j} \left\{ \frac{m_i}{Pre(p_i, t_j)} \right\}$, where $\bullet t_j$ denotes all input places of t_j .

For timed CPNs under infinite server semantics, the firing flow of a transition is proportional to the enabling degree that, if the corresponding transition has more than one input place, is based on a *minimum* function. Based on this definition of flow, the reachability set can be split in disjoint regions (except on the borders) such that the flow inside each region will be linear. Therefore, a CPN under this infinite server semantics is a linear time-invariant system provided with minimum operators. It can be proved that they can simulate Turing machines [28]. Corresponding to different abstraction views, different behaviors are obtained for the same model. Infinite server semantics corresponds to a set of *piecewise linear* differential equations while finite server semantics to a set of *piecewise constant* equations. For the same model (quite) different evolutions can be obtained under the same initial conditions. Let us consider the CPN system in fig. 1. Fig. 2 shows the evolution under infinite server semantics, while fig. 3 sketches the evolution if finite server semantics is considered.

In [1] (section 7.1.4) it is observed that, in general, infinite server semantics approximates better the marking of the underlying discrete PN than finite server semantics. In the case of mono-T-semiflow nets (nets with a unique repetitive behavior, i.e., unique solution to $(Post - Pre) \cdot x = 0, x > 0, x_1 = 1$) under a very general condition, it is proved in [29] that infinite server semantics always gives a better approximation of the steady-state throughput of the underlying Markovian discrete PN.

Being timed CPN under infinite server semantics a subclass of piecewise linear systems many results and design methods from Control Theory can be adapted. This firing semantic has been extensively considered in the PN literature during the last decade, with a special interest on problems of observability and control. The *only* control action that can be applied in a CPN is the reduction of the non-controlled (the maximum possible) firing flow. Notice that the flow of a transition represents in general the speed of a machine, hence it is upper bounded by the maximum speed. The controlled flow of a CPN is $w = f - u$. Since only the slowing down action is possible, the following constraints should be satisfied at every time moment: $0 \leq u \leq f$.

Controllability may be lost due to the token conservation laws, for example those related to P-semiflows. Additionally, it is shown that if all transitions are controllable, i.e., their flow can be reduced, the optimal steady state can be computed using a linear programming problem if the cost function is linear [30]. Moreover, in [31] it is proved that if all transitions are controllable, *consistency* (i.e., there exists $x \geq 1$ such that $(Post - Pre) \cdot x = 0$) is equivalent to controllability. The transient control problem is also considered by means of an implicit and explicit MPC control strategy [32]. A Lyapunov-function-based dynamic control algorithm was previously proposed in [33], which can ensure global convergence of both system states and input signals. A relaxation of this last approach based on linear programming instead of nonlinear programming has been proposed in [34].

With respect to observability, three different concepts have been defined based on the knowledge of the firing rate vector. The classical observability problem of linear systems is to estimate the initial state/marking, measuring only a subset of states, assuming a given (known and constant) value for the firing vector. In this case, the set of differential equations is fixed and the concept is called *observability in infinitesimal time*. Other observability concepts are *generic* and *structural* [35]. Observability criteria of piecewise affine systems can be applied to timed CPN since it is a subclass of those systems. It is well known that the observability in this case is a more difficult problem than the one of linear systems because not only the observability of continuous states is required but also of the discrete states [35]. It should be always possible to say which is the linear system governing the evolution.

Fluid PNs are usually considered as relaxations of original discrete models. Nevertheless, the dynamic behavior of a timed CPN model does not always approximate that of the corresponding timed discrete PN. Then, it is important to investigate the conditions that lead to a valid relaxation, from

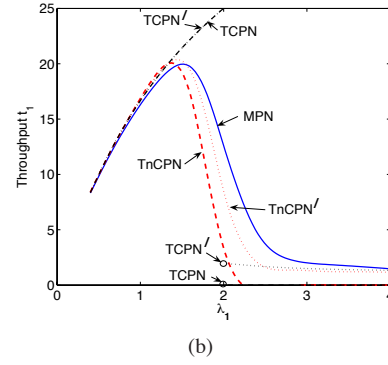
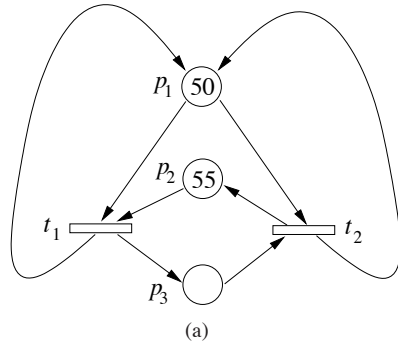


Fig. 4. Throughput for the discrete PN system of fig. 4(a) and its corresponding continuous relaxations, for rates $\lambda_1 \in [0.4, 4]$, $\lambda_2 = 1$ and initial marking. In (b), $TCPN'$ and $TnCPN'$ represent the fluid models in which the spurious deadlock has been previously removed. There is a discontinuity in $TCPN$ and $TCPN'$ at $\lambda_1 = 2$.

the performance evaluation perspective.

Two techniques have been considered for improving the fluid approximation. The first one corresponds to the stochastic extension of the timed CPN obtained by adding some gaussian *noise*. This solution improves the approximation if the marking of the system is strictly inside a region [36]. The second solution is to remove spurious solutions (i.e., solutions of the state equation that represent unreachable markings), especially those corresponding to deadlocks in continuous nets. The method has been considered for discrete PNs in [37], [38] and their application to CPN in [12]. In fig. 4 it is sketched the steady state dependency of the throughput for the same CPN system with and without these improvements of the model.

III. FLUIDIZATION OF OTHER DISCRETE EVENT SYSTEMS FORMALISMS

A. Queueing Networks

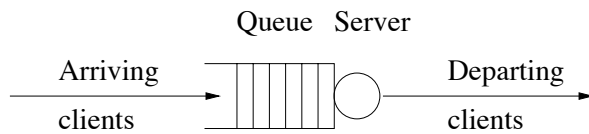


Fig. 5. A Single Service Center

It can be said that the performance evaluation theory goes back to A. K. Erlang, who worked in telecommunications, and Queueing Network (QN) is becoming a mature field in the 1960's [39]. A QN is a collection of *service centers*, which represent *system resources*, and *customers*, which represent *clients* or *users*. QN theory can be applied to transportation systems, call centers, telecommunication systems, traffic flow, etc. Fig. 5 shows a single service center where the clients arrive, wait in the queue (if necessary), receive service from the server, and depart. In fact, this service center and its arriving customers constitute a queueing model.

From the beginning, QN theory has been focused on providing high level primitives to ensure simple expressive building blocks and to restrict the models that can be specified to those that can be analyzed in a relatively efficient way

[40]. Very roughly speaking, QNs theory has been developed in a *bottom-up* approach.

One important limitation of the initial proposal of QNs was the absence of a general construction to deal with synchronizations. At the end of the 1970s, with the emergency of parallel and distributed systems, the need of synchronization becomes evident. For example, a synchronization primitive is required to describe: (i) *cooperation* relationships, as the assembly of parts A and B into a new reality or par-begin/par-end construction, and (ii) *competition* relationships, as the sharing of servers among two different production lines, or the need for several resources in order to advance some production. As a consequence to this challenge some *ad hoc* extensions have been introduced, e.g., *Extended Queueing Networks* (EQNs) [41].

Proceeding in that way, EQNs began to use a variety of specific primitives to handle synchronizations and resource constraints, sometimes with a clear redundancy in basic objects. For example, *passive resources* and *customers* reside in different kinds of nodes, or different nodes are used to model a *join* and a *resource acquisition*. In fig. 6 basic relationships between the elements of stochastic Petri nets and EQNs are provided. In this correspondence, places represent waiting rooms, timed transitions represent stations (servers), routing is modeled by means of immediate transitions with branching probabilities, and transitions can be seen as a simple and clean general synchronization primitive. In [40], a comparison has been undertaken between synchronized queueing networks and stochastic Petri nets.

Fig. 7 presents the same system modeled as stochastic PN (fig. 7. (a)) and as extended queueing network (fig. 7. (b)). Obviously, against ready to use and ad-hoc constructions in EQNs, the bipartite structure of the PN model makes more attractive formal reasoning and theory construction.

Another kind of extension of QNs in order to consider synchronization is the so called *Fork Join QN with Blocking* [43]. This class of models has similar structure with strongly connected *Marked Graphs*, a well-understood subclass of PNs. Based on this strong relation, many performance solution techniques can be adapted from QNs to PNs, e.g.,

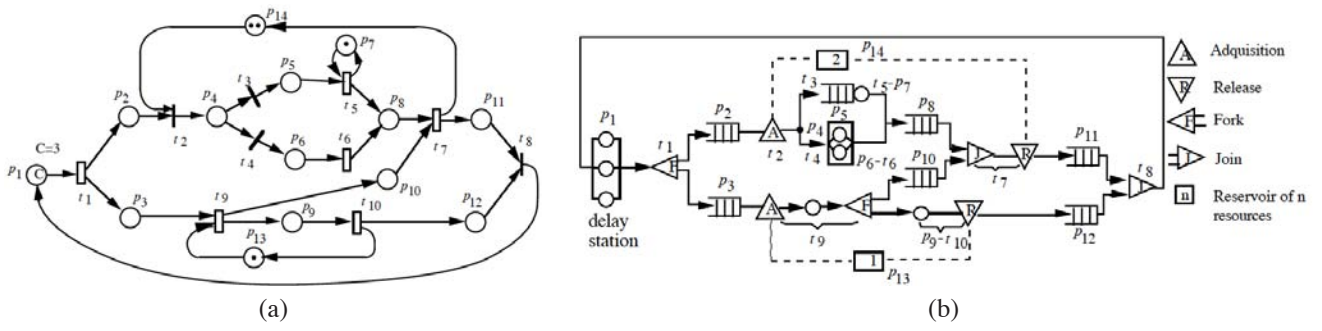


Fig. 7. A stochastic Petri net and the corresponding extended queueing network [42].

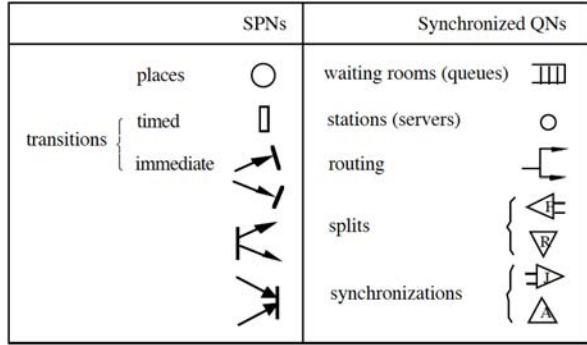


Fig. 6. Relationships between stochastic PNs and EQNs [42].

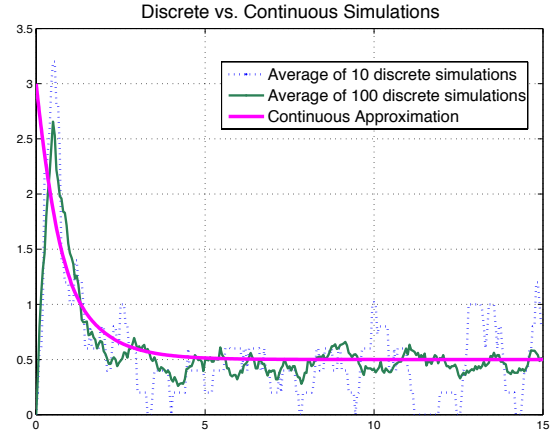


Fig. 8. Simulations of the PN system in fig. 7 as discrete and fluidized.

performance bounds, mean value analysis, response time approximations or tensor based computations [44].

The first book dealing with fluid approximations of QNs is [8]. The general acceptance of fluidization by the QN community was a slow process, being delayed for almost two decades. The fluid limit equations can be derived from the discrete QN equations by using natural extensions of the well-known law of large numbers and central limit theorem from probability theory. Its analogues in stochastic processes theory are, the *Functional Strong Laws of Large Numbers* and, dealing with distributions, the *Functional Central Limit Theorem* (also known as *Donsker Theorem*). In both cases, the idea is to study the convergence of a sequence of stochastic processes to another one, generating simple approximations (see, for example, [45]). These approximations have been used in order to compute performance, analyze stability, or optimize the behavior of QN models, for example.

The fluid relaxed model of QNs may be *deterministic* if only the functional strong law of large numbers or similar results are used [46]. Roughly speaking, in this case, arrival and services processes are replaced by their intensities, i.e., expectations or average values. Nevertheless, even if providing a simplified view, the deterministic approximation may exhibit very complex behaviors. For example, as pointed out in [47], deterministic fluid QN models do not need to have an unique solution, since they might bifurcate. Moreover, interpreting some continuous PNs as fluid EQNs, even *undecidability* issues can appear [28]. Fig. 8 shows the simulations of the CPN system in fig. 7 as discrete

and continuous obtained using *SimHPN* toolbox [48]. With dotted line is illustrated the average result obtained making 10 discrete simulations, with dashed line the average of 100 discrete simulations while with solid line the fluid simulations. As it is expected, the fluid approximation is better when more discrete simulations are considered.

Among the topics considered in the fluid QNs literature, the questions related to the quality of the approximation are important. When fluctuations cannot be neglected with respect to average values, e.g., because the population is not truly that *big*, the fluid model should be described in terms of *stochastic differential equations*. In this last case the noise on the differential equations partially reflect the stochastic variability on the behavior of the original discrete QN.

Many problems have been considered in the literature of fluid QNs [49], [50]. For example, the parametric optimization and dynamic control (sometimes referred as *scheduling* for the underlying discrete model) are considered in [51], [52], [53]. Among the many potential interests in the consideration of fluid models is the analysis of the *stability* of discrete QNs (in PNs terms, corresponds to the boundedness property). This is a topic that in the mid of 1980s was already stated to have “achieved (in certain cases) a striking success by providing a complete answer to the question of stability of stochastic networks” [54].

B. Stochastic Process Algebras

Born inside the Computer Science community, *Process Algebras* (PAs) constitute another basic paradigms to model DES. If PNs were mainly defined to express “concurrency and synchronization”, PAs base their modeling view on “components and composition”, a generalization of *product of automata*. Like PNs, PAs have been defined formally in a top-down manner considering first untimed model while the time dimension has been introduced in a second step.

The three most classical process algebras are: (i) *Calculus of Communicating Systems* (CSS) [55], (ii) *Communicating Sequential Processes* (CSP) [56] and, (iii) *Algebra of Communicating Processes* (ACP). CSP considers local variables and a message passing paradigm of communication while ACP emphasize the algebraic features, introducing the noun phrase “process algebra”. These proposals belong to an enormous set in which the assumptions are little different.

In 1990s *stochastic* PAs have been defined, extending the original untimed model definition with stochastic timings. Several proposal appears [57]: *Timed Processes and Performance evaluation* (TIPP), *Performance Evaluation Process Algebra* (PEPA), *Markovian Process Algebra* (MPA), and *Extended Markovian Process Algebra* (EMPA).

If fluid or continuous PNs have a clear existence by the beginning of the 1990s [58], [59], fluid PAs start to be defined in the second half of the first decade of this century, i.e., somehow a decade and half later. Among those, PEPA [60] is a basic markovian PA in which basic components are sequential processes (finite automata), while parallel composition is only supported at top level; its fluid-flow approximation consider large scale models of massively repeated sequential components. In this restricted case, the small set of combinators contains *prefix* and *choice*, representing a sequential behavior and alternatives, and *cooperation*, that defines synchronizations. Moreover, with *hiding* it is possible to abstract aspects of the components’ behavior (roughly speaking, this “parallel” in PNs to *silent, immediate, non-observable...* transitions). Among other constraints for the fluidization, components of the same type do not cooperate, i.e., synchronize. Having in mind chemical reactions, Cardelli (see, for example, [61]) consider the fluidification of other SPAs. Like with fluid QNs or continuous PNs, the basic goal is to look for a set of coupled *ordinary differential equations* as the underlying mathematical representation of the approximated behavior.

As in automata, in the PA paradigm the representation of the state is symbolic, what is not appropriate for fluidization. As it is well-known, in PNs (think particularly in *Place/Transition nets*), the distributed state is given by a numerical vector, the *marking*. Following that line of thinking, in [60] a numerically aggregated representation scheme is defined for PEPA expressions with replicated components. It explicitly introduces integer counters to define the state space, dealing with a state representation in *numerical vector form* which can be subject to a fluid-flow approximation. Obtained by aggregating all identical non-synchronized se-

quential components, the structure of the model is based in a so called *activity matrix* (its dimension is the number of *activities* by the number of distinct local *derivatives*), and timing is defined through a *rate function* per transition. Obviously the activity matrix is nothing more than the *incidence matrix* of an underlying Petri Net (where *activities* are transitions and *derivatives* are places). This obvious fact has been recognized sometime later [62], [63]. Additionally, due to constraints on the considered subclass of process algebra models, the underlying Petri nets are *State Machine Decomposable*, an ordinary, i.e., no weights on the arcs, net subclass that is structurally bounded. Moreover, assuming interactions aimed at computer networks, a *bounded capacity law* was considered in PEPA from the beginning: components cannot perform activities in faster way by cooperation, so the rate of a shared activity is the *minimum* of the apparent rates of the activity in the cooperating components. Therefore, the underlying CPN consider the so called *infinite server semantics* for the fluid model (on subsequent works dealing with *signaling pathways*, borrowing terminology from chemistry, the *law of mass action* is added: it replaces the minimum operator by the *product*, as in population dynamics [10]). A very positive point of this connection is that analysis and control of PEPA models can immediately use PN theory and techniques, and vice versa, an implicit goal in [64], [65], when comparing the expressiveness of PEPA and bounded SPN models. Extensions of this limited subclass of PA models can be found in [66], where more than constructing a limiting-deterministic approximation, higher order moments (in particular, variances) are estimated, something necessary to guess the accuracy of the fluid-flow approximation in a given situation.

In general, any PEPA model can be transformed into a CPN using the differential equations of the model. Let us consider a system composed by two servers offering the same service at different rates. Let us assume that the clients can be served by any of the server. A PEPA model is the following one:

$$\begin{aligned}
 C &\stackrel{def}{=} (serv_1, s_1) . C' \\
 C' &\stackrel{def}{=} (do, d) . C \\
 S'_i &\stackrel{def}{=} (reset_i, r_i) . S_i \\
 S_i &\stackrel{def}{=} (serv_i, s_i) . S'_i \\
 Sys &\stackrel{def}{=} C[100] \{serv_1, serv_2\} (S_1[50] || S_2[50])
 \end{aligned} \tag{1}$$

This is equivalent to the continuous Petri net in fig. 9 where infinite server semantics is assumed. Using SimHPN toolbox, the system can be simulated and the results are shown in fig. 10.

IV. DIRECT FLUID VIEWS OF SYSTEMS

In many natural and technological systems the consideration of discrete event models may be conceptually the more faithful view. Nevertheless, models being abstractions, fluid views may be directly the more reasonable for stability

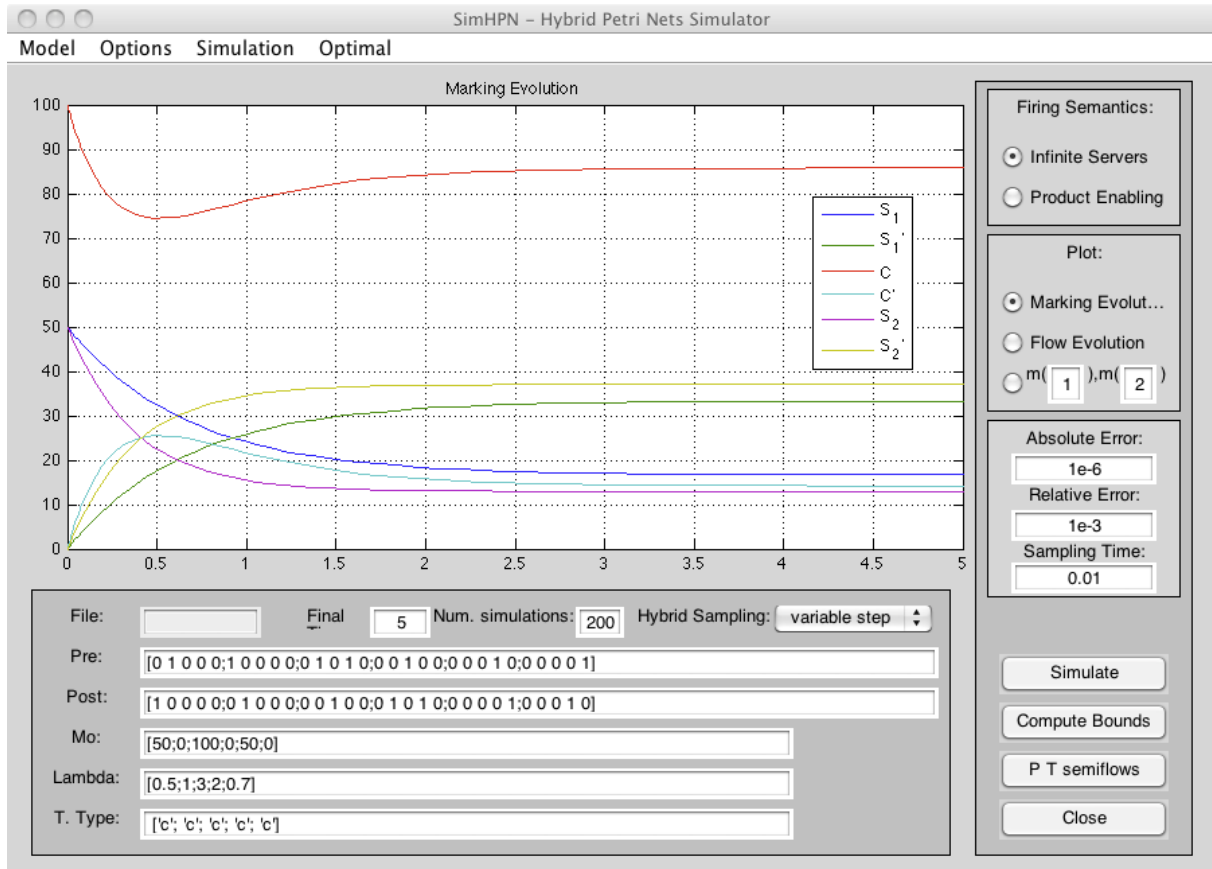


Fig. 10. Sketch of the main window of SimHPN toolbox illustrating the simulation of CPN in fig. 9.

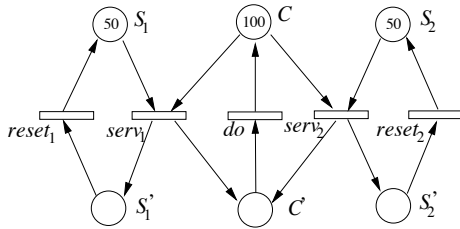


Fig. 9. Continuous Petri net system equivalent to the PEPA model in eq. (1).

analysis, performance evaluation, sensitivity analysis or optimization. As already said, fluid models may be deterministic (first-moment approximation) or stochastic. Nevertheless, the basic pattern of the systems of equations is organized around the simple “mass balance or accounting principle”: *the rate of accumulation*, e.g., of customers in queues, is equal to the difference of incoming and outgoing flows.

Even if conceptually thought as continuous, those models usually are technically hybrid (among other reasons, because of upper bounds on capacities or non-negativity on the level of “reservoirs”). From a System Theory point of view, the peculiarities of models are in the definition of the input and output flows. Among an infinity of possibilities, they can be linear, piecewise linear (as happen with CPNs under infinite server semantics), or bilinear (as happen frequently

in population dynamic problems).

One of this type of system is *Compartmental Systems* (CSs) that are composed by a finite number of subsystems (compartments), interacting by exchanging nonnegative quantities of material and energy among the compartments and with the environment [67], [68]. Compartmental “views” of systems are used in biology, medicine and ecology, among many other application fields. The system is governed by laws of transfer and conservation, while the state variables are constrained to remain nonnegative along the system trajectories. Compartmental and PNs systems are considered in [69]. Due to the existence of synchronizations (joins or *rendez-vous*, and arc weights), it can be said that the structure of PNs is richer. Nevertheless, considering the semantics associated to the networks, in PNs there is a problem to represent recipient controlled compartmental systems in a “natural” way (because in PNs the flows are defined according to the marking of places at the precondition, not of the marking of the subsequent places).

Stochastic Fluid Systems are defined using the same kind of “mass balance principle” [70]. Like in compartmental systems, the derivatives of the state variables represent the level of a set of reservoirs, while controls may be applied to regulate the amount of flows. In this case, graphs illustrate about flows, reservoirs and their connections. The key point is how equations are written, something for which there is

a significant degree of freedom. In order to address optimization problems or sensitivity analysis, exogenous and endogenous events should be considered (the former referring to changes in the defining process, the latter characterizing points where the state of the system enters a certain region). In this framework, *Infinitesimal Perturbation Analysis* (IPA) has been successfully explored in several cases in order to optimize the behavior of the system (see, for example, [71], [72]).

Finally, let us consider *System Dynamics* (SDs). Abstracting of the possible discrete “nature” of the system under consideration, systems are modeled as continuous or hybrid, now using two kind of diagrams. Here we do not explicitly deal with the methodological aspects, limiting our interest to point out the existence of the so called *Forrester Diagrams* (FDs), also expressively known as *Stock and Flow Diagrams* [73]. This kind of diagrams allows the quantitative modeling of the relationships between the parts by means of a catalog of symbols, which correspond to a classical hydrodynamic interpretation of the system (see Figure 11).

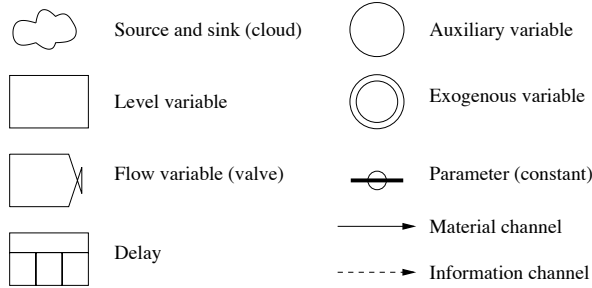


Fig. 11. Basic components of a FD model

The *stocks* correspond to the name of state variables in systems, while their values are the *levels* (stocks accumulate “material” coming from *material channels*); the *valves* determine the speed of the material flow through material channels; the required information is transmitted instantaneously by means of *information channels*; *auxiliary variables* correspond to intermediate steps in the calculation of functions associated to the valves; the *clouds* represent sources and sinks; the interaction of the system with the exterior is represented with *exogenous* variables; the *delays* can affect the material or information transmission but they do not increase the modeling power of the formalism.

Elements of comparison between FDs and CPNs are provided in [74], [75]. At the structural level, like in EQNs, the variety of symbols in FDs contrast with the frugality of basic symbols in PNs. Additionally, in FDs there exist not only *material flows*, but also a graphical representation of *information flows*. Concerning the interpretation of the graph, like in compartmental systems (or SFMs, where noise is also considered) there exist a great liberty in defining the flows through arbitrary functions. In contrast, in CPNs the flow functions will be constrained by the particular server semantics (*finite*, *infinite*, *population* or *product*, etc.), anyhow using only local state variables in the precondition of

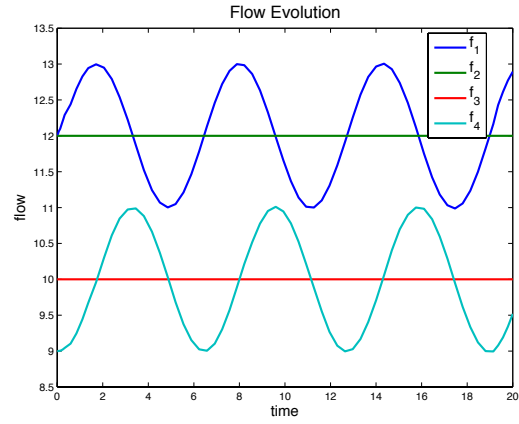


Fig. 13. Simulations results of continuous Petri net in fig. 12.(b)

the transition. Roughly speaking, if CNs, FDs and SFMs have increased expressivity in quantitatively defining the “flows”, the explicit presence of synchronizations in PNs make more natural the expression of simple tasks as the assembly of two kind of parts.

In fig. 13 it is shown a simulation result obtained using SimHPN toolbox of the continuous PN in fig. 12.(b) that is equivalent with the system represented by the Forrester diagram in fig. 12.(a). It has been considered an initial marking $\mathbf{m}_0 = [9 \ 12 \ 1 \ 1]^T$ and a firing rate vector $\boldsymbol{\lambda} = [1 \ 12 \ 10 \ 1]^T$. As it can be observed, this system has no steady state and corresponds to a simple harmonic oscillator.

V. CONCLUDING REMARKS

Fluidization of DES is a classical relaxation technique, used from the late sixties of the pass century in the Queueing Network framework. The gain by fluidization is a considerable reduction of computational complexity, in many practical problems. Nevertheless, since timed CPN under infinite server semantics can simulate Turing machines, some properties remain not decidable. However, a loss of fidelity due to fluidization implies that the results obtained in the relaxed model are not the exact solutions of the original model. Moreover, some properties cannot be studied in the relaxed model, for example the mutual exclusion.

In this paper we present a very broad description of the fluidization of different DES or directly fluid views formalisms. The central one is continuous Petri nets. Moreover, others are introduces making emphasis on the connection to CPNs. Let us point out that (E)QNs, CNs, FDs or SFMs were directly defined as evolving in the time domain. Nevertheless, CPNs were primarily defined as fully non deterministic models, without any concept of time. This may be singularly interesting in order to check some properties like the potential presence of *deadlocks*, or the existence of other time independent constraints in the behavior, as certain *synchronic distances*.

Many technical problems require a deeper understanding:

- How to do a better (the best) fluidization?
- How to guess the approximation errors?

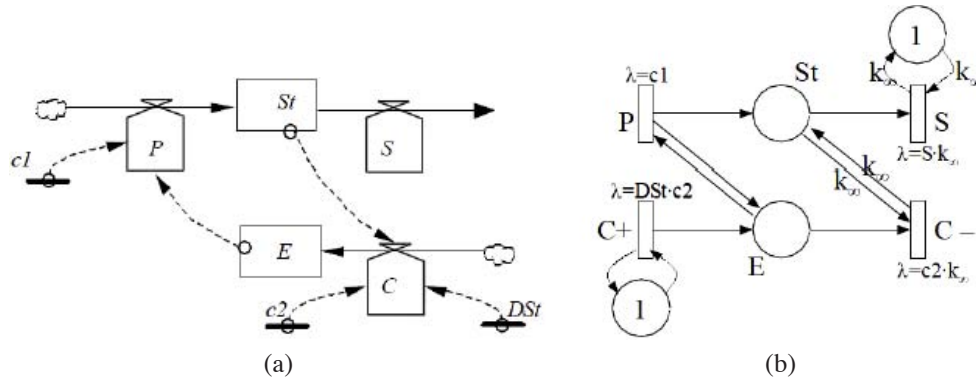


Fig. 12. A continuous Petri net (b) and the corresponding FD model (a).

- How to establish boundaries (class of models, type of properties, etc.) among problems that become computationally tractable or decidable?

In our view, this is a needed and a promising line of research.

REFERENCES

- [1] R. David and H. Alla, *Discrete, Continuous and Hybrid Petri Nets*. Berlin: Springer, 2004, (Revised 2nd edition, 2010).
- [2] M. Silva, "Introducing Petri Nets," in *Practice of Petri Nets in Manufacturing*. Chapman & Hall, 1993, pp. 1–62.
- [3] G. Nemhauser and L. Wolsey, *Integer and combinatorial optimization*. Wiley New York, 1988.
- [4] M. Silva and J. Colom, "On the computation of structural synchronic invariants in P/T nets," *Advances in Petri Nets 1988*, pp. 386–417, 1988.
- [5] G. Berthelot, "Checking Properties of Nets Using Transformations," in *Advances in Petri Nets 1985*, ser. Lecture Notes in Computer Science, G. Rozenberg, Ed. Springer, 1986, vol. 222, pp. 19–40.
- [6] M. Silva, E. Teruel, and J. M. Colom, "Linear Algebraic and Linear Programming Techniques for the Analysis of Net Systems," in *Lectures in Petri Nets. I: Basic Models*, ser. Lecture Notes in Computer Science, G. Rozenberg and W. Reisig, Eds. Springer, 1998, vol. 1491, pp. 309–373.
- [7] L. Recalde, E. Teruel, and M. Silva, "On Linear Algebraic Techniques for Liveness Analysis of P/T Systems," *Journal of Circuits, Systems, and Computers*, vol. 8, no. 1, pp. 223–265, 1998.
- [8] G. F. Newell, *Applications of Queuing Theory*. Chapman and Hall, 1971 (2nd ed. 1982).
- [9] L. Recalde and M. Silva, "Petri Nets Fluidification revisited: Semantics and Steady state," *APII-JESA*, vol. 35, no. 4, pp. 435–449, 2001.
- [10] M. Silva and L. Recalde, "Petri nets and integrality relaxations: A view of continuous Petri nets," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 32, no. 4, pp. 314–327, 2002.
- [11] —, "On fluidification of Petri net models: from discrete to hybrid and continuous models," *Annual Reviews in Control*, vol. 28, pp. 253–266, 2004.
- [12] M. Silva, J. Júlvez, C. Mahulea, and C. R. Vázquez, "On Fluidization of Discrete Event Models: Observation and Control of Continuous Petri nets," *Discrete Event Dynamic Systems: Theory and Application*, 2011, in press.
- [13] M. Silva and E. Teruel, "A Systems Theory Perspective of Discrete Event Dynamic Systems: The Petri Net Paradigm," in *Symp. on Discrete Events and Manufacturing Systems. CESA '96 IMACS Multiconference*, P. Borne, J. C. Gentina, E. Craye, and S. El Khattabi, Eds., Lille, France, July 1996, pp. 1–12.
- [14] —, "DEDS Along Their Life Cycle. Interpreted Extensions of Petri Nets," in *IEEE Int. Conf. on Systems, Man and Cybernetics*, La Jolla, San Diego, CA, USA, October 1998.
- [15] C. A. Petri, *Communication With Automata*, ser. Supplement 1 to Technical Report RADC-TR-65-377, Vol. 1, Griffiss Air Force Base, New York 1966. [Originally published in German: Kommunikation mit Automaten, University of Bonn], 1962.
- [16] B. Berthomieu and M. Diaz, "Modeling and Verification of Time Dependent Systems Using Time Petri Nets," *IEEE Trans. on Software Engineering*, vol. 17, no. 3, pp. 259–273, 1991.
- [17] J. Cardoso, R. Valette, and D. Dubois, "Fuzzy Petri Nets: An Overview," in *13th IFAC World Congress*, San Francisco, CA, USA, July 1996.
- [18] M. K. Molloy, *Fundamentals of Performance Modeling*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1988.
- [19] M. Ajmone-Marsan, G. Balbo, A. Bobbio, G. Chiola, G. Conte, and A. Cumani, "The effect of execution policies on the semantics and analysis of stochastic Petri nets," *IEEE Trans. on Software Engineering*, vol. 15, no. 7, pp. 832–846, 1989.
- [20] D. Lefebvre, "Feedback control design for manufacturing systems modeled by continuous Petri nets," *International Journal of Systems Science*, vol. 30, no. 6, pp. 591–600, 1999.
- [21] A. Amrah, N. Zerhouni, and A. E. Moudni, "On the control of manufacturing lines modelled by controlled continuous Petri nets," *International Journal of Systems Science*, vol. 29, no. 2, pp. 127–137, 1998.
- [22] M. Allam and H. Alla, "Modeling and simulation of an electronic component manufacturing systems using hybrid Petri nets," *IEEE Trans. on Semiconductor Manufacturing*, vol. 11, no. 3, pp. 374–383, 1998.
- [23] J. Júlvez and R. Boel, "A continuous petri net approach for model predictive control of traffic systems," *IEEE Trans. on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 2010.
- [24] M. Dotoli, M. Fanti, A. Giua, and C. Seatzu, "Modelling systems by hybrid Petri nets. An application to supply chains," in *Petri Net, Theory and Application*, V. Kordic, Ed. InTech, 2008.
- [25] L. Recalde, E. Teruel, and M. Silva, "Autonomous Continuous P/T systems," in *Application and Theory of Petri Nets 1999*, ser. Lecture Notes in Computer Science, J. K. S. Donatelli, Ed., vol. 1639. Springer, 1999, pp. 107–126.
- [26] J. Júlvez, L. Recalde, and M. Silva, "On reachability in autonomous continuous Petri net systems," in *24th Int. Conf. on Application and Theory of Petri Nets (ICATPN 2003)*, ser. Lecture Notes in Computer Science, W. van der Aalst and E. Best, Eds. The Netherlands: Springer, June 2003, vol. 2679, pp. 221–240.
- [27] —, "Deadlock-Freeness Analysis of Continuous Mono-T-Semiflow Petri Nets," *Automatic Control, IEEE Trans. on*, vol. 51, no. 9, pp. 1472–1481, Sept. 2006.
- [28] L. Recalde, S. Haddad, and M. Silva, "Continuous Petri Nets: Expressive Power and Decidability Issues," *Int. Journal of Foundations of Computer Science*, vol. 21, no. 2, pp. 235–256, 2010.
- [29] C. Mahulea, L. Recalde, and M. Silva, "Basic server semantics and performance monotonicity of continuous Petri nets," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 19, no. 2, pp. 189–212, 2009.
- [30] C. Mahulea, A. Ramirez, L. Recalde, and M. Silva, "Steady state control reference and token conservation laws in continuous Petri net systems," *IEEE Transactions on Automation Science and Engineering*, vol. 5, no. 2, pp. 307–320, April 2008.
- [31] C. R. Vázquez, A. Ramírez, L. Recalde, and M. Silva, "On controllability of timed continuous Petri nets," in *Egerstedt M., Mishra B. eds.: Proc. of the 11th Int. Workshop Hybrid Systems: Computational and Control (HSCC08)*, LNCS, vol. 4981, 2008, pp. 528–541.

- [32] C. Mahulea, A. Giua, L. Recalde, C. Seatzu, and M. Silva, "Optimal model predictive control of timed continuous Petri nets," *IEEE Transactions on Automatic Control*, vol. 53, no. 7, pp. 1731–1735, August 2008.
- [33] J. Xu, L. Recalde, and M. Silva, "Tracking control of join-free timed continuous Petri net systems under infinite servers semantics," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 18, no. 2, pp. 263–288, 2008.
- [34] H. Apaydin-Özkan, C. Mahulea, J. Júlvez, and M. Silva, "Approaching minimum time control of timed continuous petri nets," *Nonlinear Analysis of Hybrid Systems*, vol. 5, no. 2, pp. 136–148, May 2011.
- [35] C. Mahulea, L. Recalde, and M. Silva, "Observability of continuous Petri nets with infinite server semantics," *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 219–232, May 2010.
- [36] C. R. Vázquez, L. Recalde, and M. Silva, "Stochastic–continuous state approximation of Markovian Petri net systems," in *Proc. of the 47th IEEE Conf. on Decision and Control (CDC)*, 2008.
- [37] J. M. Colom and M. Silva, "Improving the Linearly Based Characterization of P/T Nets," in *Advances in Petri Nets 1990*, ser. Lecture Notes in Computer Science, G. Rozenberg, Ed. Springer, 1991, vol. 483, pp. 113–145.
- [38] J. Campos, J. M. Colom, and M. Silva, "Improving throughput upper bounds for net based models of manufacturing systems," *Robotics and Flexible Manufacturing Systems*, pp. 281–294, 1992.
- [39] J. Jackson, "Networks of waiting lines," *Operations Research*, vol. 5, no. 4, pp. 518–521, 1957.
- [40] M. Vernon, J. Zahorjan, and E. D. Lazowska, "A Comparison of Performance Petri Nets and Queueing Network Models," in *3rd Int. Workshop on Modelling Techniques and Performance Evaluation*, Paris, France, Mar. 1987.
- [41] G. Bolch, S. Greiner, H. de Meer, and K. Trivedi, *Queueing Networks and Markov Chains*. New York: Wiley, 1998.
- [42] J. Campos and M. Silva, "Structural Techniques and Performance Bounds of Stochastic Petri Net Models," in *Advances in Petri Nets*, ser. Lecture Notes in Computer Science, G. Rozenberg, Ed. Springer, 1992, vol. 609, pp. 352–391.
- [43] Y. Dallery, Z. Liu, and D. Towsley, "Properties of Fork/Join Queueing Networks with Blocking under various operating mechanisms," *IEEE Trans. on Robotics and Automation*, vol. 13, no. 4, pp. 503–518, August 1997.
- [44] G. Balbo and M. Silva, Eds., *Performance Models for Discrete Event Systems with Synchronizations: Formalisms and Analysis Techniques*, *Proc. of Human Capital and Mobility MATCH Performance Advanced School*, Jaca, Spain, 1998.
- [45] W. Whitt, *Stochastic Process Limits. An introduction to stochastic-process limits and their application to queues*, ser. Series in Operations Research and Financial Engineering. New York: Springer, 2002.
- [46] H. Chen and A. Mandelbaum, "Discrete flow networks: Bottleneck analysis and fluid approximations," *Mathematical Operations Research*, vol. 16, pp. 408–446, 1991.
- [47] M. Bramson, *Stability of Queueing Networks*, ser. Lecture Notes in Mathematics. New York: Springer, 2008 (also published in: *Probability Surveys*, 5:169–345, 2008), no. 1950.
- [48] J. Júlvez, C. Mahulea, and C. Vázquez, "Analysis and Simulation of Manufacturing Systems using SimHPN toolbox," in *CASE11: 7th IEEE Conference on Automation Science and Engineering*, Trieste, Italy, August 2011.
- [49] L. Kleinrock, *Queueing Systems, Volume II: Computer Applications*. Wiley, 1976, vol. 2.
- [50] F. Kelly, S. Zachary, and I. Ziedins, Eds., *Stochastic Networks: Theory and Applications*, ser. Royal Statistical Society LNS, 4. Oxford University Press, 1996 (repr. 2002).
- [51] F. Moss and A. Segall, "An optimal control approach to Dynamic Routing in Networks," *IEEE Trans. on Automatic Control*, vol. 27, no. 2, pp. 329–339, 1982.
- [52] D. Connors, G. Feigin, and D. Yao, "Scheduling semiconductor lines using a fluid network model," *IEEE Trans. on Robotics and Automation*, vol. 10, no. 2, pp. 88–98, 1994.
- [53] Y. Liu and W. Gong, "Perturbation Analysis for Stochastic Fluid Queueing Systems," *Discrete Event Dynamic Systems*, vol. 12, no. 4, pp. 391–416, 2002.
- [54] F. Avram, "Optimal control of fluid limits of queueing networks and stochasticity corrections," in *Mathematics of stochastic manufacturing systems*, vol. 33. Amer Mathematical Society, 1997, pp. 1–37.
- [55] R. Milner, *A Calculus of Communicating Systems*, ser. Lecture Notes in Computer Science. Springer Verlag, 1980, no. 92.
- [56] C. Hoare, *Communicating Sequential Processes*. Prentice Hall, 1985.
- [57] H. Hermanns, U. Herzog, and J.-P. Katoen, "Process algebra for performance evaluation," *Theoretical Computer Science*, vol. 274, no. 1–2, pp. 43–87, 2002.
- [58] R. David and H. Alla, "Continuous Petri Nets," in *Proc. of the 8th European Workshop on Application and Theory of Petri Nets*, Zaragoza, Spain, 1987, pp. 275–294.
- [59] —, "Autonomous and timed Continuous Petri Nets," in *Proc. 11th Int. Conf. on Application and Theory of Petri Nets*, Paris, France, 1990, pp. 367–386.
- [60] J. Hillston, "Fluid Flow Approximation of PEPA models," in *Procs. of the Second Int. Conf. on the Quantitative Evaluation of Systems (QEST)*. IEEE Computer Society., 2005, pp. 33–43.
- [61] L. Cardelli, "On process rate semantics," *Theoretical Computer Science*, vol. 391, no. 3, pp. 190–215, 2008.
- [62] V. Galpin, "Continuous approximation of PEPA models and Petri Nets," *Int. Journal of Computer Aided Engineering and Technology*, vol. 2, no. 4, pp. 324–339, 2010.
- [63] J. Ding, "Structural and Fluid Analysis for large scale PEPA models, with applications to content adaptation systems," Ph.D. dissertation, University of Edinburgh, 2010.
- [64] S. Donatelli, M. Ribaud, and J. Hillston, "A comparison of performance evaluation process algebra and generalized stochastic Petri nets," in *IEEE Petri Nets and Performance Models (PNPM'95)*, 1995, pp. 158–168.
- [65] J. Hillston, L. Recalde, M. Ribaud, and M. Silva, "A Comparison of the Expressiveness of SPA and Bounded SPN models," in *In Procs. of the 9th Int. Workshop on Petri Nets and Performance Models*. IEEE Computer Society Press, Aachen, September, 2001, pp. 197–206.
- [66] R. A. Hayden and J. T. Bradley, "A fluid analysis framework for a Markovian Process Algebra," *Theoretical Computer Science*, vol. 411, no. 22–24, pp. 2260–2297, 2010.
- [67] L. Benvenuti and L. Farina, "Positive and compartmental systems," *IEEE Trans. on Automatic Control*, vol. 47, no. 2, pp. 370–373, Feb 2002.
- [68] G. Walter and M. Contreras, *Compartmental Modeling With Networks*. Birkhauser Boston, 1999.
- [69] M. Silva and L. Recalde, "Unforced continuous Petri nets and positive systems," in *Positive Systems. Procs. of the First Multidisciplinary Int. Symp. on Positive Systems: Theory and Applications (POSTA 2003)*, ser. LNCIS, L. Benvenuti, A. D. Santis, and L. Farina, Eds., vol. 294. Rome: Springer, August 2003, pp. 55–62.
- [70] C. G. Cassandras, "Stochastic Flow Systems: Modeling and Sensitivity Analysis," *Chapter 6 in Stochastic Hybrid Systems (C. G. Cassandras and J. Lygeros, eds.)*, pp. 139–167, 2007.
- [71] G. Sun, C. G. Cassandras, and C. G. Panayiotou, "Perturbation Analysis of Multiclass Stochastic Fluid Models," *Discrete Event Dynamic Systems*, vol. 14, no. 3, pp. 267–307, 2004.
- [72] C. Yao and C. G. Cassandras, "Perturbation Analysis and Optimization of Multiclass Multiobjective Stochastic Flow Models," in *IEEE Int. Conf. on Decision and Control, CDC10*, Shanghai, 2010, pp. 914–919.
- [73] J. W. Forrester, *Industrial Dynamics*. Cambridge, Mass.: MIT Press, 1961.
- [74] E. Jiménez, L. Recalde, and M. Silva, "Forrester Diagrams and Continuous Petri Nets: A Comparative View," in *Proc. of the 8th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA 2001)*, 2001, pp. 85–94.
- [75] E. Jiménez, J. Júlvez, L. Recalde, and M. Silva, "Relaxed Continuous Views of Discrete Event Systems: Petri Nets, Forrester Diagrams and ODES," in *IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC)*, The Hague, The Netherlands, Oct. 2004.

Recent Developments in SLICOT Library and Related Tools

Vasile Sima

Abstract—The latest developments of the SLICOT Library and associated tools are discussed, and issues related to its future evolution are summarized. Structure-exploiting solvers for skew-Hamiltonian/Hamiltonian matrix pencils have been added. Possible applications include the computation of L_∞ - or H_∞ -norms of linear time-invariant multivariable systems, and solution of algebraic matrix Riccati equations.

I. INTRODUCTION

The SLICOT Library has been initially developed (starting in the early eighties) by the Working Group on Software (WGS), partly with external contributors, and then in cooperation with the Numerical Algorithms Group, Ltd. (NAG). The first two releases, [1], [2], based on the well-known NAG mathematical library, have been commercial versions, distributed by NAG. A public-domain version of the SLICOT Library (Release 3) has been developed during 1996–1999 [3], built on numerical linear algebra subroutines from the state-of-the-art software packages LAPACK (Linear Algebra Package) [4] and BLAS (Basic Linear Algebra Subprograms) [5], [6], [7]. The conversion offered the opportunity to improve the modularity, functionality, reliability, as well as the performance of the codes. Moreover, new algorithmic developments for the analysis and design of control systems have been included. The following major SLICOT updates were released in November 1999 (Release 4) and June 2005 (Release 5), but there were, and continue to be, many other updates. An important update took place in June 2009, when a lot of improvements have been performed and many new routines, mainly standardized versions of the LAPACK routines (see <http://www.tu-chemnitz.de/mathematik/hapack/>) for (skew-)Hamiltonian eigenvalue problems, have been added. (Related papers can be found at <http://www.tu-chemnitz.de/mathematik/hapack/pub/>.) All changes performed since 1996 have been documented in the Release files available on the SLICOT Library Web site www.slicot.org. The library contents, the associated MATLAB[®] (and Scilab) tools, and their performance have been presented, e.g., in [3], [8], [9], [10], [11], [12].

This paper discusses the latest developments of the SLICOT Library and associated tools, and summarizes some issues related to its possible future evolution. Statistics about the current library contents (specifically, the number of routines per chapter) and volume (including the associated MATLAB tools) are shown in Table I and Table II.

V. Sima is with the National Institute for Research & Development in Informatics, Bd. Averescu, Nr. 8–10, 011455 Bucharest, Romania vsima@ici.ro

TABLE I
SLICOT LIBRARY CONTENTS SUMMARY

Chapter	Main	Support	Total
Analysis	42	15	57
Benchmark	6	0	6
Data analysis	7	0+2	9
Filtering	6	0	6
Identification	5	10	15
Mathematical	84	106+17	207
Nonlinear	0	12+4	16
Synthesis	50	59+12	121
Transformation	40	11+4	55
Utility	7	0	7
Total	247	213+39	499

TABLE II
SLICOT LIBRARY AND MATLAB TOOLS SUMMARY

Category	Number	Volume
Source code for routines	499	≈ 8.4 MB
Documentation html files	465	≈ 5.3 MB
Source code for example programs	213	≈ 0.9 MB
Source code for MEX-files	61	≈ 1.3 MB
M-files (computation)	205	≈ 1.0 MB
M-files (tests)	89	≈ 0.9 MB
M-files (demonstration)	63	≈ 0.3 MB

The numbers after “+” signs in the third column of Table I represent auxiliary routines without html documentation. The volumes in Mega bytes (MB) include comments (except for the documentation html files).

II. RECENT SLICOT DEVELOPMENTS

The latest important update of the SLICOT Library took place in November 2010, when few improvements have been incorporated, and 29 new routines related to solving eigenproblems for skew-Hamiltonian/Hamiltonian matrix pencils have been added. Spectra and bases of stable deflating subspaces can be computed for such structured pencils. Several applications could benefit by the structure exploitation, including the calculation of L_∞ - or H_∞ -norms of linear time-invariant multivariable systems, and solution of algebraic matrix Riccati equations. Theory and algorithmic details are presented, e.g., in [13], [14]. Implementation issues and numerical results are given in [15], [16], [17], [18].

A. New SLICOT routines

A list of the new main implemented solvers for real matrix pencils, together with a short description of their functionality, is given below.

MB03LD - computes the eigenvalues of a real $2m \times 2m$ skew-Hamiltonian/Hamiltonian pencil $\alpha\mathcal{S} - \beta\mathcal{H}$,

$$\mathcal{S} = \begin{bmatrix} A & D \\ E & A^T \end{bmatrix}, \quad \mathcal{H} = \begin{bmatrix} C & V \\ W & -C^T \end{bmatrix}, \quad (1)$$

and $D^T = -D$, $E^T = -E$, $V^T = V$, $W^T = W$. Optionally, an orthonormal basis of the right deflating subspace of $\alpha\mathcal{S} - \beta\mathcal{H}$ corresponding to the stable eigenvalues (i.e., with strictly negative real part) is computed.

MB03LF - computes the eigenvalues of a real $2m \times 2m$ skew-Hamiltonian/Hamiltonian pencil $\alpha\mathcal{S} - \beta\mathcal{H}$, with $\mathcal{S} = \mathcal{J}\mathcal{Z}^T\mathcal{J}^T\mathcal{Z}$, $\mathcal{J} = \begin{bmatrix} 0 & I_m \\ -I_m & 0 \end{bmatrix}$, and $\mathcal{H}$ in (1). Optionally, an orthonormal basis of the right deflating subspace of $\alpha\mathcal{S} - \beta\mathcal{H}$ corresponding to the stable eigenvalues is computed. Optionally, an orthonormal basis of the companion subspace $\text{range}(P_U^-)$ [13], corresponding to the stable eigenvalues, is computed.

MB04AD - computes the eigenvalues and a generalized symplectic URV decomposition for a $2m \times 2m$ factored pencil $\alpha\mathcal{T}\mathcal{Z} - \beta\mathcal{H}$, $\mathcal{T} := \mathcal{J}\mathcal{Z}^T\mathcal{J}^T$, i.e., it determines orthogonal matrices $\mathcal{Q}_1$, $\mathcal{Q}_2$ and orthogonal symplectic matrices $\mathcal{U}_1$, $\mathcal{U}_2$ such that,

$$\begin{aligned} \mathcal{Q}_1^T \mathcal{T} \mathcal{U}_1 &= \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix}, \\ \mathcal{U}_2^T \mathcal{Z} \mathcal{Q}_2 &= \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & Z_{22} \end{bmatrix}, \\ \mathcal{Q}_1^T \mathcal{H} \mathcal{Q}_2 &= \begin{bmatrix} H_{11} & H_{12} \\ 0 & H_{22} \end{bmatrix}, \end{aligned}$$

where T_{11} , T_{22}^T , Z_{11} , Z_{22}^T , H_{11} are upper triangular and H_{22}^T is upper quasi-triangular. The matrices $\mathcal{U}_i$ are each represented by two arrays containing the submatrices U_{i1} and U_{i2} , $i = 1, 2$, such that

$$\mathcal{U}_1 = \begin{bmatrix} U_{11} & U_{12} \\ -U_{12} & U_{11} \end{bmatrix}, \quad \mathcal{U}_2 = \begin{bmatrix} U_{21} & U_{22} \\ -U_{22} & U_{21} \end{bmatrix}.$$

MB04BD - computes the eigenvalues of a real $2m \times 2m$ skew-Hamiltonian/Hamiltonian pencil $\alpha\mathcal{S} - \beta\mathcal{H}$ defined in (1). Optionally, decompositions of $\mathcal{S}$ and $\mathcal{H}$ are computed via orthogonal transformations $\mathcal{Q}_1$ and $\mathcal{Q}_2$, as follows:

$$\begin{aligned} \mathcal{Q}_1^T \mathcal{S} \mathcal{J} \mathcal{Q}_1 \mathcal{J}^T &= \begin{bmatrix} \tilde{A} & \tilde{D} \\ 0 & \tilde{A}^T \end{bmatrix}, \\ \mathcal{J}^T \mathcal{Q}_2^T \mathcal{J} \mathcal{S} \mathcal{Q}_2 &= \begin{bmatrix} \tilde{B} & \tilde{F} \\ 0 & \tilde{B}^T \end{bmatrix}, \\ \mathcal{Q}_1^T \mathcal{H} \mathcal{Q}_2 &= \begin{bmatrix} \tilde{C}_1 & \tilde{V} \\ 0 & \tilde{C}_2^T \end{bmatrix}, \end{aligned}$$

where $\tilde{A}$, $\tilde{B}$, $\tilde{C}_1$ are upper triangular, $\tilde{C}_2$ is upper quasi-triangular and $\tilde{D}$ and $\tilde{F}$ are skew-symmetric. Optionally, the orthogonal transformation matrices $\mathcal{Q}_1$ and $\mathcal{Q}_2$ are computed.

Important additional routines, called by the main solvers above are listed below.

MB04CD - computes the orthogonal matrices $\mathcal{Q}_1$, $\mathcal{Q}_2$ and $\mathcal{Q}_3$ for a real $2m \times 2m$ regular pencil

$$\begin{aligned} \alpha\mathcal{A}\mathcal{B} - \beta\mathcal{D} &= \alpha \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} B_{11} & 0 \\ 0 & B_{22} \end{bmatrix} \\ &\quad - \beta \begin{bmatrix} 0 & D_{12} \\ D_{21} & 0 \end{bmatrix}, \end{aligned} \quad (2)$$

where A_{11} , A_{22} , B_{11} , B_{22} and D_{12} are upper triangular and the generalized matrix product $A_{11}^{-1}D_{12} \times B_{22}^{-1}A_{22}^{-1}D_{21}B_{11}^{-1}$ is upper quasi-triangular, such that $\mathcal{Q}_3^T \mathcal{A} \mathcal{Q}_2$ and $\mathcal{Q}_2^T \mathcal{B} \mathcal{Q}_1$ are upper triangular, and $\mathcal{Q}_3^T \mathcal{D} \mathcal{Q}_1$ is upper quasi-triangular.

MB04HD - computes the orthogonal matrices $\mathcal{Q}_1$ and $\mathcal{Q}_2$ for a real $2m \times 2m$ regular pencil

$$\alpha\mathcal{A} - \beta\mathcal{B} = \alpha \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} - \beta \begin{bmatrix} 0 & B_{12} \\ B_{21} & 0 \end{bmatrix}, \quad (3)$$

where A_{11} , A_{22} , and B_{12} are upper triangular and the generalized matrix product $A_{11}^{-1}B_{12}A_{22}^{-1}B_{21}$ is upper quasi-triangular.

MB03BD - computes the eigenvalues of the generalized formal matrix product $A = A_1^{s_1} A_2^{s_2} \dots A_k^{s_k}$, where $s_j = 1$ or $s_j = -1$, $j = 1, \dots, k$, A_h is upper Hessenberg and A_i , $i \neq h$, is upper triangular (*upper Hessenberg/triangular form*), using a double-shift version of the periodic QZ algorithm [19], [20]. In addition, A may be reduced to *periodic Schur form*: A_h is upper quasi-triangular and all the other factors A_i are upper triangular. Optionally, the 2×2 triangular matrices corresponding to 2×2 diagonal blocks in A_h are reduced so that their product is a 2×2 diagonal matrix. Optionally, the orthogonal factors Q_j used, $j = 1, \dots, k$, are accumulated.

MB03CD - computes orthogonal matrices $\mathcal{Q}_1$, $\mathcal{Q}_2$, $\mathcal{Q}_3$ for a real 2×2 , 3×3 , or 4×4 regular block upper triangular pencil

$$\begin{aligned} \alpha\mathcal{A}\mathcal{B} - \beta\mathcal{D} &= \alpha \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{bmatrix} \\ &\quad - \beta \begin{bmatrix} D_{11} & D_{12} \\ 0 & D_{22} \end{bmatrix}, \end{aligned}$$

such that the pencil $\alpha(\mathcal{Q}_3^T \mathcal{A} \mathcal{Q}_2)(\mathcal{Q}_2^T \mathcal{B} \mathcal{Q}_1) - \beta(\mathcal{Q}_3^T \mathcal{D} \mathcal{Q}_1)$ is still in block upper triangular form, but the eigenvalues in $\Lambda(A_{11}B_{11}, D_{11})$ and $\Lambda(A_{22}B_{22}, D_{22})$ are exchanged, where $\Lambda(X, Y)$ denotes the spectrum of the matrix pencil (X, Y) .

MB03DD - computes orthogonal matrices $\mathcal{Q}_1$ and $\mathcal{Q}_2$ for a real 2×2 , 3×3 , or 4×4 regular block upper triangular pencil

$$\alpha\mathcal{A} - \beta\mathcal{B} = \alpha \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} - \beta \begin{bmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{bmatrix},$$

such that the pencil $\alpha(\mathcal{Q}_2^T \mathcal{A} \mathcal{Q}_1) - \beta(\mathcal{Q}_2^T \mathcal{B} \mathcal{Q}_1)$ is still in block upper triangular form, but the

eigenvalues in $\Lambda(A_{11}, B_{11})$ and $\Lambda(A_{22}, B_{22})$ are exchanged.

MB03ED - computes orthogonal matrices Q_1 , Q_2 , and Q_3 for a real 2×2 or 4×4 regular pencil $\alpha AB - \beta D$, with matrices in the form of the right hand side of (2), such that $Q_3^T A Q_2$ and $Q_2^T B Q_1$ are upper triangular, $Q_3^T D Q_1$ is upper quasi-triangular, and the eigenvalues with negative real parts (if there are any) are allocated on the top. The submatrices A_{11} , A_{22} , B_{11} , B_{22} and D_{12} are upper triangular. If D_{21} is 2×2 , then all other blocks are nonsingular and the product $A_{22}^{-1} D_{21} B_{11}^{-1} A_{11}^{-1} D_{12} B_{22}^{-1}$ has a pair of complex conjugate eigenvalues.

MB03FD - computes orthogonal matrices Q_1 and Q_2 for a real 2×2 or 4×4 regular pencil $\alpha A - \beta B$, with matrices in the form of the right hand side of (3), such that $Q_2^T A Q_1$ is upper triangular, $Q_2^T B Q_1$ is upper quasi-triangular, and the eigenvalues with negative real parts (if there are any) are allocated on the top. The submatrices A_{11} , A_{22} , and B_{12} are upper triangular. If B_{21} is 2×2 , then all the other blocks are nonsingular and the product $A_{11}^{-1} B_{12} A_{22}^{-1} B_{21}$ has a pair of complex conjugate eigenvalues.

MB03GD - computes an orthogonal matrix Q and an orthogonal symplectic matrix U for a real regular 2×2 or 4×4 skew-Hamiltonian/Hamiltonian pencil $\alpha \mathcal{J} B^T \mathcal{J}^T B - \beta D$ with

$$B = \begin{bmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{bmatrix}, \quad D = \begin{bmatrix} D_{11} & D_{12} \\ 0 & -D_{11}^T \end{bmatrix},$$

such that $\mathcal{J} Q^T \mathcal{J}^T D Q$ and $U^T B Q$ keep block triangular form, but the eigenvalues are exchanged.

MB03HD - computes an orthogonal matrix Q , for a real regular 2×2 or 4×4 skew-Hamiltonian/Hamiltonian pencil

$$\alpha A - \beta B = \alpha \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{11}^T \end{bmatrix} - \beta \begin{bmatrix} B_{11} & B_{12} \\ 0 & -B_{11}^T \end{bmatrix}$$

in structured Schur form, such that $\mathcal{J} Q^T \mathcal{J}^T (\alpha A - \beta B) Q$ is still in structured Schur form, but the eigenvalues are exchanged.

MB03ID - moves the stable eigenvalues of a $2m \times 2m$ real skew-Hamiltonian/Hamiltonian pencil $\alpha \mathcal{S} - \beta \mathcal{H}$ in structured Schur form, with $\mathcal{S} = \mathcal{J} \mathcal{Z}^T \mathcal{J}^T \mathcal{Z}$,

$$\mathcal{Z} = \begin{bmatrix} A & D \\ 0 & C \end{bmatrix}, \quad \mathcal{H} = \begin{bmatrix} B & F \\ 0 & -B^T \end{bmatrix},$$

to the leading principal subpencil, while keeping the triangular form. Above, A is upper triangular, B is upper quasi-triangular, and C is lower triangular. The matrices $\mathcal{Z}$ and $\mathcal{H}$ are transformed by an orthogonal symplectic matrix $\mathcal{U}$ and an orthogonal matrix Q to $\tilde{\mathcal{Z}} := \mathcal{U}^T \mathcal{Z} Q$, $\tilde{\mathcal{H}} := \mathcal{J} Q^T \mathcal{J}^T \mathcal{H} Q$,

such that

$$\tilde{\mathcal{Z}} = \begin{bmatrix} \tilde{A} & \tilde{D} \\ 0 & \tilde{C} \end{bmatrix}, \quad \tilde{\mathcal{H}} = \begin{bmatrix} \tilde{B} & \tilde{F} \\ 0 & -\tilde{B}^T \end{bmatrix},$$

where $\tilde{A}$, $\tilde{B}$ and $\tilde{C}$ remain in (quasi-)triangular form.

MB03JD - moves the stable eigenvalues of a $2m \times 2m$ real skew-Hamiltonian/Hamiltonian pencil $\alpha \mathcal{S} - \beta \mathcal{H}$ in structured Schur form,

$$\mathcal{S} = \begin{bmatrix} A & D \\ 0 & A^T \end{bmatrix}, \quad \mathcal{H} = \begin{bmatrix} B & F \\ 0 & -B^T \end{bmatrix},$$

with A upper triangular and B upper quasi-triangular, to the leading principal subpencil, while keeping the triangular form:

$$\tilde{\mathcal{S}} = \begin{bmatrix} \tilde{A} & \tilde{D} \\ 0 & \tilde{A}^T \end{bmatrix}, \quad \tilde{\mathcal{H}} = \begin{bmatrix} \tilde{B} & \tilde{F} \\ 0 & -\tilde{B}^T \end{bmatrix},$$

where $\tilde{A}$ and $\tilde{B}$ preserve the forms of A and B .

MB03KD - reorders the eigenvalues of the generalized formal matrix product [21], [22]

$$(T_{22}^{(K)})^{s_K} (T_{22}^{(K-1)})^{s_{K-1}} \dots (T_{22}^{(1)})^{s_1},$$

in the generalized periodic real Schur form,

$$T_k = \begin{bmatrix} T_{11}^{(k)} & T_{12}^{(k)} & T_{13}^{(k)} \\ 0 & T_{22}^{(k)} & T_{23}^{(k)} \\ 0 & 0 & T_{33}^{(k)} \end{bmatrix}, \quad k = 1, \dots, K,$$

such that the m selected eigenvalues pointed to by a logical vector `select` end up in the leading part of the matrix sequence $T_{22}^{(k)}$, where

- the matrices T_k are $n_{k+1} \times n_k$, if $s_k = 1$, or $n_k \times n_{k+1}$, if $s_k = -1$,
- the submatrices $T_{11}^{(k)}$ are $\tilde{n}_{k+1} \times \tilde{n}_k$, if $s_k = 1$, or $\tilde{n}_k \times \tilde{n}_{k+1}$, if $s_k = -1$, and contain dimension-induced infinite eigenvalues,
- the submatrices $T_{22}^{(k)}$ are $\hat{n} \times \hat{n}$ and contain core eigenvalues, which are generically neither zero nor infinite,
- the submatrices $T_{33}^{(k)}$ contain dimension-induced zero eigenvalues.

Given that $n_k = n_{k+1}$ for all k where $s_k = -1$, and the $T_{11}^{(k)}$ are void, then the first m columns of the updated orthogonal transformation matrix sequence $Q_1, \dots, Q_K$ span a periodic deflating subspace corresponding to the same eigenvalues.

The new routines for computing the eigenvalues of skew-Hamiltonian/Hamiltonian pencils allow to preserve the spectra symmetry, at least for problems with simple eigenvalues on the imaginary axis. This is important in some applications, such as evaluating the L_∞ - or H_∞ -norm of a continuous-time, standard or descriptor linear time-invariant multivariable system. Quadratically convergent algorithms [23], [24] for the computation of these norms use the purely imaginary eigenvalues of a Hamiltonian or symplectic matrix or matrix

pencil at each iteration. (Actually, the pencils arising in the continuous-time case are skew-Hamiltonian/Hamiltonian.) The detection of purely imaginary eigenvalues is a delicate numerical problem if an unstructured algorithm is used. The MATLAB Control Systems Toolbox function `norm` uses a certain strategy for detecting such eigenvalues, to cover their possibly wrong movement off the imaginary axis, when using the MATLAB function `eig`. An eigenvalue λ is considered to be purely imaginary if the modulus of its real part is less than $10\sqrt{\varepsilon}(1+|\lambda|)+100\varepsilon\|\Lambda\|_\infty$, where Λ is the set of eigenvalues of the current pencil, and ε is the relative machine precision.

A simpler strategy is used in some of the new routines for solving eigenproblems with skew-Hamiltonian/Hamiltonian matrix pencils. Examples have been found for which the returned eigenvalues had tiny real or imaginary parts, when these should theoretically be 0, according to the pairing requirements. The routines use ε as a relative tolerance; specifically, the real (or imaginary) part of an eigenvalue λ is set to 0 if its modulus is smaller than $\varepsilon|\lambda|$.

Some benchmarking for the skew-Hamiltonian/Hamiltonian solvers has been performed on a portable Intel Dual Core computer at 2 GHz, 2 GB RAM, and $\varepsilon \approx 2.22 \times 10^{-16}$, using Windows XP (Service Pack 2) operating system, Intel Visual Fortran 11.1 compiler, and MATLAB 7.11.0.584 (R2010b). The real, unfactored case can be significantly (between 1.2 to 1.7 times) faster than state-of-the-art standard MATLAB solver `eig` for computing the eigenvalues for problems with order less than or equal to 1600. When also computing the stable deflating subspace, currently there is no gain (even a gain loss for small order problems), due to the increased complexity of the codes. Also, the factored versions (where one matrix is given as $\mathcal{J}\mathcal{Z}^T\mathcal{J}^T\mathcal{Z}$) are slower than `eig`. (These versions do not multiply the factors, but work on $\mathcal{Z}$ directly.) Details are given in [16], [17], [18]. The work will continue trying to improve the speed.

There are already complex versions of the skew-Hamiltonian/Hamiltonian solvers, but additional efforts are needed in order to improve the calculations, especially for deflating subspaces. The codes will be available at the next SLICOT Library update.

Besides the new codes, the logic for updating the column/row norms has been replaced in nine routines as suggested in [25], to avoid a possibly erratic behavior. Such a behavior is also common to similar LAPACK routines.

B. New MEX-files

Based on the new main routines, the following MEX-files have been developed:

`skewHamildefl` - computes the eigenvalues of a skew-Hamiltonian/Hamiltonian pencil and an orthonormal basis of the right deflating subspace corresponding to the stable eigenvalues.

`skewHamildeflf` - computes the eigenvalues of a skew-Hamiltonian/Hamiltonian pencil in factored form and orthonormal bases of the right deflating subspace and companion subspace corresponding to the stable eigenvalues.

`skewHamileig` - computes the eigenvalues and orthogonal decomposition of a skew-Hamiltonian/Hamiltonian pencil.

`sympLURV` - computes the eigenvalues and generalized symplectic URV decomposition of a skew-Hamiltonian/Hamiltonian pencil in factored form.

The new skew-Hamiltonian/Hamiltonian codes might be used in the structured H_∞ solvers from the MATLAB Robust Control Toolbox (e.g., in the new `hinfstruct` function).

To illustrate the usefulness of the new codes, Fig. 1 [16] shows a comparison between the eigenvalues computed by the factored version of the structured algorithm (using `sympLURV`) and `eig` for an example of order 4 ($m = 2$). This example has only purely imaginary eigenvalues, but two eigenvalues computed by `eig` have real parts of order 10^{-11} , which might not be considered as negligible.

Moreover, Fig. 2 presents the relative residuals of the structured CARE solver, based on `skewHamildefl`, and the MATLAB function `care`, for 34 test problems from the SLICOT CARE benchmark collection. Most of them are difficult numerical examples. Details are given in [16], but improved results are shown here. Two alternative options have been used for orthogonalizing the subspace basis—QR factorization with column pivoting (QRP, for short), and singular value decomposition (SVD). The results for the structured solver are comparable to, and sometimes better than those produced by `care`. Test example # 17 still needs further investigation.

III. FURTHER SLICOT DEVELOPMENTS

Future developments of the SLICOT Library and related tools concern both improvements and extensions. Some issues are summarized below.

1) *Improvements of stable deflating subspace computation:* There are some difficulties related to the deflating subspace computations, which appeared when trying to solve certain Riccati equations from the SLICOT benchmark collection. The solvers work on extended pencils which doubled all finite eigenvalues of the original pencil. The theoretical

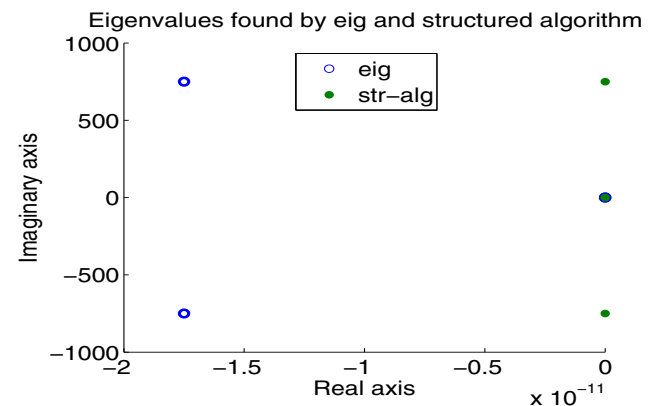


Fig. 1. Eigenvalue scatter plot for an example of order 4. Two eigenvalues computed by `eig` are close between them and close to the corresponding eigenvalues computed by the structured solver.

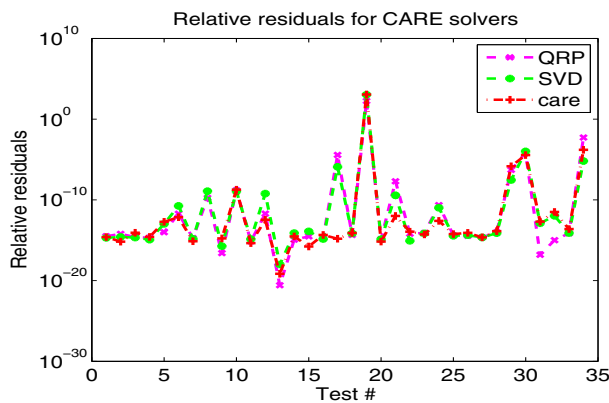


Fig. 2. Relative residuals of CARE solvers for CARE benchmark examples.

algorithms deliver a subspace basis with $2p$ columns, where p is the number of stable eigenvalues of the original pencil. But what is needed is a basis of the deflating subspaces for the original pencil (with just p columns), and this is not always easy to find numerically.

2) *Faster versions for large, dense matrices:* A new version of the MB04BD routine has been implemented, which is more efficient for a large order of the pencil. This version works on panels of columns. Briefly, the plane rotations are stored in the workspace, and applied on columns immediately after being generated, but for row transformations, the saved rotations are applied just on a panel of columns (including those used for generating the last rotations). The idea is to limit the application of the row transformations to a number of columns which might fit in the cache memory. The speed improvement for problems with large order is shown in [18]. The speedup is larger when computing just the eigenvalues. The improvement for subspace basis computation is minor, although the reduction of the CPU time is the same as for the eigenvalue calculations. Similar improvements can be made in other called routines. For problems of small order, the new version of MB04BD can be slower than the old one. Since many users do not typically solve large order problems, the SLICOT Library should offer two routines. It is known that the LAPACK package often offers pairs of routines: an unblocked one and a blocked one, which calls the unblocked routine when needed. The same can be done in SLICOT.

3) *Changes for thread safe computations:* Another change to be done in the future is to replace the calls of the LAPACK routine DLAON, which contains a SAVE statement, by calls to DLACN2; the latter has an auxiliary last input/output argument, an INTEGER array of dimension 3, used to save the needed values of the variables between successive calls to DLACN2.

4) *Setting optimal workspace size:* Older SLICOT routines do not allow to estimate the optimal workspace size in advance, but just return an “a posteriori” value (after completing the calculations). That value might be used for repeated solution of problems of the same size. However, knowing the optimal workspace size before starting the com-

putations can significantly improve the efficiency. Therefore, new SLICOT routines offer a LAPACK-style device for querying the optimal workspace. Actually, SLICOT evolved in parallel with the LAPACK package. The first device implemented in LAPACK for getting the optimal workspace was the use of the environmental routine ILAENV, assuming that ILAENV is adapted for the computer being actually used. Then, the LAPACK developers added an option based on LWORK = -1, where LWORK defines the workspace size, when positive. Its use involves two calls of the routine for which the query is needed. The first call with LWORK = -1 just returns the optimal workspace size, which can be used in the second, computational call. In many cases, the optimal workspace size is computed using also calls to ILAENV. The SLICOT routines used either calls to ILAENV, or calls with LWORK set to -1. But for some implementations (e.g., using the Intel MKL optimized library), the workspace size computed by ILAENV can be smaller than that obtained using LWORK = -1, because the first approach does not take into account additional parameters which affect performance, like CPU type, number of threads, special optimizations available in MKL for LAPACK functions depending on problem type/size, etc., which are considered by the second approach. Therefore, it will be useful to replace in the SLICOT routines the calls of ILAENV by calls with the workspace size parameter set to -1.

5) *Use of real matrices in complex computations:* For complex calculations, the MEX-files should allow that (part of) the matrices be real. Currently, just polezeroz has been modified in this respect. (The original polezeroz assumed that all matrices are complex.)

6) *Avoiding ALLOCATE statements in MEX-files:* The use of ALLOCATE statements in MEX-files is not safe in combination with MKL (and possibly with other optimized libraries). Specifically, many repeated executions of some MEX-files (involving, e.g., eigenvalue computations), with the same data, will not necessarily return exactly the same results. The use of calls to MATLAB function mxCallLoc solves this problem. For instance, a test has been performed in which an executable MEX-file for the computation of eigenvalues and of a basis for the stable deflating subspace of a real skew-Hamiltonian/Hamiltonian matrix pencil has been called repeatedly with some options omitted or set to their default values, and the returned values were compared. This strategy should produce identical results. However, even in a set of 10 repeated runs of the MEX-file which used ALLOCATE statements, the theoretical null errors have not been achieved. A similar behaviour has been observed for other MEX-files. This shows that the use of mxCallLoc is preferable instead of ALLOCATE.

Some MEX-files have been modified, but not all. The ALLOCATE statements have been replaced by mxCallLoc calls for arrays (except for INTEGER or LOGICAL arrays), in order to ensure full reproducibility of the results when using optimized LAPACK and BLAS libraries. The use of mxCallLoc implied additional changes of all references to the involved arrays. Specifically, a reference like A was

replaced by `%VAL( A )`. The modified MEX-files have been also updated for being portable on 64-bit platforms.

7) *Extensions*: Possible future additions to the SLICOT Library are routines for, e.g., the structured staircase form of (skew-)symmetric/(skew-)symmetric pencils, or additional functionality for descriptor systems.

For the L_∞ -norm computation, a new version will be implemented for the discrete-time case, via Cayley-transformed pencils which are further transformed to skew-Hamiltonian/Hamiltonian pencils [26].

Another future development can be a version of SLICOT for complex matrices. Developing a full complex version would require, however, a significant amount of work, since some parts need a lot of changes. But it is reasonable to expect producing the versions for the main needed functionality.

IV. CONCLUSIONS

The latest developments of the SLICOT Library and associated tools are discussed, and issues related to its future evolution are summarized. Structure-exploiting solvers for skew-Hamiltonian/Hamiltonian real matrix pencils have been recently added. Possible applications include the computation of L_∞ - or H_∞ -norms of linear time-invariant multivariable systems, and solution of algebraic matrix Riccati equations. The calculation of the eigenvalues of such pencils usually outperforms the standard QZ algorithm, while guaranteeing the symmetry of the spectra.

V. ACKNOWLEDGMENTS

The collaboration with Peter Benner and Matthias Voigt from Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany, is highly acknowledged.

REFERENCES

- [1] *NAG SLICOT Library Manual, Release 1*, The Numerical Algorithms Group, Wilkinson House, Jordan Hill Road, Oxford OX2 8DR, U.K., 1991.
- [2] *NAG SLICOT Library Manual, Release 2*, The Numerical Algorithms Group, Wilkinson House, Jordan Hill Road, Oxford OX2 8DR, U.K., 1993, updates Release 1 of May 1990.
- [3] P. Benner, V. Mehrmann, V. Sima, S. Van Huffel, and A. Varga, "SLICOT — A subroutine library in systems and control theory," in *Applied and Computational Control, Signals, and Circuits*, B. N. Datta, Ed. Birkhäuser, Boston, 1999, vol. 1, chapter 10, pp. 499–539.
- [4] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide: Third Edition*. Philadelphia: SIAM, 1999.
- [5] C. L. Lawson, R. J. Hanson, D. R. Kincaid, and F. T. Krogh, "Basic Linear Algebra Subprograms for Fortran usage," *ACM Trans. Math. Softw.*, vol. 5, no. 3, pp. 308–323, 1979.
- [6] J. J. Dongarra, J. Du Croz, S. Hammarling, and R. J. Hanson, "Algorithm 656: An extended set of Fortran Basic Linear Algebra Subprograms," *ACM Trans. Math. Softw.*, vol. 14, no. 1, pp. 1–17, 18–32, 1988.
- [7] J. J. Dongarra, J. Du Croz, I. S. Duff, and S. Hammarling, "Algorithm 679: A set of Level 3 Basic Linear Algebra Subprograms," *ACM Trans. Math. Softw.*, vol. 16, no. 1, pp. 1–17, 18–28, 1990.
- [8] V. Sima and S. Van Huffel, "High-performance algorithms and software for systems and control computations," in *Proceedings of the 1999 IEEE International Conference on Control Applications and IEEE International Symposium on Computer Aided Control System Design*, August 22–27, 1999, Hapuna-Beach Prince Hotel, Kohala Coast-Island of Hawai'i, Hawai'i, USA. Omnipress, 1999, pp. 85–90.
- [9] S. Van Huffel and V. Sima, "SLICOT and control systems numerical software packages," in *Proceedings of the 2002 IEEE International Conference on Control Applications and IEEE International Symposium on Computer Aided Control System Design, CCA/CACSD 2002, September 18–20, 2002, Scottish Exhibition and Conference Centre, Glasgow, Scotland, U.K.* Omnipress, 2002, pp. 39–44.
- [10] V. Sima, "SLICOT-based advanced automatic control computations," in *Advances in Automatic Control*, The Kluwer International Series in Engineering and Computer Science, M. Voicu, Ed. Boston/Dordrecht/New York/London: Kluwer Academic Publishers, 2004, pp. 337–349.
- [11] S. Van Huffel, V. Sima, A. Varga, S. Hammarling, and F. Delebecque, "High-performance numerical software for control," *IEEE Control Syst. Mag.*, vol. 24, no. 1, pp. 60–76, Feb. 2004.
- [12] P. Benner, D. Kressner, V. Sima, and A. Varga, "Die SLICOT-Toolboxen für Matlab," *at—Automatisierungstechnik*, vol. 58, no. 1, pp. 15–25, Jan. 2010.
- [13] P. Benner, R. Byers, V. Mehrmann, and H. Xu, "Numerical computation of deflating subspaces of skew Hamiltonian/Hamiltonian pencils," *SIAM J. Matrix Anal. Appl.*, vol. 24, no. 1, pp. 165–190, 2002.
- [14] P. Benner, R. Byers, P. Losse, V. Mehrmann, and H. Xu, "Numerical solution of real skew-Hamiltonian/Hamiltonian eigenproblems," Technische Universität Chemnitz, Chemnitz, Tech. Rep., Nov. 2007.
- [15] V. Sima, "Structure-preserving computation of stable deflating subspaces," in *Proceedings of the 10th IFAC Workshop "Adaptation and Learning in Control and Signal Processing" (ALCOSP 2010), Antalya, Turkey, 26–28 August 2010 (CD-ROM)*, E. Kayacan, Ed. Kudret Press & Digital Printing Co., Istanbul, Turkey, 2010, 6 pages. [Online]. Available: <http://www.ifacworkshops.org>
- [16] —, "Computational experience with structure-preserving Hamiltonian solvers in optimal control," in *Proceedings of the "8th International Conference on Informatics in Control, Automation and Robotics" (ICINCO 2011), Noordwijkerhout, The Netherlands, 28–31 July, 2011*. Vol. 1, SciTePress, Portugal, pp. 91–96, 2011.
- [17] —, "Computational experience with structure-preserving Hamiltonian solvers in complex spaces," in *Proceedings of the Physcon 2011 — 5th International Scientific Conference on Physics and Control, León, Spain, September 5–8, 2011*. To appear.
- [18] —, "Performance of skew-Hamiltonian/Hamiltonian solvers for automatic control computations," in *Proceedings. VI-th Edition of the International Conference "ASTR 2011 Academic Days". Timișoara, September 22–23, 2011*. To appear.
- [19] A. W. Bojanczyk, G. Golub, and P. Van Dooren, "The periodic Schur decomposition: algorithms and applications," in *Proc. of the SPIE Conference Advanced Signal Processing Algorithms, Architectures, and Implementations III*, F. T. Luk, Ed., vol. 1770, Nov. 1992, pp. 31–42.
- [20] J. Sreedhar and P. Van Dooren, "Periodic Schur form and some matrix equations," in *Systems and Networks: Mathematical Theory and Applications, Proceedings of the Symposium on the Mathematical Theory of Networks and Systems (MTNS'93), Regensburg, Germany, 2–6 August 1993*, U. Helmke, R. Mennicken, and J. Saurer, Eds., vol. 1. John Wiley & Sons, 1994, pp. 339–362.
- [21] R. Granat, B. Kågström, and D. Kressner, "Computing periodic deflating subspaces associated with a specified set of eigenvalues," *BIT Numerical Mathematics*, vol. 47, pp. 763–791, 2007.
- [22] —, "MATLAB tools for solving periodic eigenvalue problems," in *Third IFAC Workshop on Periodic Control Systems*, August, 29–31, 2007, Saint Petersburg, Russia (CD-ROM), G. Leonov and A. Fradkov, Eds., 2007.
- [23] S. Boyd, V. Balakrishnan, and P. Kabamba, "A bisection method for computing the H_∞ norm of a transfer matrix and related problems," *Mathematics of Control, Signals, and Systems*, vol. 2, no. 3, pp. 207–219, Sept. 1989.
- [24] N. A. Bruinsma and M. Steinbuch, "A fast algorithm to compute the H_∞ -norm of a transfer function," *Systems Control Lett.*, vol. 14, no. 4, pp. 287–293, Apr. 1990.
- [25] Z. Bujanović and Z. Drmač, "How a numerical rank revealing instability affects Computer Aided Control System Design," SLICOT Working Note 2010-1, Jan. 2010. [Online]. Available: <http://www.slicot.org/REPORTS/SLWN2010-1.pdf>.
- [26] H. Xu, "On equivalence of pencils from discrete-time and continuous-time control," *Lin. Alg. Appl.*, vol. 414, no. 1, pp. 97–124, Apr. 2006.

Sampling step importance in hierarchical semantic segmentation of microscopic images

Cristian Smochina, Vasile Manta, and Walter Kropatsch

Abstract— The objective of semantic segmentation in microscopic images is to extract the cellular, nuclear or tissue components. This problem is challenging due to the large variations of these components features (size, shape, orientation or texture). In this paper we improve the technique presented in [17] used to identify the epithelial nuclei (crypt) against interstitial nuclei in microscopic images taken from colon tissues. In the proposed enhanced approach, the crypt inner boundary is detected using the closing morphological pyramid instead of morphological hierarchy. The outer crypt border is determined by the epithelial nuclei, overlapped by the maximal isoline of the inner boundary. The use of sampling in building the pyramid offers computational efficiency, reduces the amount of used memory, increase the robustness and preserve the quality results. An analysis of the two approaches is performed considering the number of pixels processed to create each level. Also the relation between the levels of the hierarchical structures is established.

I. INTRODUCTION

IN diagnostic pathology, the pathologists give a diagnostic after a set of biological samples (tissues stained with different markers) are viewed and many specific features of the objects of interest (size, shape, colour or texture) have been analysed. This complex diagnostic process is an important part in clinical medicine but also in biomedical research and can be enhanced by providing the pathologists or the biologists with quantitative data extracted from the images. The image processing techniques are of special interest because they allow large scale statistical evaluation in addition to classical eye screening evaluation and are used in both sections of the pathology: cytology (the study of cells) and histology (anatomical study of the microscopic structure of tissues) [1]. The microscopic image segmentation is considered a hard task due to problems [2] like: low contrast and weak boundaries on out-of-focus nuclei, different grey values for the background caused by the non-

uniform illumination, lower intensities within nuclei caused by the non-uniform distribution of material inside the nucleus and considerable variation of object features like shape and/or size and/or orientation and different nuclei distribution within the epithelial layer.

This paper is organized as follows. The last part of this section points out the goal of this study. Section II presents a brief description of the morphological hierarchy [17] used for inner boundaries detection. The main contributions of this paper are the more efficient approach based on the morphological pyramid and the analysis of the computation time and relationship between these two hierarchical structures (section III). The outer borders are detected using the maximal isoline (section IV). The results are evaluated and discussed in section V and concluded in section VI.

A. State of the Art

Many studies from the literature cover fields like microscopy, biomedical engineering and imaging, bioinformatics or pattern recognition and introduce techniques for solving the mentioned problems [2]. In [3] a threshold is used to identify the gland seeds which are grown to obtain the nuclei chain. In [4] the pixel labelling to different classes is performed using a clustering approach based on the textural properties. An object-graphs approach is described in [5] where the relationship between the primitive objects (nucleus and lumen) is considered. The prostate cancer malignancy is automatically graded (Gleason system) in [6] after the prostate glands are detected.

The multiresolution analysis plays an important role in medical image processing. In [7] the mammographic images are hierarchically decomposed into different resolutions and segmented by analyzing the coarser resolutions while in [8] the multiresolution wavelet analysis is used for texture classification. A Gaussian multiresolution segmentation technique is combined with the expectation maximization (EM) algorithm to overcome the drawbacks of the EM algorithm [9].

B. Aim of the Study

The basic functional unit of the small intestine is the crypt (crypt of Lieberkühn) [10] and it comprises two main structures of interest: the lumen and the epithelial layer (Fig.1.a). The epithelial layer contains epithelial nuclei and surrounds the lumen which is an empty area. The interstitial cells on the other side form heterogeneous regions (stroma)

Manuscript received June 18, 2011. This work was supported in part by the "BRAIN - An Investment in Intelligence" doctoral Scholarship Program, within the Technical University of Iasi, Romania and by the Austrian Science Fund under grant no. P20134-N13.

Cristian Smochina is with the Department of Computer Science and Engineering, "Gheorghe Asachi" Technical University of Iasi, Romania (e-mail: smochina.cristian@yahoo.com).

Vasile Manta is with the Department of Computer Science and Engineering, "Gheorghe Asachi" Technical University of Iasi, Romania (e-mail: vmanta@cs.tuiasi.ro).

Walter Kropatsch is with the Pattern Recognition and Image Processing Group, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Austria (e-mail: krw@prip.tuwien.ac.at).

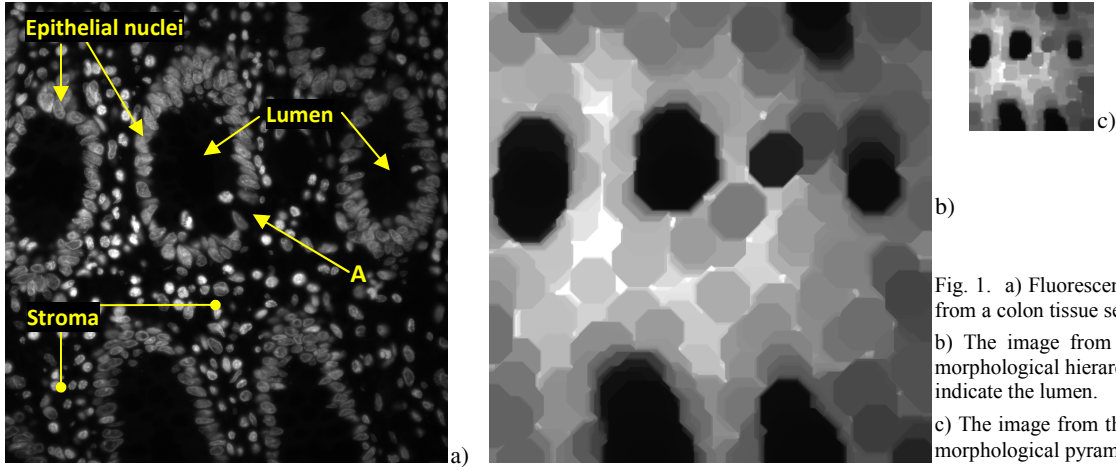


Fig. 1. a) Fluorescence image with crypts from a colon tissue section.
b) The image from the top level of the morphological hierarchy; the black regions indicate the lumen.
c) The image from the critical level of the morphological pyramid.

placed between crypts. The stroma areas (Fig.1.a) contain isolated cells with non-regular shape and without particular patterns of arrangement.

This work provides an enhancement technique to segment the crypts from fluorescence images of colorectal cancer tissue sections. We used 8 bit greyscale images (Fig.1.a) containing nuclei labelled with DAPI, a fluorescent stain that binds strongly to DNA [11] and acquired using a TissueFAXS slide scanner (TissueGnostics GmbH, Austria).

The main motivation for segmenting crypts is to provide the pathologists with quantitative data regarding the areas covered by epithelial nuclei. Our objective is to find the boundaries of the crypts, i.e. to delineate the area covered by the epithelial nuclei without dealing with the individual nuclei (Fig. 1.a).

II. LUMENS SEGMENTATION USING THE MORPHOLOGICAL HIERARCHY

Without considering the relations between the crypt components (epithelial layer and lumen), the stroma and the background, the low level cues will not be able to separate the regions having a particular meaning [12]. A way must be found to keep only the important information and to remove the unnecessary details. In order to detect these regions, the role of *local* information (pixel grey values or gradient) is very important but not sufficient; also *global* information like the region's size and relation with the other region types must be included [12].

A rough assumption about the nuclei distribution over different region types can be made. The lumen does not contain nuclei and appears like a big round black area surrounded by a 'ring' with variable thickness. This ring contains a high density of touched/chained epithelial cells. The exceptional cases appear when the lumen gets to be in touch with the stroma area due to missing cells that 'break' this ring (region A from Fig. 1.a).

By applying the morphological closing operation [13] on the grey image, the nuclei closer than the size of the structure element (SE) will be connected. The epithelial

nuclei and those from the stroma area can be connected by relating the SE's size to the size of the lumen region (the connection should not pass over the lumen).

The last part of this section roughly presents the process of building the hierarchical image decomposition (without sub-sampling); more details can be found in [17].

A. Building the Morphological Hierarchies

Let I denote the input grey scale image and $\bullet$ denote the morphological closing operation. The SE ψ_k is a two-dimensional disk of diameter $2k+1$. The hierarchical morphological representation H consists of L_H levels and each level $\ell > 0$ is given by:

$$H^\ell = H^1 \bullet \psi_{2^\ell}, \ell = \overline{1, L_H}, H^1 = I \quad (1)$$

Since the size of the SE increases according to the level of the hierarchy, in the lower levels only the small gaps will be filled, while the bigger ones will be closed in the upper levels. The top level is established by limiting the SE's size so that it covers maximally 2-3 nuclei (in our experiments the SE ψ_{50} gives 25 levels). The lumens should 'survive' till the top level (Fig.1.b) but the gaps from the crypts should be filled.

B. Lumen Reconstruction

The proper reconstruction of each lumen based on the found regions from the top level must be done by analyzing the lower levels of the hierarchy where more details are present. A binary hierarchy H_{bw} (L_H levels) is built such that each level l is given by

$$H_{bw}^l = H^l < c \cdot thr_{Otsu}(H^l), l = \overline{1, L_H} \quad (2)$$

$thr_{Otsu}(\cdot)$ computes the Otsu's threshold [16] and $0 < c \leq 1$.

Our goal is to find for each partition (ancestor) from the top of Π_{bw} the corresponding partition (descendent) from a bottom level which properly identifies the lumen. Each

level l of the Π_{bw} contains np_l unconnected regions and each one is included in a partition from the below level.

The base level Π_{bw}^1 will not give for sure the proper regions because the global threshold is applied on the original image where the main structures are not properly highlighted. For each partition of the top level $P_r^L, 1 \leq r \leq np_L$ (by analyzing level by level) a certain lower level l_r is chosen such that the descendent region $P_r^{l_r}$ of P_r^1 from l_r properly approximates a lumen region. The selection process checks two rules based on the solidity if the region and the difference between the centroids of two corresponding regions from two consecutive levels.

The border of the found lumens actually describes the inner border of the crypt (Fig.3.a). The false positive (FP) results are eliminated by a validation rule in IV.A.

III. LUMENS SEGMENTATION USING THE MORPHOLOGICAL PYRAMID

In chapter II the morphologic closing operator was applied on the original image to obtain each level of the hierarchy. Since the closing is an idempotent operation (its reapplication produces no further changes to the previously result) [15], the size of the SE grows according to the hierarchy levels so that the bigger gaps to be filled in the higher levels. The partitions obtained by thresholding the top level are analyzed and reconstructed to obtain the lumens regions using the more detailed information from the lower levels (II.B).

A. Motivation

The previous technique provides stable results [17] but presents some solvable problems ($p1$, $p2$ and $p3$). It makes use of a high number of levels (25); each level is actually an image with the same size as the original one. The used memory linearly increases by adding levels to the hierarchy ($p1$). The closing operator applied with increasing SE increases the computation time for each level as moving to the top of the hierarchy ($p2$). The top level 25 takes the longest time to be created since the maximum SE (101x101) is used.

A number of N_H^l pixels are accessed and processed to create the level $l > 1$ and N_H pixels to compute the hierarchy H with L_H levels:

$$N_H^l = w \times h \times (4l)^2 \quad (3)$$

$$N_H = \sum_{l=2}^{L_H} N_H^l = 16 \times w \times h \times \left[\frac{L_H(L_H+1)(2L_H+1)}{6} - 1 \right] \quad (4)$$

where w and h represent the width and the height of the image ($w=1392$ and $h=1024$ in our images). Considering the SE from the top level ($d=101$), more than

14.5 trillion pixels are processed to obtain the top image. In case the size of the images increases, also the computation time needed to obtain each level will increase, especially in the higher levels ($p3$).

Considering these problems ($p1$, $p2$ and $p3$) another approach must be found to deal more easily with a big amount of data, to reduce the computation time and the used memory but in the same time to preserve or even enhance the quality of the results.

In building the morphological hierarchy, the image size remains constant and the SE's size is increasing. A constant SE can be used on the sub-sampled image to reduce the computation time. This problem gives the suggestion of applying the closing operation in a multi-resolution scheme and has been also raised in [18]: "Are the operations equivalent if we decrease the size of the image or if we increase the size of the structuring element?".

A representation structure in which the image size is reduced is the pyramid. The pyramid of an image describes its content at multiple levels of resolution [12]. The type of the pyramid to be chosen is very important for the stability of the final results and must be done according to the application, the type of the image, the properties and relations between images' regions.

The multi-resolution representation has been used with success in image processing and proved their computational efficiency in data representations and processing ([7], [8], [9]). This stack of images with exponentially decreasing resolutions allows easy access to different resolutions of an image making this a good reason to be used in processing high amount of data.

B. Morphological pyramid on DAPI images

In building a regular pyramid on an image, the reduction factor r determines the rate by which the number of cells decreases from level to level; the reduction window associates to a pixel in a higher level a set of pixels in the level directly below. E.g. in a $2 \times 2/4$ pyramid, the number of cells decrease from level to level by a factor of 4 with a window of 2×2 .

Due to the advantages of the morphological closing (II), the pyramid is generated using this operator. One level of the morphological pyramid (a non-linear multi-resolution) [15] is obtained in two steps: the current level is first morphologically filtered with the SE ψ_Π , second the result is sub-sampled to obtain the next level. The input image represents the first level. The morphological pyramid Π consists of L_Π levels and each level $\ell > 0$ is given by:

$$\Pi^\ell = (\Pi^{\ell-1} \bullet \psi_\Pi) \downarrow_4, \ell = \overline{2, L_\Pi}, \Pi^1 = I \quad (5)$$

The $\downarrow_4$ denotes the sub-sampling process which reduce the image size by four times (halving the w and h). To create the level l of the pyramid a number of N_Π^l pixels are

accessed and N_{Π} pixels to compute the entire pyramid with L_{Π} levels.

$$N_{\Pi}^l = \frac{w}{2^{l-1}} \times \frac{h}{2^{l-1}} \times d^2 \quad (6)$$

$$N_{\Pi} = \sum_{l=2}^{L_{\Pi}} N_{\Pi}^l = w \times h \times d^2 \times \left(\frac{1}{2^2} + \frac{1}{2^{2^2}} + \dots + \frac{1}{2^{L_{\Pi}^2}} \right) \quad (7)$$

C. Moving from un-sampled to sampled domain

Fig. 2. shows the difference between the numbers of accessed pixels (NAP) for building the two morphological structures. The NAP decreases exponentially as going to the top levels in the pyramid's case, while for the hierarchy the NAP increases based on a quadratic polynomial.

In our case it is a question if by making use of this computation efficiency the quality of the results would be the same or even similar with the results offered by [17].

The 25th level of the morphological hierarchy gives the proper partitions from which the lumens are reconstructed. Also in case of the pyramid, a suitable resolution level must be selected. From experimental tests resulted that the 5th level ($w=87$, $h=64$) (critical level) is the maximum level of the pyramid which offers the minimum details to extract relevant information.

In order to make the relationship between the morphological closing operation in the sampled (pyramid Π) and in the un-sampled (hierarchy H) domain, the level H^{25} of the hierarchy must be related with the level Π^5 from pyramid; this implies the relation between the SEs ψ_{50} and ψ_{Π} . The unknown ψ_{Π} is obtained by solving the following equation:

$$H^{25} \downarrow_{4^4} \approx \Pi^5 \rightarrow (H^1 \bullet \psi_{50}) \downarrow_{4^4} \approx (\Pi^4 \bullet \psi_{\Pi}) \downarrow_4 \quad (8)$$

The SE ψ_{50} was chosen so that it covers maximally 2-3 nuclei. The same rule must be applied for the 5th level of the pyramid so that it provides the proper partitions from which the lumens are reconstructed. Since the image Π^4 is 4^3 times smaller than the original one, also the size of the SE should be reduced to cover 2-3 'smaller' nuclei.

$$\psi_{\Pi} = \psi_{50} \downarrow_{4^3} = \psi_{50/2^3} \approx \psi_7 \quad (9)$$

From (8) and (9) we derive (10) which relate the level l_H of the hierarchy with the level $l+1$ of the pyramid:

$$(H^1 \bullet \psi_{l_H}) \downarrow_{4^l} \approx (\Pi^l \bullet \psi_{\frac{l_H}{2^{l-1}}}) \downarrow_4 \quad (10)$$

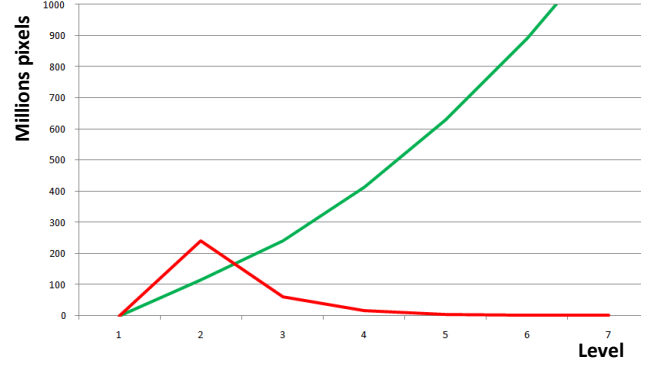


Fig. 2. The number of pixels accessed per level to compute the morphological pyramid (red) and the morphological hierarchy (green).

D. Lumen reconstruction

Similar to II.B, a binary pyramid Π_{bw} is build in which each level represents the result of the thresholding applied on the corresponding level from Π :

$$\Pi_{bw}^l = \Pi^l < c \cdot thr_{Otsu}(\Pi^l), l = \overline{1, L_{\Pi}} \quad (11)$$

In II.B for each partition from H_{bw}^{25} the proper lower level must be found so that the corresponding descendent partition properly identifies the lumen. When using the pyramid, the descendent partitions of the Π_{bw}^5 from the level Π_{bw}^3 properly identify the lumens. The FP results are eliminated by a validation rule in IV.A.

E. Filtering and sampling vs. sampling and filtering

A new level of the pyramid can be obtained using the following two possibilities:

--the current image is filtered with the SE ψ' and then sub-sampled, $o_1: \Pi_1^{l+1} = (\Pi_1^l \bullet \psi') \downarrow_4$ (used in (5)).

--the current image is sub-sampled before filtering with the SE ψ'' , $o_2: \Pi_2^{l+1} = \Pi_2^l \downarrow_4 \bullet \psi''$.

Two questions arise: how different are the two pyramids (Π_1, Π_2) created using the two possibilities if $\psi' = \psi''$ and, if they are different what should be the relation between ψ' and ψ'' such that $\Pi_1^l \approx \Pi_2^l, \forall l$.

The SE ψ' gives the critical level l_c in the case o_1 . Considering the case o_2 and $\psi' = \psi''$, at the level l_c of Π_2 the SE covers 4-5 nuclei instead 2-3 nuclei (in case o_1) since ψ'' is applied on the sub-sampled image. This results in over-filling the important regions like the lumens. The ψ'' is applied (in case o_2) on an image 4 times smaller than the corresponding image from case o_1 :

$$\Pi_1^{l-1} \uparrow_4 \approx \Pi_2^l \quad (12)$$

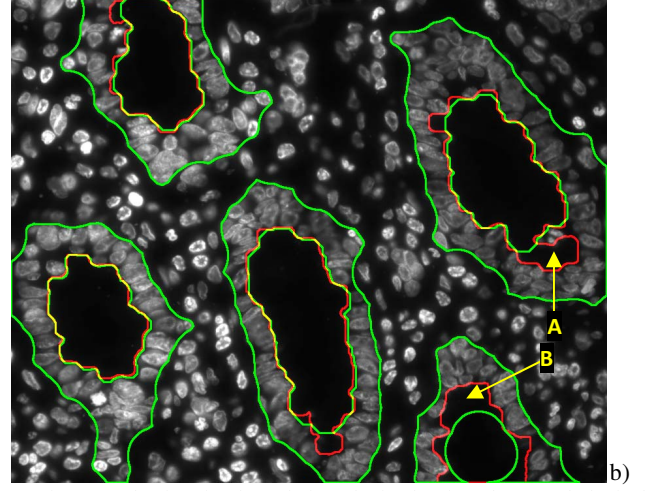
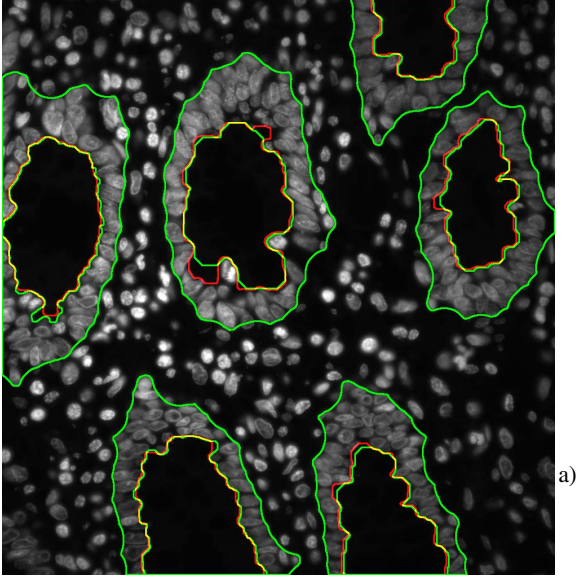


Fig. 3. The interior boundaries obtained using the morphological hierarchy (green curves) and the pyramid (red curves); the results are identical on the yellow portions. The outer borders are depicted with green.

If $\Pi_1^l \approx \Pi_2^l$ is desired, then also the size of SE must be reduced, i.e. $\psi'' = \psi' \downarrow_4$. In this case, the SE ψ'' will cover the same area as the SE ψ' .

IV. CRYPT'S OUTER BORDER

The epithelial layer is differentiated from the stroma areas by considering the nuclei distribution: the crypt's nuclei are packed tightly together while those from the stroma areas are wide spread with considerable distances between them. The isolines of the inner boundary are used to eliminate the FP lumens and to detect the outer border which delineates the epithelial nuclei.

Each isoline contains pixels situated on the same distance from the inner boundary. The maximum distance $d_{\max}$ (60 in our experiments) is related to the average width of the epithelial layer. The maximum value (green square) indicates the isoline which gives the maximum sum of pixels intensities. This maximal isoline is used for two purposes: to validate the detected lumens and to mark the epithelial nuclei (used to get the outer boundary).

A. Lumen validation

The area covered by nuclei can be identified by subtracting a highly blurred version from the original image: $N_{bw} = (I - G * I) > 0$, where G is a big Gaussian filter (201 by 201 in our experiments) and $*$ denotes the convolution operation. Considering the high nuclei concentration around the lumen, there should be only few situations in which the maximal isoline crosses over the background in N_{bw} , i.e. situations of big distances between epithelial nuclei or in case of crypt breaks. Based on this, the following rule is used to validate the lumen results from II.B and 0: if the portions of the maximal isoline overlapping the background in N_{bw} are not considerable high compared to those overlapping the nuclei than the

found boundary does not delimit a lumen area [17]. The FP boundaries are depicted with orange in Fig. 4.

B. Outer border detection

A region from N_{bw} is marked as part of a crypt iff it is overlapped by the band formed by the inner boundary and the maximal isoline. The outer border of the crypt represents the exterior perimeter of the morphological closing applied on the found epithelial nuclei with a SE covering 2-3 nuclei.

V. RESULTS

In Fig. 3 a, b and Fig. 4 the inner boundaries from both approaches are drawn: the green interior curves are obtained using the hierarchical morphological approach (II, detailed in [17]), the red interior curves are obtained using the morphological pyramid (III) and the yellow portions represents the parts where these two results are identical.

We tested the proposed segmentation techniques on different datasets of images from tissues labelled with DAPI. The results confirmed that the proposed methods could efficiently segment the crypts with a high degree of accuracy.

Even that the computation time and the used memory have been considerably reduced in the proposed approach based on pyramids, the quality of the results has been preserved, i.e. in Fig. 4 they differ only by a few non-significant pixels. More than that, in many cases the pyramid approach provides even better results, for instance in Fig. 3 b the red curves delimit with higher accuracy the lumens (regions A and B). The stability of the segmentation technique has been also improved by eliminating the parameters from II.B. In III.D the lumens are reconstructed by considering the descendent partitions from the third level without any addition analysis and computation time.

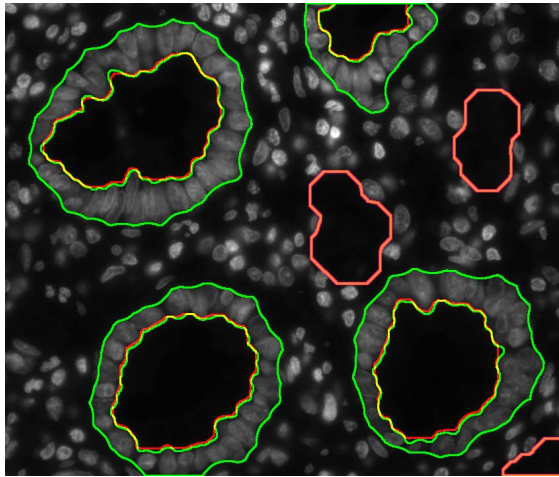


Fig. 4. The interior boundaries obtained using the morphological hierarchy (green curves) and the pyramid (red curves); the results are identical on the yellow portions. The outer borders are depicted with green. The FP boundaries are depicted with orange

A more rigorous evaluation must be done in [17] by comparing the results against the ground truth segmentations. The segmentation quality is analysed by visual inspecting the number of the mis-segmented nuclei per crypt and an average accuracy of 94.07% per crypt has been established.

The over-segmented nuclei are the only type of miss-segmented nuclei. They are produced by the approach used to detect the outer-border which wrongly considered the interstitial nuclei as epithelial nuclei. Since the differences between the inner boundaries from the two results do not include miss-segmented nuclei, the outer border detection and the accuracy are not influenced by the new results.

VI. CONCLUSIONS

An enhanced automatic technique for robust crypt segmentation based on pyramids is presented in this paper. A closing morphological pyramid is used to identify the lumen positions and the associated binary pyramid for the lumen reconstruction. The maximal isoline is used to eliminate the false positive lumens and to validate the epithelial nuclei belonging to crypts.

A detailed analysis of the benefits of using the pyramid instead of the hierarchy is presented. The importance of the sampling step in reducing the computation time and the amount of processed data is described by comparing the number of processed pixels in both situations. The transition from the un-sampled to the sampled domain is made using a set of formula derived from the rules used to create the morphological hierarchy. These results can be used in any situations in which the sampling step is needed for the same purposes like here or different others.

This technique uses a coarser-to-fine approach and can be easily extended on any image with human body cell nuclei from different tissues types (e.g. prostate, breast or lung) but also in any other field in which the objects of

interest have the features considered in designing this method.

REFERENCES

- [1] V. T. Ta, O. Lezoray, A. El Moataz, and S. Schupp, "Graph-based tools for microscopic cellular image segmentation," *Pattern Recognition*, vol. 42, no. 6, pp. 1113-1125, 2009.
- [2] T. W. Nattkemper, "Automatic segmentation of digital micrographs: A survey," *Journal Studies in health technology and informatics*, vol. 107, no. 2, pp. 847-852, 2005.
- [3] H. S. Wu, R. Xu, N. Harpaz, D. Burstein, and J. Gil, "Segmentation of intestinal gland images with iterative region growing," *Journal of Microscopy*, vol. 220, no. 3, pp. 190-204, 2005.
- [4] R. Farjam, H. Soltanian-Zadeh, K. Jafari-Khouzani, and R. A. Zoroofi, "An image analysis approach for automatic malignancy determination of prostate pathological images," *Clinical Cytometry*, vol. 72B, no. 4, pp. 227-240, 2007.
- [5] C. Gunduz-Demir, M. Kandemir, A. B. Tosun, and C. Sokmensuer, "Automatic segmentation of colon glands using object-graphs," *Medical Image Analysis*, vol. 14, no. 1, pp. 1-12, 2010.
- [6] S. Naik, S. Doyle, M. Feldman, J. Tomaszewski, and A. Madabhushi, "Gland segmentation and computerized Gleason grading of prostate histology by integrating low-, high-level and domain specific information," In *2nd MICCAI Workshop Microscopic Image Analysis with Appl. in Biology*, Piscataway, NJ, USA, 2007.
- [7] C. H. Chen, and G. G. Lee, "Image Segmentation Using Multiresolution Wavelet Analysis and Expectation-Maximization (EM) Algorithm for Digital Mammography," *International Journal of Imaging Systems and Technology*, vol. 8, no. 5, pp. 491-504, 1997.
- [8] V. S. Roshni, and G. Raju, "Image Segmentation using Multiresolution Texture Gradient and Watershed Algorithm," *International Journal of Computer Applications*, vol. 22, no. 5, pp. 21-28, 2011.
- [9] M. F. Tolba, M. G. Mostafa, T. F. Gharib, and M. A. Salem, "MR-Brain Image Segmentation Using Gaussian Multiresolution Analysis and the EM Algorithm," in *Proc. ICEIS (2)*, pp.165-170, 2003.
- [10] A. Humphries, and N.A. Wright, "Colonic Crypt Organization and Tumorigenesis: Human Colonic Crypts," *Nature Reviews Cancer*, vol. 8, no. 6, pp. 415-424, 2008.
- [11] K. Morikawa, and J. Yanagida, "Visualization of individual DNA molecules in solution by light microscopy: DAPI staining method," *Japanese Biochemical Society*, vol. 89, no. 2, pp. 693-696, Tokyo 1981.
- [12] W. G. Kropatsch, Y. Haxhimusa, and A. Ion, "Multiresolution Image Segmentations in Graph Pyramids," *Applied Graph Theory in Computer Vision and Pattern Recognition*, Springer Wien New York, vol. 52, pp. 3-41, 2007.
- [13] R. C. Gonzalez, and R.E. Woods, *Digital Image Processing* (third edition), Reading, Massachusetts: Addison-Wesley, 1992.
- [14] R. Haralick, J. Lee, C. Lin, and X. Zhuang, "Multi-resolution Morphology," In *First IEEE conference on Computer Vision*, London, 1987.
- [15] R. M. Haralick, X. Zhuang, C. Lin, and J. S. J. Lee, "The digital morphological sampling theorem," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 2067-2090, 1989.
- [16] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," in *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62-66, 1979.
- [17] C. Smochina, V. Manta, and W.G. Kropatsch, "Semantic segmentation of microscopic images using a morphological hierarchy," *14th International Conference on Computer Analysis of Images and Patterns*, Seville, Spain, August 2011, to be published.
- [18] A. Morales, Raj Acharya, and S. J. Ko, "Morphological Pyramids with Alternating Sequential Filters," *IEEE Transactions on Image Processing*, vol. 4, no. 7, pp. 965-977, July 1995.

Sliding-mode Control of Coordinated Nonholonomic Mobile Manipulators

Razvan Solea, *Member, IEEE* and Daniela Cristina Cernega

Abstract— Coordinated systems of multiple mobile manipulators can be much more flexible than single mobile manipulator in order to accomplish complex and changeable tasks. In this paper, sliding-mode control strategies are used to control the coordinated multiple mobile manipulators which carry a common object. The control ensures that the output tracking errors of the system converge to zero. Simulation studies show the effectiveness of the proposed scheme.

I. INTRODUCTION

The main application area of manipulators are factories and industrial environments where they are capable to perform tasks such as welding, assembly, painting, polishing, grinding. The latest researches related to these kinds of systems mostly focus on the improvement of performance, accuracy and speed. These kind of manipulators have limited working areas due to their fixed based structures. It is possible to increase the workspaces of the manipulators robots by combining them with mobile type robots.

Any system combining a wheeled mobile platform and one or several manipulators (classically arms) is named wheeled mobile manipulator (WMM) - like in Fig. 1. From the set of constraints and characteristics, different approaches have been developed to control WMM.

Motion planning approach has the objective to obtain open loop controls that steer the system from an initial state to a final state without violating the nonholonomic constraints.

To keep the end-effector robotic arm in constant contact with the constraint surface is a basic requirement for many tasks. Only a few control methods have been proposed to solving this problem. These methods can be classified into two categories: (i) hybrid position force control, for example, [1] and [2], where the position of the object is controlled in a certain direction of the workspace and the internal force of the object is controlled in a small range of the origin; (ii) leaderfollower method, for example, [3], [4], and [5], where one or a group of mobile manipulators play the role of the leader, with a preplanned trajectory.

As a solution of this problem, in this paper, a coordinated motion control algorithm for multiple mobile manipulators handling a single object without using force/torque sensors is proposed.



Fig. 1. Pioneer 3DX mobile robot with Cyton Veta manipulator

There are three control algorithms and strategies, such as continuous time-variant, discontinuous and hybrid control strategies [6], [7], [8].

Output tracking laws are easier to design and implement, and can be embedded in a sensorbased control architecture when the task is not fully known in advance. For this reason, with the exception of [9] that takes a somehow intermediate approach, most works on WMMs focus on kinematic control, e.g., [10], [11], [12].

The rest of the paper is organized as follows: Section 2 develops the notation and the kinematic model for the WMM under consideration. Section 3 focuses on creation of a kinematic control law based on sliding-mode strategy. Section 4 presents simulation results to prove the effectiveness of the trajectory- tracking control scheme. Section 5 concludes the paper with a brief discussion and summarizes the directions for future work.

II. MODELING OF THE SYSTEM

In the considered case, the mobile manipulator system consists of a differentially driven wheeled mobile robot and an unactuated planar 2 d.o.f arm (for simplicity) mounted on it. This arm is capable of motions in the horizontal plane.

We consider a system consisting of multiple mobile manipulators carrying a common object as shown in Fig. 2. To solve the control problem, it is considered a system with two wheeled mobile manipulators and an object to be rigidly held by the two robots as shown in Fig.2

In order to obtain a suitable model of the considered system, the following assumptions are made.

Assumption 2.1 All the end-effectors of the manipulators are rigidly attached to the common object so that there is no relative motion between the object and the end-effectors.

This work was supported by CNCIS-UEFISCSU, Romania, project PNII-IDEI 506/2008

R. Solea and D.C. Cernega are with Faculty of Automatic Control, Computers, Electrical and Electronics Engineering, "Dunarea de Jos", University of Galati, Romania razvan.solea@ugal.ro daniela.cernega@ugal.ro

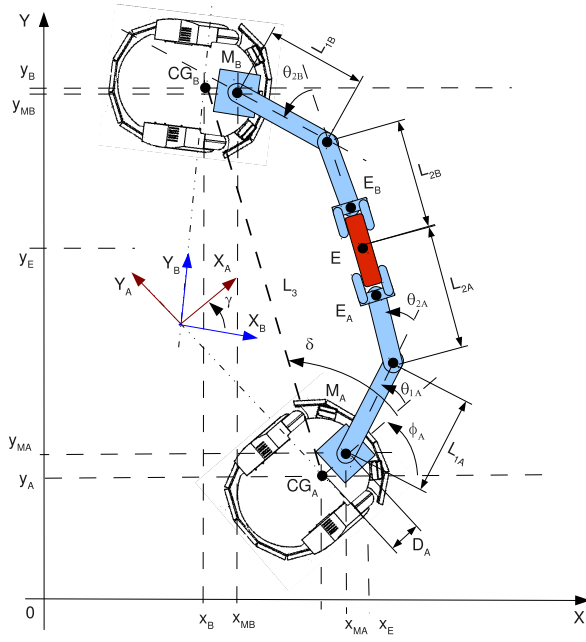


Fig. 2. Two wheeled mobile manipulators carrying a common object

Assumption 2.2 The object is rigid, meaning that, the object does not get deformed as an effect of the application of forces.

Assumption 2.3 Each manipulator is non-redundant and is operating away from any singularity.

The determination of the actuator rates for a given end effector motion of a redundant manipulator is typically an under-constrained problem. A number of schemes have been proposed in the literature to solve the redundancy. The principal underlying theme is to optimize some measurements of performances based on kinematics and/or dynamics of the system [13], [14]. In this paper the attention is focused on kinematic redundancy resolution schemes.

In this section is explained the coordinated control algorithm of multiple mobile manipulators handling a single object, which is extension of the control algorithms proposed in [15]. As shown in Fig. 2, in this control algorithm, the desired trajectory of the object is given to a mobile manipulator, called leader. The other mobile manipulator, called followers, estimate the desired trajectory of the object and manipulate the object in coordination with the leader, based on the estimated trajectory.

A. Kinematic Model for Each Mobile Manipulator

In this section, we present the notation and the kinematic model of one mobile manipulator under consideration. Referring to Figure 3, the WMM_A under consideration consists of a differentially driven WMR_A base with a mounted planar two-link manipulator. The wheels are located at a distance b from the center of the wheel axle. The wheel has a radius r . The base of the manipulator is located at a distance D_A from the center of the wheel axle. The length of the first and second links are L_{1A} and L_{2A} .

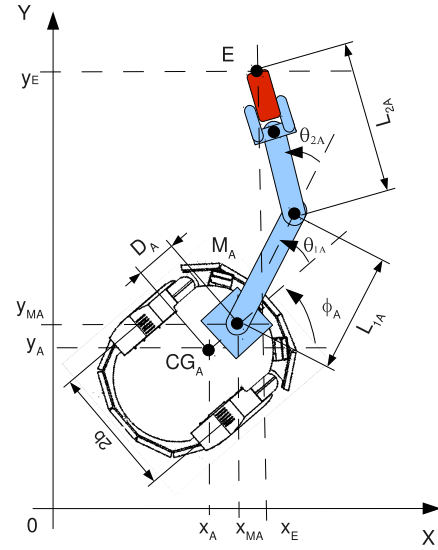


Fig. 3. Schematic of the WMM_A

The midpoint of the commonly object which forms the effective coupler link of the four-bar (link 2) is selected to be the end-effector (point E) of the overall system.

Motion planning has been treated mostly as a kinematic problem where the dynamics of the system have been generally neglected. However, with nonholonomic systems, ignoring the dynamics reduces the significance of the results to low speeds although it is well documented that avoidance of obstacles, parking maneuverability and more motion control is feasible at higher speeds as well.

The configuration of a WMM can be completely described by the following generalized coordinates:

$$q_A^T = [x_A, y_A, \phi_A, \theta_{1A}, \theta_{2A}] \quad (1)$$

where $[x_A, y_A, \phi_A]$ describes the configuration of the WMR_A and $[\theta_{1A}, \theta_{2A}]$ describes the configuration of the planar manipulator. (x_A, y_A) is the Cartesian position of the center of the axle of the WMR_A , ϕ_A is the orientation of the WMR_A , and θ_{1A}, θ_{2A} are the relative angles that parameterize the first and second link of the mounted manipulator. The kinematics of the differentially-driven WMR_A can be represented by its equivalent unicycle model, and described as:

$$\begin{aligned} \dot{x}_A &= v_A \cdot \cos(\phi_A) \\ \dot{y}_A &= v_A \cdot \sin(\phi_A) \\ \dot{\phi}_A &= \omega_A \end{aligned} \quad (2)$$

where v_A and ω_A are the forward and angular velocities inputs.

The position and orientation of the end-effector in the world frame can be derived from homogeneous transform according to the position and orientation of the mobile robot in the world frame. The kinematics of the mobile manipulator

can be described like:

$$\begin{aligned} x_E &= x_{MA} + L_{1A} \cdot \cos(\phi_A + \theta_{1A}) + \\ &+ L_{2A} \cdot \cos(\phi_A + \theta_{1A} + \theta_{2A}) \\ y_E &= y_{MA} + L_{1A} \cdot \sin(\phi_A + \theta_{1A}) + \\ &+ L_{2A} \cdot \sin(\phi_A + \theta_{1A} + \theta_{2A}) \end{aligned} \quad (3)$$

where (x_{MA}, y_{MA}) is the position of mounting point M_A of the mobile platform and ϕ_A is the platform orientation. Eqs.3 show that the position of the end-effector E depends on the position and the orientation of the mobile platform. This illustrates the fact that mobile manipulators, in contrast to fixed ones can have an infinite workspace.

$$\begin{aligned} x_{MA} &= x_A + D_A \cdot \cos(\phi_A) \\ y_{MA} &= y_A + D_A \cdot \sin(\phi_A) \end{aligned} \quad (4)$$

By differentiating eqs. (3) and (4) will get:

$$\begin{aligned} \dot{x}_E &= v_A \cdot \cos(\phi_A) - D_A \cdot \omega_A \cdot \sin(\phi_A) - \\ &- L_{1A} \cdot (\omega_A + \omega_{1A}) \cdot \sin(\phi_A + \theta_{1A}) - \\ &- L_{2A} \cdot (\omega_A + \omega_{1A} + \omega_{2A}) \cdot \sin(\phi_A + \theta_{1A} + \theta_{2A}) \end{aligned} \quad (5)$$

$$\begin{aligned} \dot{y}_E &= v_A \cdot \sin(\phi_A) + D_A \cdot \omega_A \cdot \cos(\phi_A) + \\ &+ L_{1A} \cdot (\omega_A + \omega_{1A}) \cdot \cos(\phi_A + \theta_{1A}) - \\ &- L_{2A} \cdot (\omega_A + \omega_{1A} + \omega_{2A}) \cdot \cos(\phi_A + \theta_{1A} + \theta_{2A}) \end{aligned}$$

If the next inequality is not satisfied, then the target is outside the manipulator, reach and therefore the mobile platform must move in order to bring the target into the manipulator's workspace.

$$\begin{aligned} |\cos(\phi_A)| &\leq 1 \Rightarrow \\ \Rightarrow (x_E - x_{MA})^2 + (y_E - y_{MA})^2 &\leq (L_{1A} + L_{2A})^2 \end{aligned} \quad (6)$$

The first constraint refers to the non-holonomic behavior of the wheels. It constrains the velocity of the WMM to be along the rolling direction of the wheels only. Velocity perpendicular to the rolling direction must be zero as follows:

$$\dot{x}_A \cdot \sin(\phi_A) - \dot{y}_A \cdot \cos(\phi_A) = 0 \quad (7)$$

This constraint, written for the manipulator attachment point M_A , becomes:

$$\dot{x}_A \cdot \sin(\phi_A) - \dot{y}_A \cdot \cos(\phi_A) + \dot{\phi}_A \cdot D_A = 0 \quad (8)$$

B. Leader-Following Formation Model

Figure 2 is a leader-following control model where the formation pattern is specified by the separate distance L_3 (L_3 is the length of virtual baseline between the centers of the two mobile platforms while δ is the angle made by this baseline with respect to the frame of reference robot A) and the relative bearing γ (γ represents the orientation of the frame of robot B with respect to the frame of robot A) for two robots A and B .

The ideal trajectory for mobile robot B is parallel with the trajectory of mobile robot A at desired distance L_3^d and the relative bearing γ^d (see Fig. 4).

The x_e , y_e and ϕ_e , are the trajectory-tracking errors (for mobile robot B) defined in Fig. 4:

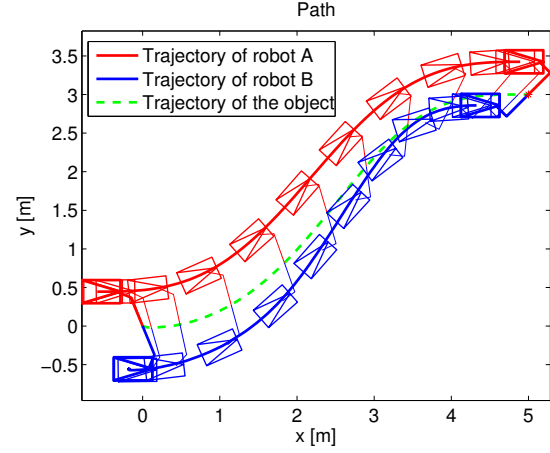


Fig. 4. Maneuver example using WMMs. Initial pose: $x_E(0) = 0$, $y_E(0) = 0$, $\phi_A(0) = 0$, $\theta_{1A}(0) = -\pi/8$, $\theta_{2A}(0) = -\pi/4$, $\theta_{2B}(0) = -\pi/4$. Final pose: $x_E = 5$, $y_E = 3$, $\phi_A = 0$, $\theta_{1A} = -\pi/4$, $\theta_{2A} = -\pi/2$, $\theta_{2B} = -\pi/2$

$$\begin{aligned} x_e &= (x_B - (x_A + L_3^d \cdot \cos(\delta^d))) \cdot \cos(\phi_A) + \\ &+ (y_B - (y_A + L_3^d \cdot \sin(\delta^d))) \cdot \sin(\phi_A) \\ y_e &= -(x_B - (x_A + L_3^d \cdot \cos(\delta^d))) \cdot \sin(\phi_A) + \\ &+ (y_B - (y_A + L_3^d \cdot \sin(\delta^d))) \cdot \cos(\phi_A) \\ \phi_e &= \delta^d - \phi_B - \phi_A \end{aligned} \quad (9)$$

III. SLIDING MODE CONTROLLER DESIGN

A WMM system is especially useful when the manipulator task is outside the manipulator reach. Therefore, in this section we assume that this is always the case, in other words that inequality (6) is not satisfied for a given target.

In this chapter is developed a control routine for the mobile robot that allows for independent control of both the task-space (end-effector) and the configuration-space (mobile base). As mentioned before the primary task is controlling the position of the end-effector and attached payload. The trajectory of the mobile base consists of a time varying function of the position of the end-effector. Once the end-effector final position is known, there exists extra degrees-of-freedom that need to be controlled. These consist of the posture of the mobile robot base and arms. This is depicted in Figure 4, where a mobile robot is shown moving from initial position to the final position.

Assumption: The prescribed final posture (position, $[x_E, y_E]$ and orientation $[\phi_A, \theta_{1A}, \theta_{2A}, \theta_{2B}]$) is required.

A trajectory planner for wheeled mobile manipulators must generate smooth velocity profiles (linear and angular) with low associated accelerations. The trajectory planning process can be divided into two separate parts. First, a continuous collision-free path is generated. In a second step, called trajectory generation, a velocity profile along the path is determined.

In this section, is proposed a sliding-mode trajectory-tracking controller, in Cartesian space, where trajectory-tracking is achieved even in the presence of large initial pose

errors and disturbances. A well-studied example of a non-holonomic system is a WMM which is subject of the rolling without slipping constraint.

In trajectory tracking is an objective to control the non-holonomic WMM to follow a desired trajectory, with a given orientation relatively to the path tangent, even when disturbances exist.

Let us define the sliding surface $S = [s_1, s_2, s_3, s_4, s_5]^T$ as:

$$\begin{aligned} s_1 &= \dot{x}_e + \gamma_x \cdot x_e \\ s_2 &= \dot{y}_e + \gamma_y \cdot y_e + \gamma_0 \cdot \text{sgn}(y_e) \cdot \phi_e \\ s_3 &= \dot{\theta}_{e1} + \gamma_{\theta_{1A}} \cdot \theta_{e1} \\ s_4 &= \dot{\theta}_{e2} + \gamma_{\theta_{2A}} \cdot \theta_{e2} \\ s_5 &= \dot{\theta}_{e3} + \gamma_{\theta_{2B}} \cdot \theta_{e3} \end{aligned} \quad (10)$$

where $\gamma_0, \gamma_x, \gamma_y, \gamma_{\theta_{1A}}, \gamma_{\theta_{2A}}$ and $\gamma_{\theta_{2B}}$ are positive constant parameters, $x_e, y_e, \phi_e, \theta_{e1}, \theta_{e2}$ and θ_{e3} are the trajectory-tracking errors defined in Fig. 5:

$$\begin{aligned} x_e &= (x_B - x_d) \cdot \cos(\phi_d) + (y_B - y_d) \cdot \sin(\phi_d) \\ y_e &= -(x_B - x_d) \cdot \sin(\phi_d) + (y_B - y_d) \cdot \cos(\phi_d) \\ \phi_e &= \phi_B - \phi_d \\ \theta_{e1} &= \theta_{1A} - \theta_{d1} \\ \theta_{e2} &= \theta_{2A} - \theta_{d2} \\ \theta_{e3} &= \theta_{2B} - \theta_{d3} \end{aligned} \quad (11)$$

If s_1 converges to zero, trivially x_e converges to zero. If s_2 converges to zero, in steady-state it becomes $y_e = \gamma_y \cdot y_e \gamma_0 \cdot \text{sgn}(y_e) \cdot \phi_e$. For $y_e < 0 \Rightarrow \dot{y}_e > 0$ if only if $\gamma_0 < \gamma_y \cdot |y_e|/|\phi_e|$. For $y_e > 0 \Rightarrow \dot{y}_e < 0$ if only if $\gamma_0 < \gamma_y \cdot |y_e|/|\phi_e|$. Finally, it can be known from s_2 that convergence of y_e and $\dot{y}_e$ leads to convergence of ϕ_e to zero. If s_3, s_4 and s_5 converges to zero, trivially θ_{e1}, θ_{e2} and θ_{e3} converges to zero.

The reaching law is a differential equation which specifies the dynamics of a switching function S . Gao and Hung [17] proposed a reaching law which directly specifies the dynamics of the switching surface by the differential equation

$$\dot{s}_i = -p_i \cdot s_i - q_i \cdot \text{sgn}(s_i) \quad (12)$$

where $p_i > 0, q_i > 0, i = 1 \text{ to } 5$.

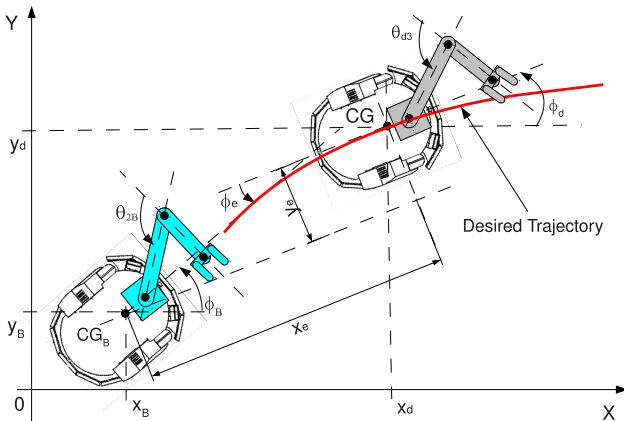


Fig. 5. Lateral, longitudinal and orientation errors for robot B

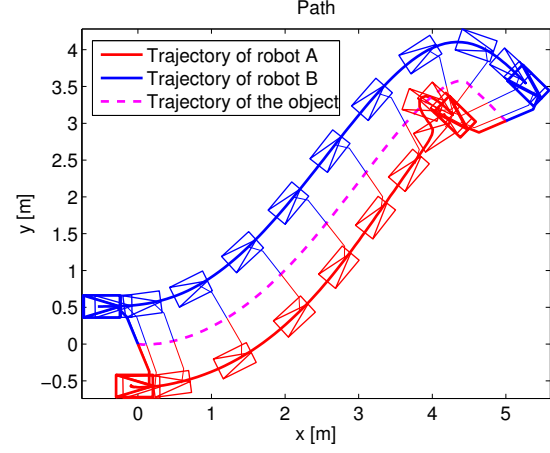


Fig. 6. Scenario 1 - Path of multiple mobile manipulators carrying a common object

By adding the proportional rate term $p_1 \cdot s_i$, the state is forced to approach the switching manifolds faster when s is large. It can be shown that the reaching time is finite and is given by:

$$T_i = \frac{1}{p_i} \cdot \ln \frac{p_i \cdot |s_i| + q_i}{q_i} \quad (13)$$

From the time derivative of (10) and using the reaching laws defined in (12) yields:

$$\begin{aligned} \ddot{x}_e + \gamma_x \cdot \dot{x}_e &= -p_1 \cdot s_1 - q_1 \cdot \text{sgn}(s_1) \\ \ddot{y}_e + \gamma_y \cdot \dot{y}_e + \gamma_0 \cdot \text{sgn}(y_e) \cdot \dot{\phi}_e &= -p_2 \cdot s_2 - q_2 \cdot \text{sgn}(s_2) \\ \ddot{\theta}_{e1} + \gamma_{\theta_{1A}} \cdot \dot{\theta}_{e1} &= -p_3 \cdot s_3 - q_3 \cdot \text{sgn}(s_3) \\ \ddot{\theta}_{e2} + \gamma_{\theta_{2A}} \cdot \dot{\theta}_{e2} &= -p_4 \cdot s_4 - q_4 \cdot \text{sgn}(s_4) \\ \ddot{\theta}_{e3} + \gamma_{\theta_{2B}} \cdot \dot{\theta}_{e3} &= -p_5 \cdot s_5 - q_5 \cdot \text{sgn}(s_5) \end{aligned} \quad (14)$$

From (2), (5) (11) and (14), and after some mathematical manipulation, we get the output commands of the sliding-mode trajectory-tracking controller:

$$\begin{aligned} \dot{v}_{cB} &= \frac{1}{\cos(\phi_e)} \cdot (-p_1 \cdot s_1 - q_1 \cdot \text{sgn}(s_1) - \gamma_x \cdot \dot{x}_e - \\ &\quad - y_e \cdot \dot{\omega}_d - \dot{y}_e \cdot \omega_d + v_B \cdot \dot{\phi}_e \cdot \sin(\phi_e) + \dot{v}_d) \\ \omega_{cB} &= \frac{1}{v_B \cdot \cos(\phi_e) + \gamma_0 \cdot \text{sgn}(y_e)} \cdot (-p_2 \cdot s_2 - q_2 \cdot \text{sgn}(s_2) - \\ &\quad - \gamma_y \cdot \dot{y}_e - \dot{v}_B \cdot \sin(\phi_e) + x_e \cdot \dot{\omega}_d + \dot{x}_e \cdot \omega_d) + \omega_d \\ \dot{\omega}_{c\theta_{1A}} &= -p_3 \cdot s_3 - q_3 \cdot \text{sgn}(s_3) - \gamma_{\theta_{1A}} \cdot \dot{\theta}_{e1} \\ \dot{\omega}_{c\theta_{2A}} &= -p_4 \cdot s_4 - q_4 \cdot \text{sgn}(s_4) - \gamma_{\theta_{2A}} \cdot \dot{\theta}_{e2} \\ \dot{\omega}_{c\theta_{2B}} &= -p_5 \cdot s_5 - q_5 \cdot \text{sgn}(s_5) - \gamma_{\theta_{2B}} \cdot \dot{\theta}_{e3} \end{aligned} \quad (15)$$

The signum functions in the control laws were replaced by saturation functions, to reduce the chattering phenomenon [18].

IV. SIMULATION AND RESULTS

In this section, simulation results for the proposed sliding-mode controllers are presented. The simulations are performed in Matlab/Simulink environment to verify the behavior of the controlled system. The parameters of the WMM model were chosen to correspond as closely as possible to the real experimental robot presented in Fig. 1 in the following manner: $D = 0.25$ [m], $L_{1A} = 0.20$ [m], $L_{2A} = 0.40$ [m],

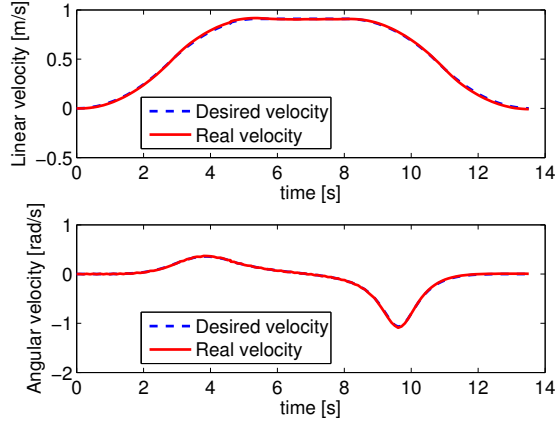


Fig. 7. Scenario 1 - Real and desired linear and angular velocities for second WMM (B)

$L_{1B} = 0.20$ [m], $L_{2B} = 0.40$ [m], $b = 0.04$ [m], $r = 0.04$ [m]. Wheel velocity commands are sent to the power modules of the follower mobile robot and encoder measures N_R and N_L are received in the robots pose estimator for odometric computations.

$$\begin{aligned}\omega_{Aright} &= \frac{v_A + b \cdot \omega_A}{r}; \quad \omega_{Aleft} = \frac{v_A - b \cdot \omega_A}{r} \\ \omega_{Bright} &= \frac{v_B + b \cdot \omega_B}{r}; \quad \omega_{Bleft} = \frac{v_B - b \cdot \omega_B}{r}\end{aligned}\quad (16)$$

The experimental results for first scenario are shown in Fig.6. The solid lines show the trajectory of leader and the follower (red for robot A - leader, and blue for robot B - follower) The dotted lines show the trajectory for common object. As shown in Fig. 7, it can be seen that the real velocities of the robot B (solid line) are approximately with the desired velocities (dotted lines).

TABLE I
INITIAL AND FINAL POSE - SCENARIO 1.

Name	Value
Initial position of end-effector	$x_E = 0, y_E = 0$
Initial angular position	$\phi_A = 0, \theta_{1A} = \pi/2,$ $\theta_{2A} = \pi/8, \theta_{2B} = \pi/8$
Final position of end-effector	$x_E = 5, y_E = 3$
Final angular position	$\phi_A = -\pi/4, \theta_{1A} = \pi/8,$ $\theta_{2A} = \pi/4, \theta_{2B} = \pi/4$

Fig.9 shows another example of the experiments of handling a single object by two mobile manipulators. For this scenario, the states and workspace of the simulation are tabulated below (see Table II). The errors are rapidly compensated and remain very small in the range of zero (see Figs. 8 and 11).

From these results, it can be seen that the handling of a single object by two mobile manipulators in coordination was achieved without using force/torque sensors.

The simulation results show that the trajectory and tracking errors tend to the desired values, which validates the effectiveness of the control law.

TABLE II
INITIAL AND FINAL POSE - SCENARIO 2.

Name	Value
Initial position of end-effector	$x_E = 1, y_E = 1$
Initial angular position	$\phi_A = -\pi/4, \theta_{1A} = \pi/8,$ $\theta_{2A} = \pi/8, \theta_{2B} = \pi/8$
Final position of end-effector	$x_E = 3, y_E = 8$
Final angular position	$\phi_A = \pi/2, \theta_{1A} = \pi/4,$ $\theta_{2A} = \pi/4, \theta_{2B} = \pi/4$

V. CONCLUSIONS AND FUTURE WORKS

In this paper, we examined the development of a framework for selection of the system configuration of two collaborating wheeled mobile manipulators for transporting a common object. The most important result obtained in this paper suggests that the trajectory tracking problem formulated in the workspace for nonholonomic mobile manipulators can be solved using sliding-mode control laws. The main disadvantage is that the number of variables in the formulation rapidly increases for very complex problems. Also, this paper addressed the problem of trajectory control of nonholonomic vehicles, based on the leader-follower strategy to keep the formation. Both the leader and the follower adopt similar trajectories. The only difference is that the leader traveling along the predefined trajectory, the follower is then adapting its own speed according to the information of system (leader + manipulators). Simulation results illustrated the efficacy of the sliding-mode control.

Future research lines include the experimental validation of our control scheme and the extension of our results to wheeled mobile manipulators. Future investigations will cover the more general case of coordinated system, consisting of multiple mobile manipulators carrying a common object.

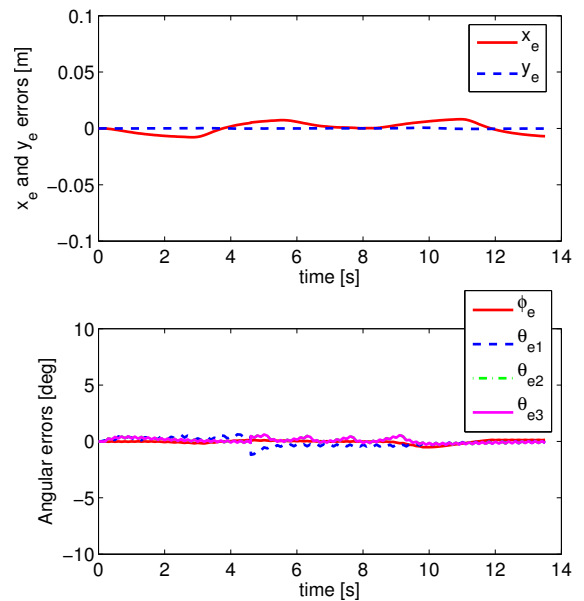


Fig. 8. Scenario 1 - Evolution of the errors

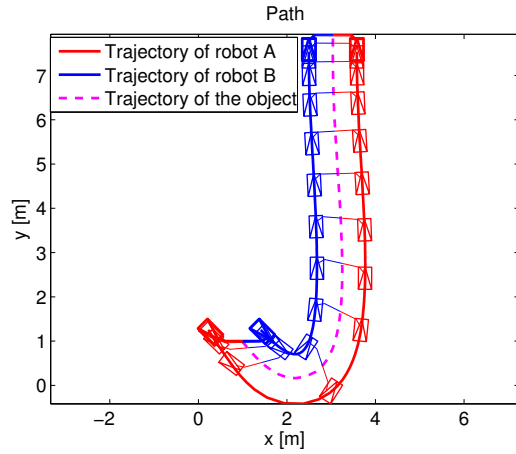


Fig. 9. Scenario 2 - Path of multiple mobile manipulators carrying a common object

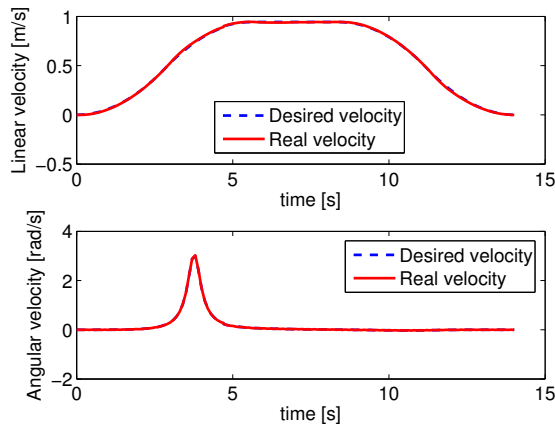


Fig. 10. Scenario 2 - Real and desired linear and angular velocities for second WMM (B)

VI. ACKNOWLEDGMENTS

This work was supported by CNCSIS-UEFISCSU, Romania, project PNII-IDEI 506/2008.

REFERENCES

- [1] M. Fang, W. Chen and Z. Li, "Adaptive Tracking Control of Coordinated Nonholonomic Mobile Manipulators", *Proceedings of the 17th World Congress The International Federation of Automatic Control*, Seoul, Korea, 2008, pp. 4343-4348.
- [2] H. G. Tanner and S. Loizou and K. J. Kyriakopoulos, Nonholonomic navigation and control of cooperating mobile manipulators, *IEEE Transactions on Robotics and Automation*, vol. 19(1), 2003, pp. 53-64.
- [3] A. Trebi-Ollennu, H. Das-Nayar, H. Aghazarian, A. Ganino, P. Piranian, B. Kennedy, T. Huntsberger, P. Schenker, "Mars rover pair cooperatively transporting a long payload", *In Proc. 2002 IEEE International Conf. on Robotics and Automation - ICRA 2002*, 2002, pp. 3136-3141.
- [4] T. G. Sugar and V. Kumar, Control of Cooperating Mobile Manipulators, *IEEE Transactions on Robotics and Automation*, vol. 18(1), 2002, pp. 94-103.
- [5] Y. Kume, Y. Hirata and K. Kosuge, "Coordinated Motion Control of Multiple Mobile Manipulators Handling a Single Object without Using Force/Torque Sensors", *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, USA, 2007, pp. 4077-4082.
- [6] I. Kolmanovsky and N. McClamroch, Developments in nonholonomic control problems, *IEEE Control Systems Magazine*, vol. 15(6), 1995, pp. 20-36.

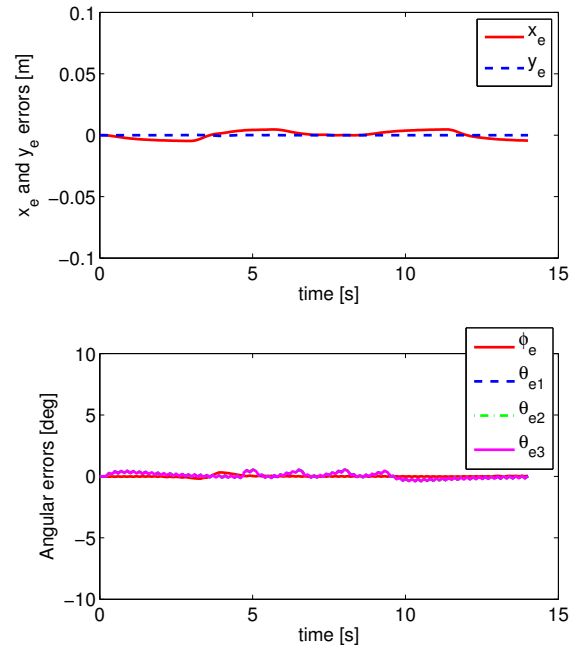


Fig. 11. Scenario 2 - Evolution of the errors

- [7] B. N. Sharma and J. Vanualailai and A. Prasad, Trajectory planning and posture control of multiple mobile manipulators, *International Journal of Applied Mathematics and Computation*, vol. 2(1), 2010, pp. 11-31.
- [8] R. M. Murray, Recent research in cooperative control of multi-vehicle systems, *Journal of Dynamic Systems, Measurement and Control*, vol. 129(5), 2007, pp. 571-583.
- [9] M. Fruchard and P. Morin and C. Samson, "A framework for the control of nonholonomic mobile manipulators", *Rapport De Recherche INRIA*, vol. 5556, 2005, pp. 1-52.
- [10] B. Bayle and J.-Y. Fourquet and F. Lamiriaux and M. Renaud, "Kinematic control of wheeled mobile manipulators", *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, 2002, pp. 1572-1577.
- [11] A. De Luca and G. Oriolo and P. R. Giordano, "Kinematic Control of Nonholonomic Mobile Manipulators in the Presence of Steering Wheels", *IEEE International Conference on Robotics and Automation*, 2010, pp. 1792-1798.
- [12] C. P. Tang and P. T. Miller and V. N. Krov and Ji-C. Ryu and S. K. Agrawal, "Kinematic control of a nonholonomic wheeled mobile manipulator - a differential flatness approach", *Proceedings of ASME Dynamic Systems and Control Conference*, 2008, pp. 1-8.
- [13] Z. Li, P.Y. Tao, S.S. Ge, M. Adams and W.S. Wijesoma, Robust Adaptive Control of Cooperating Mobile Manipulators With Relative Motion, *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 39(1), 2009, pp. 103-116.
- [14] Z. Li, W. Chen, Adaptive neural-fuzzy control of uncertain constrained multiple coordinated nonholonomic mobile manipulators, *Engineering Applications of Artificial Intelligence - ELSEVIER*, vol. 21, 2008, pp. 895-1000.
- [15] R. Solea, D. Cernega, "Trajectory Tracking Control of Mobile Manipulators Based on Kinematics", *Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics - ICINCO 2011*, Noordwijkerhout, Netherlands, 2011.
- [16] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzer, C. Taylor, A vision-based formation control framework, *IEEE Transactions on Robotics and Automation*, vol. 18(5), 2002, pp. 813-825.
- [17] W. Gao and J. Hung, Variable structure control of nonlinear systems: A new approach, *IEEE Transactions on Industrial Electronics*, vol. 40(1), 1993, pp. 45-55.
- [18] J. Slotine and W. Li, Applied Nonlinear Control, *Prentice Hall*, New Jersey, 1991.

The design of a run-time monitoring structure for a MPSoC

A. Stan, *Member, IEEE*, Al. Valachi, Al. Bârleanu

Abstract — MPSoC (MultiProcessor System on Chip) are computing architectures that employ multiple processing cores on the same chip in order to provide the required level of performance and power dissipation for the running applications. MPSoC are attractive because they allow the execution of complex embedded applications that are made of multiple tasks. The tasks are concurrently executed on the available processing cores of the MPSoC and exchange data using the memory resources that may be organized as FIFO buffers. The applications may have real time constraints and in this case any missing deadline may negatively alter the quality of service provided by the system. This paper presents a method that can be used to detect timeliness violations inside a MPSoC. The processing cores of the MPSoC are augmented with a monitoring structure that watches over the communication infrastructure inside the MPSoC. All data transfers are recorded and their timing is compared against a reference expected behavior. If abnormal execution is detected then an exception may be triggered and serviced in a manner suitable for the applications.

I. INTRODUCTION

THE current embedded applications must implement a large variety of services that are required by the end users of embedded devices [1]. A media player, for instance, may play an audio stream and simultaneously display some pictures or render a visualization plug-in. These are complex applications that require suitable computing platforms to execute them.

MPSoC are computing architectures that may be used to execute complex embedded applications. These computing platforms consist of multiple processing cores (CPU or DSP tiles), memory tiles and peripherals. The communication between the elements of a MPSoC is implemented using an interconnection infrastructure.

The applications that are executed on a MPSoC are decomposed in a set of tasks which are executed concurrently by sharing the available resources of the MPSoC. Some of the applications may have real time constraints if the correctness of their operation depends of the timing behavior (e.g. the audio playback). Other applications may not be real time (e.g. file download). In real time applications, missing deadlines has a negative impact on the quality of the service provided by the system (e.g. the

degradation of the quality of the output for audio playback in case the sample rate is altered).

One important attribute of the real time systems is their predictability. A system is predictable if the timing behavior of its parts and interactions between them can be forecast. In order to ensure system predictability careful design methodology must be enforced.

The complexity of a MPSoC and the applications that are running make difficult an accurate analysis of timing behavior of the system and its predictability at design time.

In a real time data-driven system (e.g. streaming applications like audio playback) each application is made of a set of communicating tasks which are executed only when input data is available. The data flow in a data-driven application is described using acyclic data-flow graphs as presented in figure 1.

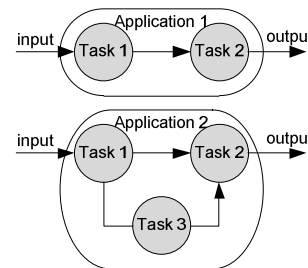


Fig. 1. Data driven application model

The applications might interfere because of the shared resources of a MPSoC and resource conflicts might arise. This interference may cause the timing behavior of the applications to change. More resources may be provided in order to eliminate the resource conflicts. This solution would increase the cost of the hardware part of the system.

The timing behavior of the applications may be altered by malicious attacks (e.g. viruses) and the degradation of the provided service may appear. The malicious attacks are a realistic threat taking into consideration that MPSoC platforms use operating systems (to handle the complexity of application development), application software and communication protocol stacks (to exchange data using various interface to external devices).

The timing behavior of the applications may also be altered by clock frequency variations (e.g. caused by temperature variations in operating environment) or by some hardware failures.

Taking into consideration the above mentioned reasons, a monitoring structure is designed for watching the timing behavior of the running applications on a MPSoC computing platform.

Manuscript received May 18, 2011.

A. Stan is with the Technical University of Iasi, Romania (phone: +40-232-278680 / int. 1319; e-mail: andreis@cs.tuiasi.ro).

Al. Valachi is with the Technical University of Iasi, Romania (phone: +40-232-278680 / int. 1317; e-mail: avalachi@cs.tuiasi.ro).

Al. Bârleanu is with the Technical University of Iasi, Romania (phone: +40-232-278680 / int. 1349; e-mail: alexb@cs.tuiasi.ro).

II. SYSTEM ARCHITECTURE

A. Hardware

In figure 2 is presented the architecture of the processor tile of a MPSoC platform.

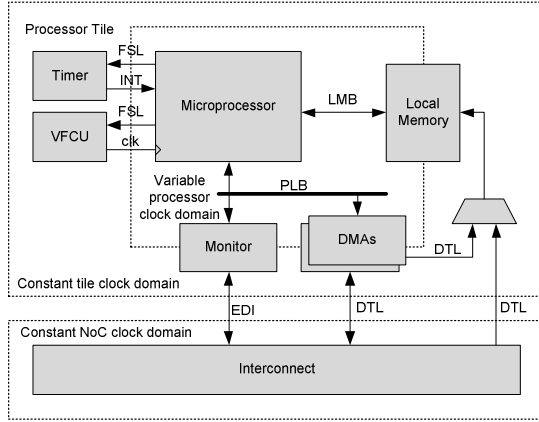


Fig. 2. Processor tile architecture.

A microprocessor tile of a MPSoC has several components. The microprocessor core executes the code of the applications. The Microblaze soft-core processor from Xilinx is used [2], the whole MPSoC platform being implemented in a FPGA circuit.

Local memory is used for local tasks code and data storage. The memory has a LMB (Local Memory Bus) [3] port for transactions with the microprocessor. The memory has two regions: one that is accessible only by the local microprocessor via LMB, and a second region that is used as a communication memory (CMEM). This memory region is further split into outgoing communication memory (CMEM OUT) and incoming communication memory (CMEM IN) as detailed in figure 2. The communication memory is accessible for DMA transactions that move data between tiles. Each application that has tasks mapped to a microprocessor tile has a DMA unit and a communication memory area. The DMA devices are connected to the Peripheral Local Bus (PLB) [4] of the microprocessor. The DMAs are capable of running transfers using the DTL protocol (Device Transaction Level) [5].

A timer module is used by the operating system [6] that runs on the microprocessor core for scheduling algorithms of the tasks and applications.

A voltage frequency control unit (VFCU) is used for power management. It is connected to the processor core using the Fast Simplex Link (FSL) [7] of the Microblaze processor. This unit is responsible for setting the operating frequency of the processor core and also for clock gating for a specified and programmable period of time.

The monitor module is a device that watches transactions on PLB bus and implements the monitoring structure. The details of its implementation and operation are described in section III.

All the tiles of the MPSoC are connected by the Aethereal

NoC (network on chip) [8]. This NoC is designed to provide the applications with a guaranteed level of service and performance. The NoC has DTL ports for data transactions and also features EDI (Event Distribution Interconnect) [9] ports which are used to broadcast events at high speeds between monitoring structures.

The processor tiles and the NoC are implemented into separated clock domains. Each domain has its own clock and therefore clock domain crossing modules [10] are used for the signals that pass from one domain to another.

B. Communication

The communication of data between the tasks mapped on the processing cores of the MPSoC is implemented using the C-HEAP protocol [11] and FIFOs. The details of the communication infrastructure are presented in figure 3.

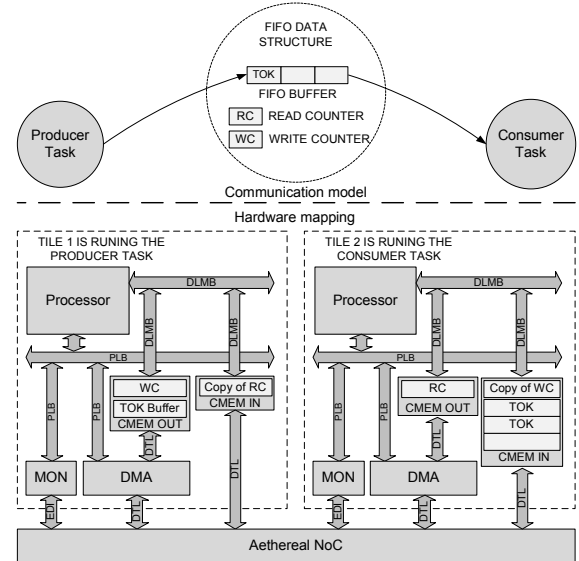


Fig. 3. Inter-task communication infrastructure

The communication model between tasks is a producer-consumer model that uses a FIFO data structure with a circular buffer for data storage and two data pointers that indicate the read (RC) position and write (WC) position into the buffer.

For two tasks that are mapped to two distinct processor tiles, the circular buffer is stored in the local memory of the consumer's task tile. In this way, only write operations move data tokens through the NoC interconnect from the producer's task tile to the consumer's task tile.

The FIFO control information represented by read and write pointers is replicated across the consumer and producer tiles. This is done in order to speed up execution in case the operating system checks the status of a circular buffer in order to test for the availability of input data for tasks. A task is activated by the scheduler if only if there are enough data tokens in the input queue for that task.

The monitoring infrastructure watches the memory accesses that are issued to addresses that correspond to the circular buffer or to the control information of the buffers. In

this way, the monitoring infrastructure non-intrusively hooks up into the monitored system. So, the communication infrastructure provides the basic input data into the monitoring structure.

In a MPSoC platform, the communication infrastructure is good point of observation for the internal working of the device since the operation of the device makes heavily use of the communication infrastructure. The correctness of the operation may be assessed by analyzing the communication timing between tasks inside a MPSoC.

III. MONITORING STRUCTURE

The monitoring structure is implemented as a peripheral device that is mapped into the addressing space of the tile's processor. The high level architecture of the monitoring structure is presented in figure 4.

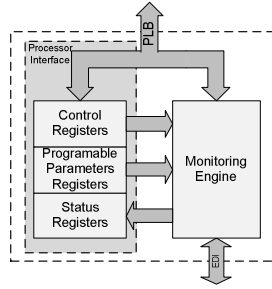


Fig. 4. The architecture of the monitoring structure

The monitoring structure implements a window watchdog mechanism that counts the number of bus events that are triggered before a time limit (TooEarly Counter) or between two time limits (Event Counter) or after a time limit (TooLate Counter). The limits of the time intervals used for watching a certain bus event are stored in programmable parameters registers and are set by applications to appropriate values. These values are computed at design time and verified experimentally [12].

The correct and expected timing behavior of the applications that run on a MPSoC platform may be experimentally determined by running and profiling the application in a safe environment.

The monitoring structure has two main modules. The processor interface module implements an interface for a PLB slave device and provides control, parameters and status registers.

The control registers contain bit fields that are used to enable or disable the monitor engine operation and also to enable or disable the counting of the events that are triggered too early or too late. These settings are made by applications according to their needs and provide a flexible way of watching events.

The status registers of the monitoring structure are loaded with the values of the event counters. These registers may be read by applications or operating system and take appropriate actions when abnormal operating conditions are detected. The values of these registers may trigger

exceptions which are serviced in a manner suitable for the application. The handling mechanism of the abnormal activity is not the scope of this paper.

Figure 5 presents the details of the monitoring engine. The PLB Bus Detector module compares the programmed data from programmable parameters registers with the current values of the PLB bus. If a match is detected then a signal is asserted to the Monitoring Engine FSM (Finite State Machine) module. This module is a window watchdog FSM that verifies if an input signal is asserted between two programmed values of a time counter. The counter inside the FSM uses as a clock source the NoC clock. This clock source is used by all the monitoring structures inside a MPSoC. This choice ensures a consistent time source for all the monitors. In this way, the values of all counters are closely synchronized and their values may be safely used to infer various states for the running tasks (e.g. one task produces a data token without previously consuming one or a task consumes a data token without subsequently producing one in an expected time window).

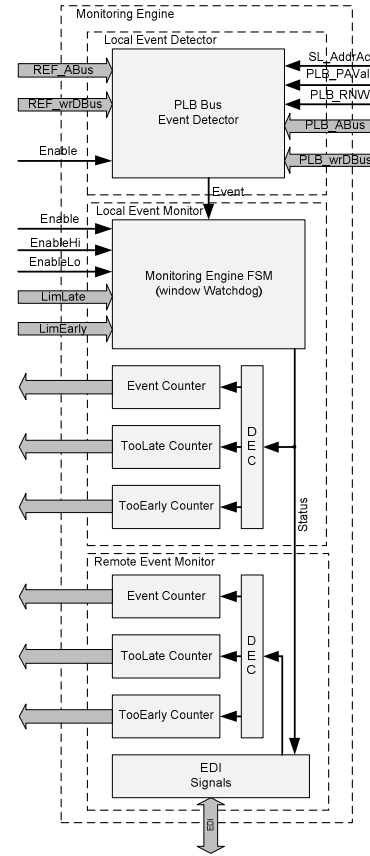


Fig. 5. The architecture of the monitoring engine

The state chart of the window watchdog FSM is presented in figure 6. Two of the inputs (Early and Late) for the FSM are generated by two comparators that compare the current value of the time counter with the programmed limit values from the programmable parameters registers. The third input (Event) comes from the Event Detector module and

represents the monitored event. The FSM has one output (RST) that is used to reset the value of the time counter. The FSM contains a resynchronization state in order to resume monitoring after events that occurred later than normally expected.

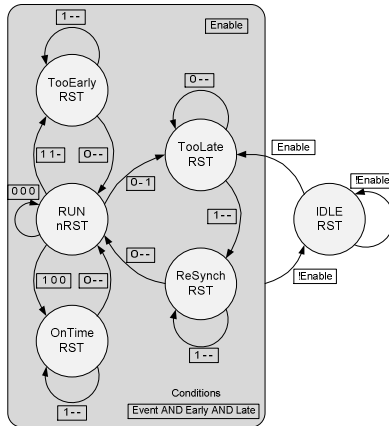


Fig. 6. The state chart of the window watchdog

The bus events that are detected and counted locally by a processor tile are also broadcasted to the other tiles by using the EDI interface. This interface is a low latency interface compared to the DTL interfaces of the NoC. The signals of the EDI interface are not routed according to the routing policy inside the NoC. They are implemented as simple wires that pass through the NoC internal routers.

The intended use of the EDI interface is for manufacturing testing and post-silicon debug [13]. In this paper we propose other usage of this interface: to signal events between tiles at run-time enabling in this way the run-time verification of the applications. By using this simple communication interface at run-time, the utilization of this resource is almost 100% along the life cycle of a product: in design, test and use phases.

As monitoring structure uses counter devices extensively and correctness assessment depends on the counters values there may be used a fault secure implementation of the counters as presented in [14].

IV. CONCLUSION

This paper presented the design of a monitoring structure that captures the communication events between tasks running on a MPSoC device. The events are counted considering the valid time windows for event occurrence. The value of the counters may be used in the assessment of operational correctness.

Also the usage of the EDI interface of the utilized NoC (Aethereal) was extended in order support the distribution of events between monitoring structures in the design. This allows for a distributed and more complex monitoring of the running applications.

The presented mechanism may provide the input information need for a run-time verification environment.

REFERENCES

- [1] S. Gal-On, "Multicore benchmarks help match programming to processor architecture", *MultiCore Expo*, April 2008
- [2] Xilinx, Inc., "Microblaze processor reference guide". Available: www.xilinx.com
- [3] Xilinx, Inc., "Local memory bus product specification". Available: www.xilinx.com
- [4] Xilinx, Inc., "Processor local bus product specification". Available: www.xilinx.com
- [5] Philips Semiconductors, "Device Transaction Level (DTL) Protocol Specification", Version 2.2, July 2002
- [6] A. Hansson, M. Ekerhult, A. Molnos, A. Milutinovic, A. Nelson, J. Ambrose, and K. Goossens, "Design and implementation of an operating system for composable processor sharing", *Microprocessors and Microsystems*, pp. 246-260, Volume 35, Issue 2, March 2011, <http://www.sciencedirect.com/science/article/pii/S0141933110000608>
- [7] Xilinx, Inc., "Fast simplex link product specification". Available: www.xilinx.com
- [8] K. Goossens, J. Dielissen, and A. Rădulescu, "The Aethereal network on chip: Concepts, architectures, and implementations", *IEEE Design and Test of Computers*, pp. 414-421, no. 5, 2005
- [9] B. Vermeulen, K. Goossens, S. Umrani, "Debugging distributed-shared-memory communication at multiple granularities in networks on chip", *International Symposium on Networks-on-Chip*, pp. 3-12, 2008
- [10] C. E. Cummings, "Simulation and synthesis techniques for asynchronous FIFO design", *SNUG*, San Jose, 2002
- [11] A. Nieuwland, J. Kang, O. P. Gangwal, R. Sethuraman, N. Busa, K. Goossens, R. P. Llopis, and P. Lippens, "C-HEAP: A heterogeneous multi-processor architecture template and scalable and flexible protocol for the design of embedded signal processing systems", *Design Automation for Embedded Systems*, pp. 233-270, no. 3, 2002
- [12] R. Wilhelm, "The worst-case execution time problem - overview of methods and survey of tools", *ACM Transactions on Embedded Computing Systems*, vol. 7, no. 3, 2008
- [13] E. Larsson, B. Vermeulen, K. Goossens, "A distributed architecture to check global properties for post-silicon debug", pp. 182-187, 15th IEEE European Test Symposium, 24-28 May, 2010
- [14] K.S. Papadomanolakis, A.P. Kakarountas, N. Sklavos and C.E. Goutis, "A fast Johnson-Mobius encoding scheme for fault secure binary counters", *Design, Automation and Test in Europe*, 2002

Achieve Faster Spanning Tree Convergence

Roxana Stănică and Emil Petre, *Member, IEEE*

Abstract—Spanning Tree Protocol (STP) is a Layer-2 protocol that provides path redundancy and ensures a loop-free topology for bridged Local Area Networks (LANs). Rapid Spanning Tree Protocol (RSTP) is an evolution of STP. RSTP provides faster spanning tree convergence after a topology change. While STP can take 30 to 50 seconds to respond to a topology change, RSTP is able to respond to changes within 3*hello-timer or within a few milliseconds of a physical link failure. This paper contains detailed explanations of STP and RSTP features and new developed commands to ensure a faster convergence. The STP and RSTP structures permit the configuration of different parameters such as hello, forward-delay and MaxAge timers and priority for Bridge Protocol Data Units (BPDUs).

I. INTRODUCTION

SPANNING Tree Protocol (STP), also known as IEEE 802.1d, is a network protocol that ensures a loop-free topology for any bridged Ethernet local area network. Using this protocol, a network can include redundant links that provide automatic backup paths in case of failure of an active link.

STP is based on an algorithm invented by Radia Perlman [3, 4] while working for Digital Equipment Corporation.

STP creates a spanning tree as a mesh network of connected Layer-2 bridges (typically Ethernet switches), and disables the links that are not part of the spanning tree, leaving a single active path between any two network nodes. An Ethernet network functions properly if only one active path exists between any two stations. STP operation is transparent to end stations, which cannot perceive whether they are connected to a single Local Area Network (LAN) segment or to a switched LAN with multiple segments.

To prevent the delay when hosts are connected to a switch and during some topology changes, Rapid STP (RSTP) was developed and standardized by IEEE 802.1w, which allows a switch port to rapidly transition into the Forwarding state (see Table 3).

RSTP shares most of STP basic operation characteristics. However, there are some major differences [9]:

- The detection of Root switch failure is done in 3*hello-timer (see Table 2).
- Ports can be configured as edge ports (see Table 5) if they are attached to a LAN that has no other bridges attached. RSTP can also be configured to automatically detect edge ports. As soon as the bridge detects a Bridge

Protocol Data Unit (BPDU) coming to an edge port, the port becomes a non-edge port.

- Unlike STP, RSTP will respond to BPDUs sent from the direction of the Root bridge. An RSTP bridge will "propose" its spanning tree information to its Designated ports (see Table 4). If another RSTP bridge receives this information and determines this is the superior Root information, it sets all its other ports to Discarding. The bridge sends an "agreement" to the first bridge confirming its superior spanning tree information. This allows RSTP to achieve faster convergence times than STP [5].

- RSTP maintains backup details regarding the Discarding status of ports. This avoids timeouts if the current Forwarding ports failed or BPDUs were not received on the Root port in a defined interval.

II. SPANNING TREE PROTOCOL

An example of a network is presented in Fig. 1. The boxes are the bridges and the number associated with each box is the bridge identifier (ID). Each bridge has a unique ID and a configurable priority number (the bridge ID contains both numbers). The letteres, in Fig. 1, are the network segments. The bridge with ID 5 has the smallest ID, therefore it is the Root bridge. If the cost of traversing any network segment is 1, the lowest path cost from bridge 7 to the Root bridge (bridge 5) goes through network segment c. Therefore, the Root port for bridge 7 is the one on network segment c. The lowest path cost to the Root from network segment f goes through bridge 15. Therefore, the Designated port for network segment f is the port that connects bridge 15 to network segment f.

To compare two bridge IDs, the priority is compared first. If two bridges have equal priority, then their MAC addresses are compared.

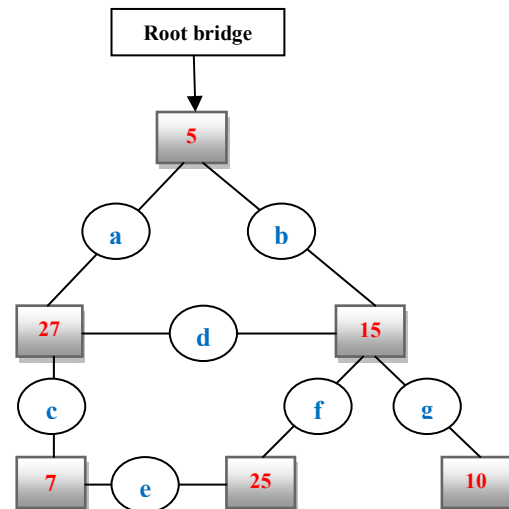


Fig. 1. Network example

R. Stănică is with the S.C. Polystart Security S.R.L., Craiova, Romania (corresponding author; e-mail: roxana_batm@yahoo.com).

E. Petre is with Department of Automatic Control, University of Craiova, Romania (e-mail: epetre@automation.ucv.ro).

A. STP algorithm

The STP algorithm [3] calculates each path cost for all switches in the network's spanning tree. The paths with the lower cost are active paths and the other paths are blocked. STP activates the blocked paths in case the active link fails or if the path cost changes.

- Root Bridge

First step of the algorithm is to choose the active paths in a network, STP first determines a Root bridge. (The Root is the switch towards which all other switches calculate the path cost.) STP then choose the path with the lowest cost from each switch to the Root as the active path, while blocking all other redundant paths.

- Designated bridge

The second step is to choose the Designated bridge, per network segment, after choosing the Root bridge. This is the closest bridge to the Root, and forwards packets from that segment towards the Root bridge.

Each segment has only one Designated bridge. The Designated bridge has one Designated port that forwards packets from the Root bridge to this segment.

- Root and Alternate ports

The last step is to select a Root port (per bridge) that sends data towards the Root bridge.

To avoid loops, all other ports that provide redundant paths to the Root bridge are set as Alternate ports. These ports do not forward traffic unless the Root port goes down.

Each bridge has only one Root port, as a single path toward the Root bridge.

B. Line Error Detection

STP allows interchanging the roles of the Root and Alternate ports when the Cyclic Redundancy Check (CRC) errors reach a critical level [1]. In this case, the Root port's path cost automatically changes into a higher value, triggering the interchange of the Root and Alternate port statuses. CRC is an error-detecting code designed to detect accidental data changes. This *check* (data verification) code is redundant (it adds zero information to the message) and its algorithm is based on *cyclic codes*.

C. STP Bridge Protocol Data Units (BPDUs)

BPDUs are data messages sent to detect loops in a network topology [2]. BPDUs include:

- the Root bridge ID;
- the Designated bridge ID;
- the path cost - the distance from the Root to the switch;
- the Designated port ID.

There are two BPDU types:

- Configuration BPDU - used to define loop-free technology;
- Topology Change Notification (TCN) BPDU - used to announce network topology change. The Designated bridge acknowledges the TCN by immediately sending back a normal configuration BPDU with the topology change acknowledgement (TCA) bit set [9].

D. STP Path Cost

Each bridge port has an assigned path cost, a parameter that determines the port's preference to be included in the

active spanning tree topology. During BPDU exchange, STP sums the path costs along all Designated ports (Designated path cost). This value is the bridge's distance from the Root.

The lower the cost is, the closer the switch is to the Root. If two switches have identical path costs, STP selects the path based on the priorities and bridge IDs.

III. STP PORT STATES AND TIMERS

STP has five port states [7, 8] (see Fig. 2 and Table 1).

To ensure a loop-free network during topology changes inactive ports:

- cannot start forwarding prior to the new topology-information propagating through the switched LAN;
- have to allow frames (that were forwarded using the old topology) to expire.

A. Topology Changes Detection

When a bridge detects a topology change in the network (such as a link failure or the link changing to Forwarding state), it sends this event to the entire bridged network.

The process is done in two stages [6]:

- The bridge notifies the Root.
- The Root broadcasts the information to the entire network.

When a topology change occurs, the address tables of all switches are flushed and new paths are learned.

When the Root is aware of a topology change, it sends the configuration BPDUs with the Topology Change (TC) flag set. As a result, all bridges become aware of the topology change and reduce the MaxAge timer to the forward-delay timer (see Table 2).

B. Calculating the STP Timers

To calculate the STP timers, the following formulas can be used:

$$MaxAge = 4 \times hello + 2 \times dia - 2 \quad (1)$$

$$Forward-delay = (4 \times hello + 3 \times dia) / 2 \quad (2)$$

- Where *hello* is the hello-timer and *dia* is the STP diameter.

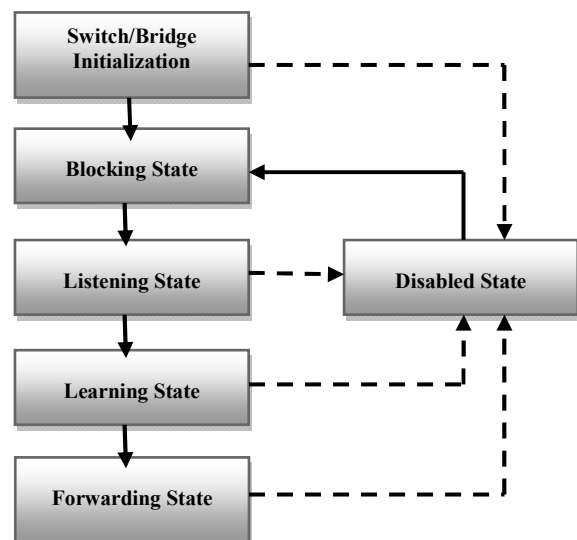


Fig. 2. STP port states

TABLE I
STP STATES

State	Description
Blocking	The port does not forward frames. It moves to this state after the initialization phase, when a different switch/port was selected as Root.
Listening	The listening state is the first transitional state a port enters after the Blocking state. STP determines if the port should forward frames.
Learning	The port is preparing to forward frames. This is the second transitional state through which a port moves in anticipation of frame forwarding.
Forwarding	The port forwards frames. The bridge/switch processes BPDUs and waits for possible new information that might cause it to return to Blocking state to prevent a loop.
Disabled	The port does not participate in forwarding frames and spanning tree.

TABLE II
STP TIMERS

Timer	Description
Hello timer	The interval (between two consecutive BPDUs) a switch sends to other switches.
Forward-delay timer	The time a port is in Listening and Learning states before the port begins forwarding frames.
Maximum-age timer (MaxAge)	The time the switch stores protocol information received on a port.
Message Age	Shows how far a switch is from the Root when it receives a BPDU.

Based on the above formulas, a low hello-timer value decreases the other STP parameters. However, it doubles the amount of BPDUs sent/received by each bridge, causing additional load on the Central Processing Unit (CPU).

STP timers' settings are based on the STP *diameter*, the maximum number of bridges between any two end points on the network. IEEE 802.1D recommends a maximum network diameter of 7 hops.

IV. RAPID SPANNING TREE PROTOCOL

Rapid Spanning Tree Protocol (RSTP) [10] is an evolution of STP. The protocol provides faster convergence (less than one second) when a network topology change occurs. This is critical in networks that carry voice, video, and other delay-sensitive traffic. RSTP automatically selects the bridge that has the best bridge Identifier (ID) as the Root bridge. Each bridge has a unique bridge ID that is derived from the bridge address and from a manageable priority.

A. RSTP algorithm

The RSTP algorithm dynamically creates a tree through the network, used to efficiently direct packets to their destinations. It reduces the bridged network to a single spanning tree topology in order to eliminate packet loops (multiple paths linking one switch to another, resulting in an infinite loop situation).

The RSTP algorithm reactivates redundant connections in the event of a link or switch failure. RSTP distinguishes

between the port state and the port role [10]:

- The port state (see Table 3) describes the relationship of that port to the frame processing (filtering and forwarding) and learning functions.
- The port role (see Table 4) describes the role of the port in the spanning tree function.

In order to create a loop-free environment and to provide rapid convergence, RSTP selects the switch with the highest priority as the Root bridge, assigns the port roles, and determines the active topology.

RSTP assigns a role to each bridge port throughout the bridged LAN.

B. RSTP Bridge Protocol Data Units (BPDUs)

The RSTP BPDU [10] has the same format as the STP BPDU except for the protocol version that is set to 2. The switch proposes itself to be the designated device by setting (see Fig. 3):

- the Proposal flag (bit 1)
- the Port Role flag (bits 2-3) to Designated port
- the Agreement flag (bit 6)
- the Port role flag to Root port

RSTP uses the Topology Change (TC) flag [10] to indicate topology changes. RSTP does not have a separate topology change notification (TCN) BPDU as STP. However, for interoperability with STP switches, the RSTP switch processes and generates TCN BPDUs. The Learning and Forwarding (bits 4 and 5) flags are determined according to the sending port state.

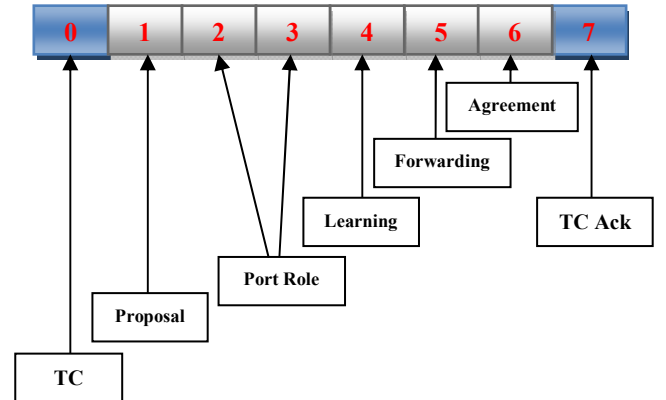


Fig. 3. RSTP BPDU format

TABLE III
RSTP PORT STATES

State	Description
Learning	As in STP, the port prepares to forward frames. It learns source addresses from the frames received and adds them to the filtering database.
Forwarding	As in STP, the port enters this state from the Learning state. The switch processes BPDUs and waits for possible new information that may cause it to move to the Discarding state to prevent a loop.
Discarding	This state contains STP Disabled, Blocking, and Listening states. In this state, a port that does not forward traffic in either direction. The port discards received frames and no learning occurs.

TABLE IV
RSTP PORT ROLES

State	Description
Root port	Port connected to the Root bridge/switch. A Root port is in Forwarding state.
Designated port	Port connected to the Designated bridge/switch - the bridge/switch closest to the Root bridge/switch. Frames are forwarded to the Root through the Designated bridge/switch. A Designated port is in Forwarding state.
Alternate port	Port that offers a path to the Root bridge/switch as an alternative to the path provided by the Root port. The Alternate Port can replace the current Root port if a link failure or a configuration change (such as port priority change) occurs. Alternate port is in Discarding state. This role is equivalent to the STP Blocking state.
Backup port	Backup for the path provided by a Designated port in the direction of the Spanning Tree leaves. Backup port is in Discarding state. This role is equivalent to the STP Blocking state.
Disabled port	Disabled ports do not forward frames and are not operational.

TABLE V
RSTP RAPID CONVERGENCE

State	Description
Edge ports	Edge ports are configured by users. Once they are configured, these ports transit to Forwarding state.
Root ports	When RSTP selects a new Root port, it blocks the old Root port and puts the new Root port in Forwarding state.
Point-to-point links	Point-to-point links are links that directly connect two switches.

C. Line Error Detection

This feature is the same as in STP. For more information, see the *Line Error Detection* chapter.

D. Edge ports

All ports that are directly connected to end stations cannot create loops in the network. Therefore, the edge port directly transitions to the Forwarding state (see **rstp edge-port** command).

V. NEW STP AND RSTP COMMANDS

Fig. 4 details the steps required to configure the STP and RSTP parameters [7, 8]. To apply and configure the new STP and RSTP commands, the steps displayed in Fig.4 must be followed.

A. Enable or disable STP

- stp enable command enables the STP protocol on the switches.
- stp disable command disables the STP protocol.

B. Set STP Bridge Priority

- stp priority *<priority value>* command sets the STP bridge priority. The default bridge priority is 32768.
 - o *<priority value>* is bridge priority in the range of *<0–65535>*. The bridge with the highest bridge priority

(the lowest numerical priority value) is selected as Root switch.

C. Set STP Timers

- stp hello-timer *<hello value>* command sets the time interval between e transmission of BPDUs by a Bridge that is attempting to become the Root or is the Root.
 - o *<hello value>* is the hello-timer interval in the range of *<1–9>* seconds. This value must be less than $MaxAge/2-1$.
- stp maxage *<maxage value>* command sets the interval the switch waits for receiving a BPDU before attempting a reconfiguration.
 - o *<maxage value>* is the maximum aging time in the range of *<6–28>* seconds. The MaxAge value must be *greater than $2*(hello-time+1)$ and less than $2*(forward-delay-1)$* .
- stp forward-delay *<delay value>* command sets the time spent by a port in the Listening state and the Learning state before moving to the Learning state or Forwarding state.
 - o *<delay value>* is the forward-delay interval in the range of *<11–30>* seconds. This value must be greater than $MaxAge/2+1$. When a topology change is underway and is detected, this parameter is used to age all dynamic entries in the Forwarding database.

D. Set STP Path Cost

- stp path-cost *<cost value>* command sets the STP port path cost.
 - o *<cost value>* is the path cost value in the range of *<1–200000000>*. Lower cost values have to be assigned to ports the users want to select first. If all ports have the same cost value, STP puts the port with the lowest number in Forwarding state and blocks the other ports.

E. Set STP BPDU

- stp bpdudiscard | disable | standard command defines the STP reaction when a BPDU is received on the specified port.
 - o discard: the switch drops received BPDUs (ignores the BPDU information)
 - o disable: receiving a BPDU disables the port
 - o standard: BPDUs are processed according to standard STP mechanisms (default)

F. Display STP Configuration

- stp display command shows the current STP configuration.

G. Enable or disable RSTP

- rstp enable command enables the RSTP protocol on the switches.
- rstp disable command disables the RSTP protocol on the switches.

H. Set RSTP Bridge Priority

- rstp priority *<priority value>* command sets the RSTP bridge priority. The default bridge priority is 32768.
 - o *<priority value>* is the RSTP bridge priority in increments of 4096. The valid priority values are: 0, 4096,

8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440.

I. Set RSTP Timers

- `rstp hello-timer <hello value>` command sets the time interval between the transmission of BPDUs by a Bridge that is attempting to become the Root or is the Root.

- o `<hello value>` is the hello-timer interval in the range of `<1–9>` seconds. The default hello-timer value is 2.

- `rstp maxage <maxage value>` command sets the time that learned RSTP information is kept before being discarded.

- o `<maxage value>` is the maximum aging time in the range of `<4–60>` seconds. The default MaxAge value is 20 seconds (see *Calculating the STP timers* chapter).

- `rstp forward-delay <delay value>` command sets the time spent by a port in the Listening state and the Learning state before moving to the Learning or Forwarding state.

- o `<delay value>` is the forward-delay interval in the range of `<4–60>` seconds. The default forward-delay value is 15 seconds (see *Calculating the STP timers* chapter).

J. Set RSTP Edge Port

- `rstp edge-port UU/SS/PP` command set the specified port (`UU/SS/PP`) as an Edge Port.

- o Admin EdgePort: configuring a port as an Edge port is known as Administrative Edge Port; the port has the permission to transition directly to Forwarding state when it becomes Designated.

- o EdgePort: indicates whether the port operates as an Edge Port or not. When a port that was configured as Administrative Edge Port receives a BPDU, it automatically changes its operational state to operate as a non-Edge Port, in order to prevent loops in the network.

K. Set RSTP Path Cost

- `rstp path-cost <cost value>` command defines the RSTP port path cost.

- o `<cost value>` is the RSTP path cost value, in the range of `<1–200000000>`. The path cost value can be used to give priority to preferred links (for example physical speed and bandwidth). When building the active spanning tree, the port path-cost determines which port is included in the active topology. Ports with lower cost values are preferred to ports with higher cost values. If all ports that provide redundant paths to the root bridge have the same path-cost value, RSTP puts the port with the lowest number in Forwarding state and blocks the other ports.

L. Display RSTP Configuration

- `rstp display` command shows the current RSTP configuration.

VI. TESTING RESULTS

The following is an STP/RSTP configuration in a network:

```
1) Enable STP/RSTP:
# stp enable
# rstp enable
```

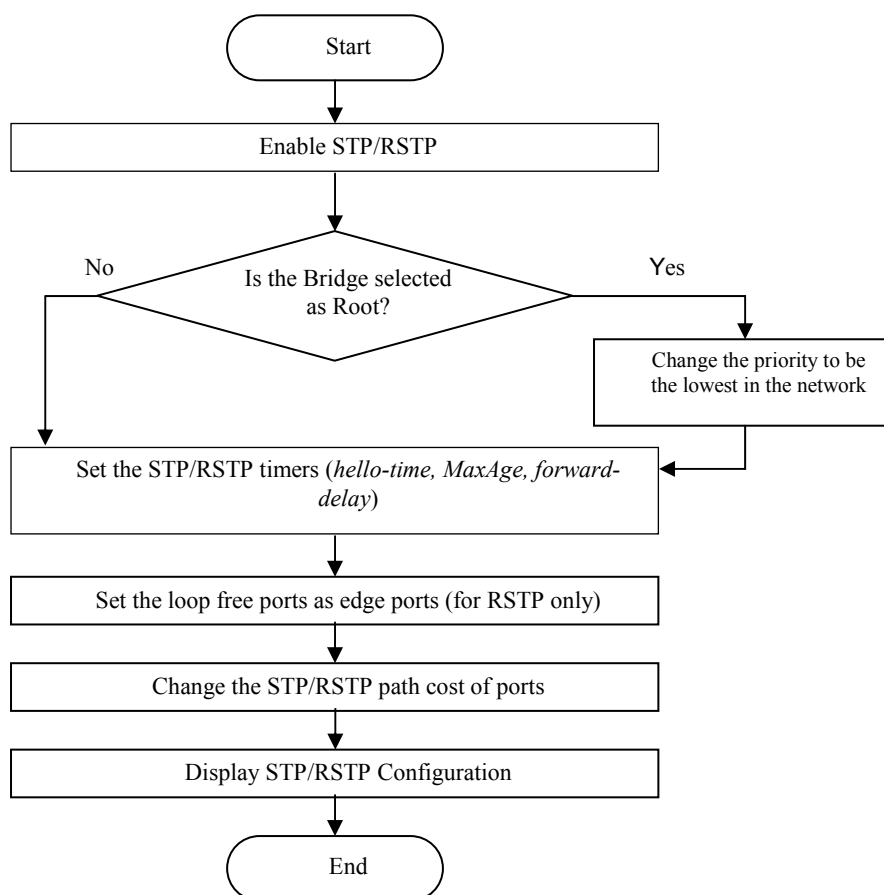


Fig. 4. STP/RSTP configuration

2) Set STP/RSTP priority to 10:

```
# stp priority 10
# rstp priority 10
```

3) Set STP/RSTP MaxAge timer. $Max_age = (4 \times hello) + (2 \times dia) - 2$; hello-timer is 2 (default value) and the diameter is 2:

```
# stp maxage 10
# rstp maxage 10
```

4) Set STP/RSTP forward-delay. $Forward_delay = ((4 \times hello) + (3 \times dia)) / 2$; hello-timer is 2 (default value) and the diameter is 2:

```
# stp forward-delay 7
# rstp forward-delay 7
```

5) Set path cost:

```
# stp path-cost 500
# rstp path-cost 100
```

6) Set port 1/1/1 as edge port:

```
# rstp edge-port 1/1/1
```

7) Display STP/RSTP configuration

stp display

```
Spanning tree      enabled
ProtocolSpecification = ieee8021d
Priority            = 10
TimeSinceTopologyChange = 0 (Sec)
TopChanges         = 4
DesignatedRoot     = 04096.00:12:13:14:00:C0
RootPort           = 1/1/1
RootCost            = 500
MaxAge              = 10 (Sec)
HelloTime           = 2 (Sec)
ForwardDelay        = 7 (Sec)
HoldTime            = 1 (Sec)
BridgeMaxAge        = 20 (Sec)
BridgeHelloTime     = 2 (Sec)
BridgeForwardDelay  = 15 (Sec)
DetectLineCRCReconf = disabled
```

rstp display

```
Rapid spanning tree = enabled
ProtocolSpecification = ieee8021w
Priority             = 10
TimeSinceTopologyChange = 4 (Sec)
TopChanges          = 5
DesignatedRoot      = 04096.00:12:13:14:00:C0
RootPort            = 1/1/1
RootCost             = 100
MaxAge               = 10 (Sec)
HelloTime            = 2 (Sec)
ForwardDelay         = 7 (Sec)
BridgeMaxAge         = 20 (Sec)
BridgeHelloTime      = 2 (Sec)
BridgeForwardDelay   = 15 (Sec)
DetectLineCRCReconf = disabled
EdgePort             = 1/1/1
```

VII. CONCLUDING REMARKS

The Spanning Tree Algorithm and Protocol first standardized in ANSI/IEEE Std. 802.1D-1990 and unchanged in 802.1D-1998, was based on the work of Radia Perlman [3, 4].

Spanning Tree Protocol (STP) is a Layer-2 protocol that runs on bridges and switches that are 802.1D-compliant. STP main purpose is to make sure that users do not create loops when redundant paths appear in a network. Redundant links are as important as backups if a failure occurs in a network. A failure of primary link activates the backup links so that users can continue to use the network. Without STP enabled on bridges and switches, such a failure can result in a loop [8].

STP provides fault tolerance on networks by allowing the implementation of parallel paths for network traffic. The STP algorithm creates a spanning tree (a loop-free subset of the network topology) that enables a Learning bridge to dynamically work around loops in a network topology. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. When two ports on a switch are part of a loop, the STP priority and path cost settings determine which port is put in the Forwarding state and which is put in the Blocking state.

RSTP improves the convergence time. RSTP overcomes slow convergence times by introducing new port roles (edge port, Alternate port and Backup port) and new convergence behavior under specific connectivity conditions (for example, point-to-point or shared medium). These enhancements enable RSTP to achieve sub-second forwarding and network convergence [11].

In this paper, new commands were added for configuring the STP and RSTP features, as well as new commands for displaying the changes performed.

REFERENCES

- [1] S. Angelescu, "CCNA® Certification All-In-One for Dummies®," Wiley Publishing Inc., pp. 385-415, 2010.
- [2] T. Lammle, "CCNA®: Cisco® Certified Network Associate: Study Guide, Sixth Edition," Wiley Publishing Inc., pp. 493-550, 2007.
- [3] R. Perlman, "An algorithm for distributed computation of a Spanning Tree in an Extended LAN," ACM SIGCOMM Computer Communication Review, Proceedings of the ninth symposium on Data communications, vol. 15 (4), pp. 44-53, 1985.
- [4] R. Perlman, "Interconnections, Second Edition: Bridges, Routers, Switches, and Internetworking Protocols" Addison-Wesley Professional Computing Series, pp. 58-94, 2000.
- [5] W. Wojdak, "Rapid Spanning Tree Protocol: A new solution from an old technology," Reprinted from CompactPCI Systems, 2003.
- [6] Z. Xu, "Designing and Implementing IP/MPLS-Based Ethernet Layer 2 VPN Services: An Advanced Guide for VPLS and VLL", Wiley Publishing Inc., pp. 840-895, 2010.
- [7] <http://www.batm.com/int/index/en/sdoc/type/111>
- [8] http://www.cisco.com/en/US/tech/tk389/tk621/technologies_config_uration_example09186a008009467c.shtml
- [9] http://www.cisco.com/en/US/tech/tk389/tk621/technologies_tech_note09186a0080094797.shtml
- [10] http://www.cisco.com/en/US/tech/tk389/tk621/technologies_white_paper09186a0080094cfa.shtml
- [11] <http://www.juniper.net/us/en/local/pdf/implementation-guides/8010002-en.pdf>

Predictive Control of a bioprocess – a hybrid approach

Florin Stinga and Dan Popescu, *Member, IEEE*

Abstract—In this paper the problem of modelling and control of a bioprocess is approached. Those tasks were performed using the multi-model and predictive control approaches. Modelling formalism like MLD (mixed logical dynamical systems) was used to describe the hybrid feature of multiple linear models. Model predictive control, which is a technique used for automatic control of the processes under operating constraints was used in this paper. Finally, some simulation results were presented, in order to demonstrate the effectiveness of the proposed modelling and control algorithms.

I. INTRODUCTION

BIOPROCESSES are characterized by a complex behaviour which results from their complicated structure and characteristics. Bioprocesses are involved in producing some chemical compound synthesized by microorganisms, cultivating a biomass for its utilization, extraction of its metabolites, and to degrade a pollutant. The control of biotechnological processes is considered an important and challenging problem. In the past decade, different control strategies were proposed: neural network [1], model predictive control [2], adaptive control and fuzzy control.

The multiple model approaches allow the modelling of a complex system; such a bioprocess, if partitioned into a number of different local models, which have a limited range of operating conditions where these are sufficiently accurate or perform sufficiently well in order to serve their purpose [3]. The control technique developed on local linear models, known in literature as multiple model control, became an attractive research field.

In this paper we studied the modelling and control of a bioprocess using hybrid and predictive control theory. MLD formalism describes the hybrid feature of multiple linear models, laying the foundation for the predictive control strategies which were considered. Model predictive controls require an optimization problem to be solved on-line in order to compute the next command action [4]. The computation effort can be moved off-line by solving multi-parametric programs [5].

The paper is organized as follows: in Section II the nonlinear mathematical model of the bioprocess is presented and the multiple linear models were obtained, in Section III the MLD form of the studied system is formulated and in

Section IV we proposed a predictive control algorithm. Finally, some simulation results are presented in order to illustrate the performance of the proposed modelling and control algorithms.

II. BIOPROCESS MODEL

A. Nonlinear mathematical model

Let us consider a bioprocess described by the following differential equations [6]:

$$\begin{aligned} \frac{dS_1(t)}{dt} &= -\frac{\mu_{s_1}(t)}{r_{s_1}} X(t) + F(t)(S_{1,in}(t) - S_1(t)) \\ \frac{dS_2(t)}{dt} &= -\frac{\mu_{s_2}(t)}{r_{s_2}} X(t) + F(t)(S_{2,in}(t) - S_2(t)) \\ \frac{dX(t)}{dt} &= (\mu_{s_1}(t) + \mu_{s_2}(t) - F(t))X(t) \end{aligned} \quad (1)$$

where: S_i ($i=1, 2$) are the substrates concentrations (g/l), X is the biomass concentration (g/l), $S_{i,in}$ ($i=1, 2$) are the inlet substrates concentrations (g/l), r_{s_i} ($i=1, 2$) are the yield coefficients, μ_{s_i} ($i=1, 2$) are the specific growth rates (h^{-1}), k_{s_i} ($i=1, 2$) are Michaelis-Menten constants (g/l), F is the dilution rate (h^{-1}).

The model (1) is a nonlinear model; the nonlinearities are present in the reaction rates that depend on state variables of the system:

$$\mu_{s_1}(t) = \mu_{s_1 \max} \frac{S_1(t)}{S_1(t) + k_{s_1}}, \quad \mu_{s_2}(t) = \mu_{s_2 \max} \frac{S_2(t)}{S_2(t) + k_{s_2}}$$

where $\mu_{s_i \max}$ ($i=1, 2$) are the maximum specific growth rates (h^{-1}).

The ranges of the variables are as follows:

$$\begin{aligned} [X, S_1, S_2] &\in [0, 20] \times [0, 6] \times [0, 6], \quad F \in [0.0002, 0.3], \\ [S_{1,in}, S_{2,in}] &\in [0, 6] \times [0, 6]. \end{aligned}$$

The equilibrium states were obtained by solving the following system:

$$\begin{aligned} -\frac{\mu_{s_1 \max}}{r_{s_1}} \frac{\bar{S}_1}{k_{s_1} + \bar{S}_1} \bar{X} + \bar{F}(\bar{S}_{1,in} - \bar{S}_1) &= 0 \\ -\frac{\mu_{s_2 \max}}{r_{s_2}} \frac{\bar{S}_2}{k_{s_2} + \bar{S}_2} \bar{X} + \bar{F}(\bar{S}_{2,in} - \bar{S}_2) &= 0 \\ \left(\mu_{s_1 \max} \frac{\bar{S}_1}{k_{s_1} + \bar{S}_1} + \mu_{s_2 \max} \frac{\bar{S}_2}{k_{s_2} + \bar{S}_2} - \bar{F} \right) \bar{X} &= 0 \end{aligned} \quad (2)$$

This work was supported in part by the grant no. POSDRU/6/1.5/S/14.

Florin Stinga is with the Department of Automatic Control, University of Craiova, 13 A.I. Cuza str., 200585 Craiova, Romania. (e-mail: florin@automation.ucv.ro).

Dan Popescu is with the Department of Automatic Control, University of Craiova, 13 A.I. Cuza str., 200585 Craiova, Romania. (e-mail: dpopescu@automation.ucv.ro).

where: $\bar{S}_1, \bar{S}_2$ and $\bar{X}$ are the equilibrium states defined for the equilibrium constants $\bar{F}, \bar{S}_{1,in}$ and $\bar{S}_{2,in}$.

Two types of equilibrium states are relieved:

- Wash-out states:

$$\bar{X} = 0 \Rightarrow \begin{cases} \bar{S}_1 = \bar{S}_{1,in} \\ \bar{S}_2 = \bar{S}_{2,in} \end{cases}$$

- Operational states determined as solutions of the following system :

$$\begin{aligned} -\frac{\mu_{s1\max}}{r_{s1}} \frac{\bar{S}_1}{k_{s1} + \bar{S}_1} \bar{X} + \bar{F}(\bar{S}_{1,in} - \bar{S}_1) &= 0 \\ -\frac{\mu_{s2\max}}{r_{s2}} \frac{\bar{S}_2}{k_{s2} + \bar{S}_2} \bar{X} + \bar{F}(\bar{S}_{2,in} - \bar{S}_2) &= 0 \\ \mu_{s1\max} \frac{\bar{S}_1}{k_{s1} + \bar{S}_1} + \mu_{s2\max} \frac{\bar{S}_2}{k_{s2} + \bar{S}_2} - \bar{F} &= 0 \end{aligned} \quad (3)$$

In fact, $\bar{S}_1$ and $\bar{S}_2$ are the positive and real solutions of the second order system, described by:

$$\begin{aligned} \bar{S}_1^2(\mu_{s1\max} r_{s1} - \bar{F}) + \bar{S}_1 \left(-\mu_{s1\max} r_{s1} \bar{S}_{1,in} - \right. \\ \left. -\mu_{s1\max} r_{s2} \bar{S}_{2,in} - \bar{F} k_{s1} + \bar{F} \bar{S}_{1,in} \right) + \\ + \mu_{s1\max} r_{s2} \bar{S}_1 \bar{S}_2 + \bar{F} \bar{S}_{1,in} k_{s1} = 0 \\ \bar{S}_2^2(\mu_{s2\max} r_{s2} - \bar{F}) + \bar{S}_2 \left(-\mu_{s2\max} r_{s1} \bar{S}_{1,in} - \right. \\ \left. -\mu_{s2\max} r_{s2} \bar{S}_{2,in} - \bar{F} k_{s2} + \bar{F} \bar{S}_{2,in} \right) + \\ + \mu_{s2\max} r_{s1} \bar{S}_1 \bar{S}_2 + \bar{F} \bar{S}_{2,in} k_{s2} = 0 \end{aligned} \quad (4)$$

where: $\bar{X} = r_{s1} \bar{S}_{1,in} + r_{s2} \bar{S}_{2,in} - r_{s1} \bar{S}_1 - r_{s2} \bar{S}_2$.

B. Multiple linear models

Using Taylor series expansion of nonlinear system equation (1) and retaining only linear terms, we obtain the linear model, valid around each equilibrium point:

$$\dot{x}(t) = Ax(t) + Bu(t) + Dw(t) \quad (5)$$

where:

$$\begin{aligned} x &= \begin{bmatrix} S_1(t) - \bar{S}_1 \\ S_2(t) - \bar{S}_2 \\ X(t) - \bar{X} \end{bmatrix}, \quad u = [F(t) - \bar{F}], \quad w = \begin{bmatrix} S_{1,in}(t) - \bar{S}_{1,in} \\ S_{2,in}(t) - \bar{S}_{2,in} \end{bmatrix} \\ A &= \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, \quad D = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \\ d_{31} & d_{32} \end{bmatrix} \end{aligned}$$

$$\begin{aligned} a_{11} &= \mu_{s1\max} \frac{\bar{S}_1 \bar{X}}{r_{s1} (\bar{S}_1 + k_{s1})^2} - \mu_{s1\max} \frac{\bar{X}}{r_{s1} (\bar{S}_1 + k_{s1})} - \bar{F}, \\ a_{12} &= 0, \quad a_{13} = -\mu_{s1\max} \frac{\bar{S}_1}{r_{s1} (\bar{S}_1 + k_{s1})}, \quad a_{21} = 0 \\ a_{22} &= \mu_{s2\max} \frac{\bar{S}_2 \bar{X}}{r_{s2} (\bar{S}_2 + k_{s2})^2} - \mu_{s2\max} \frac{\bar{X}}{r_{s2} (\bar{S}_2 + k_{s2})} - \bar{F}, \\ a_{23} &= -\mu_{s2\max} \frac{\bar{S}_2}{r_{s2} (\bar{S}_2 + k_{s2})}, \quad a_{31} = \frac{\mu_{s1\max} \bar{X}}{(\bar{S}_1 + k_{s1})} - \frac{\mu_{s1\max} \bar{S}_1}{(\bar{S}_1 + k_{s1})^2}, \\ a_{32} &= \frac{\mu_{s2\max} \bar{X}}{(\bar{S}_2 + k_{s2})} - \frac{\mu_{s2\max} \bar{S}_2}{(\bar{S}_2 + k_{s2})^2}, \\ a_{33} &= \frac{\mu_{s1\max} \bar{S}_1}{(\bar{S}_1 + k_{s1})} + \frac{\mu_{s2\max} \bar{S}_2}{(\bar{S}_2 + k_{s2})} - \bar{F}, \quad b_1 = \bar{S}_{1,in} - \bar{S}_1, \\ b_2 &= \bar{S}_{2,in} - \bar{S}_2, \quad b_3 = -\bar{X}, \quad d_{11} = \bar{F}, \quad d_{12} = 0, \quad d_{21} = 0, \\ d_{22} &= \bar{F}, \quad d_{31} = 0, \quad d_{32} = 0 \end{aligned}$$

The multiple models are obtained by subdividing the operating domain in multiple small operating regimes which characterize microorganism dynamics [6]. Based on biomass concentration versus dilution rate stationary characteristic (figure 1), determined by solving system (4), with the following input parameters:

$$\mu_{s1\max} = 0.1, \quad \mu_{s2\max} = 0.2, \quad r_{s1} = 1.2, \quad r_{s2} = 2.5, \quad k_{s1} = 1, \\ r_{s1} = 0.5, \quad \bar{F} \in [0.0002, 0.3], \quad \bar{S}_{1,in} = \bar{S}_{2,in} = 4.$$

the nonlinear model (1) can be approximated by 5 linear models.

By solving the system (3) for the following values of $\bar{F}$:

$$\bar{F} = 0.05 \quad (\text{for } i = 1)$$

$$\bar{F} = 0.125 \quad (\text{for } i = 2)$$

$$\bar{F} = 0.175 \quad (\text{for } i = 3)$$

$$\bar{F} = 0.215 \quad (\text{for } i = 4)$$

$$\bar{F} = 0.24 \quad (\text{for } i = 5)$$

we obtain the five operation states $[\bar{X}, \bar{S}_1, \bar{S}_2]$, $i = 1 \dots 5$.

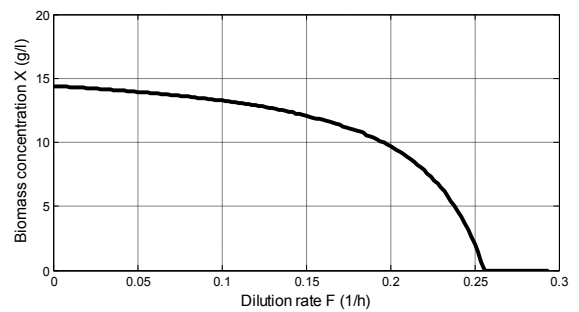


Fig. 1. Biomass concentration versus dilution rate static characteristic

The full state space $R: (0, 20] \times [0, 6] \times [0, 6]$ is divided into five sub-regions:

$$\begin{aligned} R_1 &: (13.26, 20] \times [0, 6] \times [0, 6], \quad R_2: (12.13, 13.26] \times [0, 6] \times [0, 6], \\ R_3 &: (9.82, 12.13] \times [0, 6] \times [0, 6], \quad R_4: (6.76, 9.82] \times [0, 6] \times [0, 6], \\ R_5 &: (0, 6.76] \times [0, 6] \times [0, 6], \text{ so that } R_1 \cup R_2 \cup R_3 \cup R_4 \cup R_5 = R, \\ R_1 \cap R_2 &= \emptyset, \quad R_2 \cap R_3 = \emptyset, \quad R_3 \cap R_4 = \emptyset, \quad R_4 \cap R_5 = \emptyset. \end{aligned}$$

So, the nonlinear system (1) can be approximated by the following linear models:

$$\begin{aligned} \dot{x}(t) &= A_i x(t) + B_i u(t) + D_i w(t) \\ y(t) &= C_i x(t) \end{aligned} \quad (6)$$

for $[X, S_1, S_2] \in R_i, \quad i = 1 \dots 5$

where A_i, B_i, D_i are computed from A, B, D defined in (5) for the 5 operational states, and C_i is chosen as follow:

$$C_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad i = 1 \dots 5.$$

III. MLD REPRESENTATION

MLD is a framework for modelling and controlling models of systems described by interacting physical laws, logic rules and operating constraints. MLD systems are described by dynamic equations subject to linear mixed-integer inequalities. Using this formalism, various classes of systems (linear hybrid systems, finite state machines, some classes of discrete event systems, constrained linear systems and nonlinear systems) can be modelled.

The general form of MLD hybrid systems is as follow [7]:

$$\begin{aligned} x(k+1) &= \bar{A}x(k) + \bar{B}_1 u(k) + \bar{B}_2 \delta(k) + \bar{B}_3 z(k) \\ y(k) &= \bar{C}x(k) + \bar{D}_1 u(k) + \bar{D}_2 \delta(k) + \bar{D}_3 z(k) \\ \bar{E}_2 \delta(k) + \bar{E}_3 z(k) &\leq \bar{E}_1 u(k) + \bar{E}_4 x(k) + \bar{E}_5 \end{aligned} \quad (7)$$

where $k \in \mathbb{Z}$, $x \in \mathbb{R}^{n_c} \times \{0, 1\}^{n_l}$ are the continuous and binary states of the system, $u \in \mathbb{R}^{m_c} \times \{0, 1\}^{m_l}$ are the continuous real and binary inputs $y \in \mathbb{R}^{p_c} \times \{0, 1\}^{p_l}$ are the continuous real and binary outputs, $z \in \mathbb{R}^{r_c}$ and $\delta \in \{0, 1\}^{r_l}$ are the continuous and binary auxiliary states.

MLD representation of the system (6) requires the following steps:

1. The system (6) is discretized using approximation: $\dot{x} \approx \frac{x(k+1) - x(k)}{T}$ with T - sampling time.
2. Define the following continuous and binary variables: $x \in \mathbb{R}^3$ are the states of the system, $u \in \mathbb{R}^3$ are the inputs of the system, $y \in \mathbb{R}^2$ are the outputs of the system, $\delta \in \{0, 1\}^6$ are the binary auxiliary states and $z \in \mathbb{R}^{15}$ are the continuous

auxiliary states.

3. Define the proposition and linear constraints:

$$\begin{aligned} [\delta_1 = 1] &\leftrightarrow [X \leq 20] \\ [\delta_2 = 1] &\leftrightarrow [X \leq 13.26] \\ [\delta_3 = 1] &\leftrightarrow [X \leq 12.13] \\ [\delta_4 = 1] &\leftrightarrow [X \leq 9.82] \\ [\delta_5 = 1] &\leftrightarrow [X \leq 6.76] \\ [\delta_6 = 1] &\leftrightarrow [X \leq 0] \\ [\delta_1 = 1] &\rightarrow [\delta_2 = 1] \wedge [\delta_3 = 1] \wedge [\delta_4 = 1] \wedge [\delta_5 = 1] \wedge [\delta_6 = 1] \\ [\delta_2 = 1] &\rightarrow [\delta_3 = 1] \wedge [\delta_4 = 1] \wedge [\delta_5 = 1] \wedge [\delta_6 = 1] \\ [\delta_3 = 1] &\rightarrow [\delta_4 = 1] \wedge [\delta_5 = 1] \wedge [\delta_6 = 1] \\ [\delta_4 = 1] &\rightarrow [\delta_5 = 1] \wedge [\delta_6 = 1] \\ [\delta_5 = 1] &\rightarrow [\delta_6 = 1] \\ [\delta_1 = 1] \wedge [\delta_2 = 0] &\rightarrow [z_1(k) z_2(k) z_3(k)]^T = [A_1 x + B_1 u] \delta_1 (1 - \delta_2) \\ [\delta_2 = 1] \wedge [\delta_3 = 0] &\rightarrow [z_4(k) z_5(k) z_6(k)]^T = [A_2 x + B_2 u] \delta_2 (1 - \delta_3) \\ [\delta_3 = 1] \wedge [\delta_4 = 0] &\rightarrow [z_7(k) z_8(k) z_9(k)]^T = [A_3 x + B_3 u] \delta_3 (1 - \delta_4) \\ [\delta_4 = 1] \wedge [\delta_5 = 0] &\rightarrow [z_{10}(k) z_{11}(k) z_{12}(k)]^T = [A_4 x + B_4 u] \delta_4 (1 - \delta_5) \\ [\delta_5 = 1] \wedge [\delta_6 = 0] &\rightarrow [z_{13}(k) z_{14}(k) z_{15}(k)]^T = [A_5 x + B_5 u] \delta_5 (1 - \delta_6) \end{aligned}$$

4. Transform the above proposition in inequalities according to the rules defined in [7].
5. Define the constraints on state and input variables according to the state space partitioned.

Based on the above steps, the matrices of the system (7) were determined. The compilation of the MLD model (7) was realized by using the software package Hysdel [8].

The evolutions of the states of nonlinear system (1) and MLD system (7) are presented in figures 2, 3 and 4.

The below evolutions were obtained using the following parameters:

$$T = 0.1 \text{ h}, \quad F(k) = \begin{cases} 0.05 & \text{for } 0 \leq k < 100 \\ 0.125 & \text{for } 100 \leq k < 200 \\ 0.175 & \text{for } 200 \leq k < 300, \\ 0.215 & \text{for } 300 \leq k < 400 \\ 0.24 & \text{for } 400 \leq k \leq 500 \end{cases}$$

and the initial states on each interval are exactly the 5 operational states, i.e.

$$\begin{cases} [\bar{X} \ \bar{S}_1 \ \bar{S}_2]_1^T & \text{for } 0 \leq k < 100 \\ [\bar{X} \ \bar{S}_1 \ \bar{S}_2]_2^T & \text{for } 100 \leq k < 200 \\ [\bar{X} \ \bar{S}_1 \ \bar{S}_2]_3^T & \text{for } 200 \leq k < 300 \\ [\bar{X} \ \bar{S}_1 \ \bar{S}_2]_4^T & \text{for } 300 \leq k < 400 \\ [\bar{X} \ \bar{S}_1 \ \bar{S}_2]_5^T & \text{for } 400 \leq k \leq 500 \end{cases}$$

Also, for parametric disturbances $S_{1,in}$ and $S_{2,in}$ the waveforms presented in figure 5 and 6 were used.

From the figures 2÷4 we observe that the evolutions of

states of the MLD model are similar with those of the initial nonlinear model.

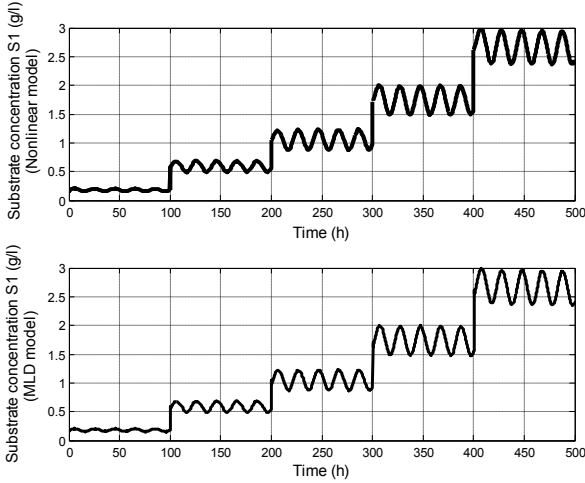


Fig. 2. Evolution of the substrate concentration S_1

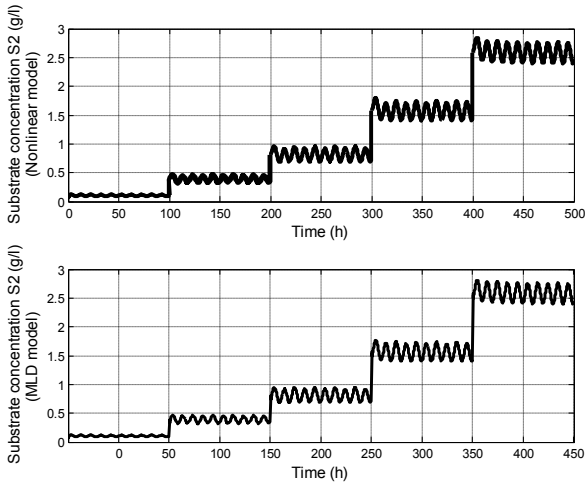


Fig. 3. Evolution of the substrate concentration S_2

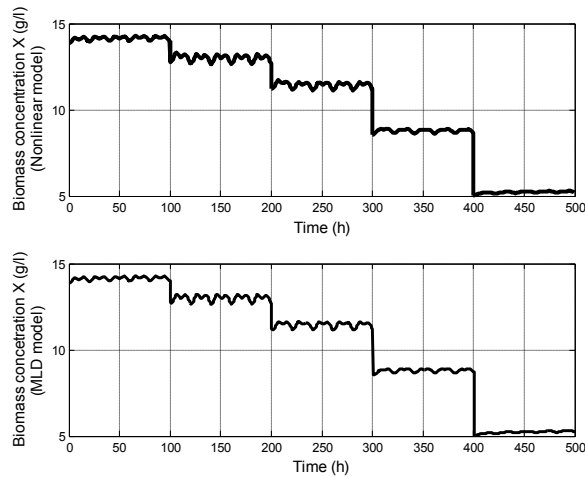


Fig. 4. Evolution of the biomass concentration X

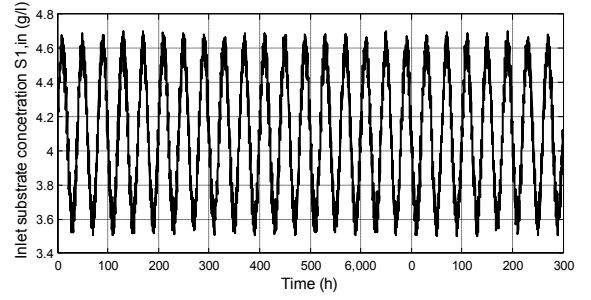


Fig. 5. Inlet substrate concentration $S_{1,in}$

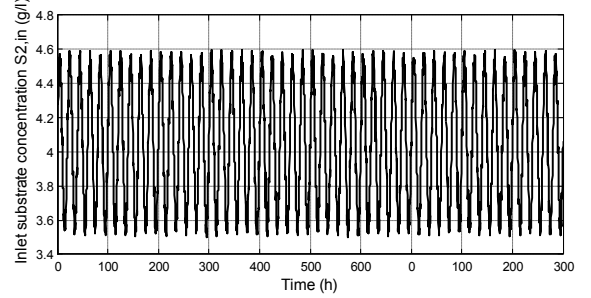


Fig. 6. Inlet substrate concentration $S_{2,in}$

IV. PREDICTIVE CONTROL

The model predictive control uses a discrete-time model of the system in order to obtain a prediction of its future behavior, by applying a set of input sequence to a model, taking into account constraints on state, outputs and inputs [9].

For the hybrid system described by MLD formalism, a mixed integer predictive control law can be defined [10]:

$$\begin{aligned} \min_{\{u, \delta, z\}_0^{N-1}} J(\{u, \delta, z\}_0^{N-1}, x(t)) = & \\ = & \|Q_{xN}(x(N|t) - x_r)\|_p + \sum_{k=1}^{N-1} \|Q_x(x(k) - x_r)\|_p + \\ & + \sum_{k=1}^{N-1} \|Q_u(u(k) - u_r)\|_p + \|Q_z(z(k|t) - z_r)\|_p + \|Q_y(y(k|t) - y_r)\|_p \end{aligned} \quad (8)$$

$$\text{subj.to} \begin{cases} x(0|t) = x(t) \\ x(k+1|t) = \bar{A}x(k|t) + \bar{B}_1u(k) + \bar{B}_2\delta(k|t) + \bar{B}_3z(k|t) \\ y(k|t) = \bar{C}x(k|t) + \bar{D}_1u(k) + \bar{D}_2\delta(k|t) + \bar{D}_3z(k|t) \\ \bar{E}_2\delta(k|t) + \bar{E}_3z(k|t) \leq \bar{E}_1u(k) + \bar{E}_4x(k|t) + \bar{E}_5 \\ u_{\min} \leq u(t+k) \leq u_{\max}, \quad k = 0, 1, \dots, N-1 \\ x_{\min} \leq x(t+k|t) \leq x_{\max}, \quad k = 1, \dots, N-1 \\ y_{\min} \leq y(t+k) \leq y_{\max}, \quad k = 0, 1, \dots, N-1 \\ S_x x(N|t) \leq T_x \end{cases}$$

where N is the optimal control interval, $x(k|t)$ is the state predicted at time $t+k$, $u(t+k)$ is the input at time $t+k$, $x(0|t)$ is the initial state, $u_{\min}, u_{\max}, y_{\min}, y_{\max}, x_{\min}$ and $x_{\max}$ are bounds on the inputs, outputs, and states, respectively, $\{x : S_x x \leq T_x\}$ is a final polyhedral subset of the

state-space $\mathfrak{R}^n$, p is norm ($p=1, 2$ or ∞), x_r, u_r, z_r and y_r are desired references for the state, input, and auxiliary variables, Q_{xN}, Q_x, Q_u, Q_z and Q_y are weighted matrices.

If the optimal solution $\{u_t^*(0), \dots, u_t^*(N-1), \delta_t^*(0), \dots, \delta_t^*(N-1), z_t^*(0), \dots, z_t^*(N-1)\}$, for the above optimization problem exists, then in accordance with the receding horizon theory, only the first value $u(t)=u_t^*(0)$ is used. At the next step a new optimization problem is solved and a new optimal solution is determined.

When $p=1$ or ∞ , the optimization problem is a mixed integer linear programming problem (MILP), and when $p=2$ the problem becomes a problem of mixed integer quadratic programming (MIPQ).

For the studied process we suppose that the control variable is the dilution rate and the output variable were substrate concentration S_1 and substrate concentration S_2 .

The presence of parametric disturbances requires augmentation of the discretized system (6) with 2 additional states:

$$\begin{aligned}\tilde{x}(k+1) &= \tilde{x}(k) + T(\tilde{A}_i \tilde{x}(k) + \tilde{B}_i u(k)) \\ \tilde{y}(k) &= \tilde{C}_i \tilde{x}(k) \\ \text{for } [X, S_1, S_2] &\in R_i, \quad i=1 \dots 5\end{aligned}\quad (9)$$

where,

$$\tilde{x} = \begin{bmatrix} x \\ w \end{bmatrix}, \quad \tilde{A}_i = \begin{bmatrix} A_i & D_i \\ O_{2 \times 3} & O_{2 \times 2} \end{bmatrix}, \quad \tilde{B}_i = \begin{bmatrix} B_i \\ O_{2 \times 1} \end{bmatrix}, \quad \tilde{C}_i = [C_i \quad O_{2 \times 2}].$$

$O_{m \times n}$ is an $m \times n$ zero matrix.

The optimization problem may be formulated by:

$$\min_{\{u, \delta, z\}_{0}^{N-1}} J(\{u, \delta, z\}_{0}^{N-1}, \tilde{x}(t)) = \|Q_y(y(k|t) - y_r)\|_p \quad (10)$$

$$\text{subj.to} \begin{cases} \tilde{x}(0|t) = \tilde{x}(t) \\ \tilde{x}(k+1|t) = \hat{A}\tilde{x}(k|t) + \hat{B}_1 u(k) + \hat{B}_2 \delta(k|t) + \hat{B}_3 z(k|t) \\ y(k|t) = \hat{C}\tilde{x}(k|t) + \hat{D}_1 u(k) + \hat{D}_2 \delta(k|t) + \hat{D}_3 z(k|t) \\ \hat{E}_2 \delta(k|t) + \hat{E}_3 z(k|t) \leq \hat{E}_1 u(k) + \hat{E}_4 \tilde{x}(k|t) + \hat{E}_5 \\ u_{\min} \leq u(t+k) \leq u_{\max}, \quad k=0, 1, \dots, N-1 \\ \tilde{x}_{\min} \leq \tilde{x}(t+k|t) \leq \tilde{x}_{\max}, \quad k=1, \dots, N-1 \\ y_{\min} \leq y(t+k) \leq y_{\max}, \quad k=0, 1, \dots, N-1 \\ S_x \tilde{x}(N|t) \leq T_x \end{cases}$$

V. SIMULATION RESULTS

The references S_1^* and S_2^* are chosen so they reflect biomass consumption needs of the time horizon considered.

For optimization problem (10) the following parameters

were defined: $Q_y = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$, $N=3$, $T=0.1h$ and $p=\infty$.

In figures 7, 8 and 9 we presented the evolution of substrate concentration S_1 , substrate concentration S_2 respectively the command applied to the system (dilution rate F).

The simulation time was fixed, in both cases at $t=400h$. Also, the parametric disturbances $S_{1,in}$ and $S_{2,in}$, in both studied cases have profiles presented in figures 10 and 11.

The optimization problems were solved through the Hybrid toolbox software [9].

We observe that the outputs follow the imposed references.

In figure 12 is represent the evolution of the biomass concentration X of nonlinear initial model.

Figure 13 represents the space partitions where the system works. We remark that in this control case, for the input considered parameters only the first 3 partitions of full state space (R_1, R_2 and R_3) are covered.

VI. CONCLUSIONS

The control of biotechnological processes is considered an important and challenging problem.

In this paper we studied the modelling and control of a bioprocess using hybrid and predictive control theory.

Starting from the nonlinear model of the studied process, multiple linear models were defined by subdividing the operating domain in small operating regimes, which covers all the considered state space.

MLD formalism was used to describe the hybrid feature of multiple linear models.

For the system described by MLD formalism, a mixed integer predictive control law was defined, in order to control the concentrations of the substrates, by computing at every step the optimal input applied to the system (dilution rate), subject to constraints on states and inputs.

The proposed control algorithm relieves its effectiveness; the outputs follow the imposed references, even in presence of non constant parametric perturbations.

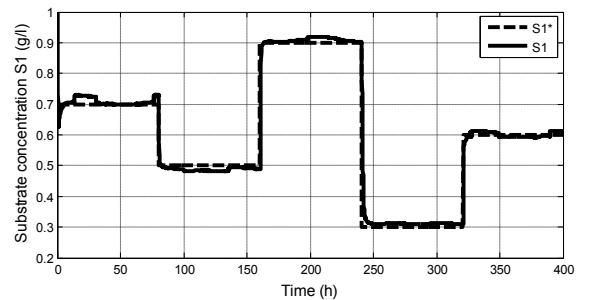


Fig. 7. Control of substrate concentration S_1 of nonlinear model (1)

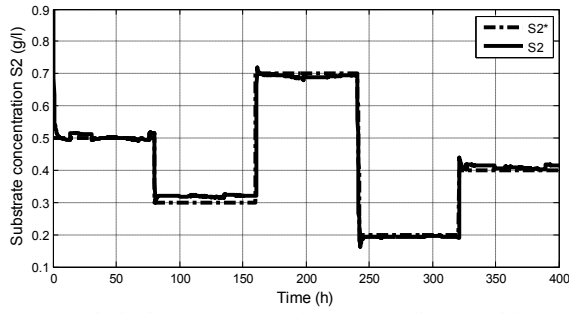


Fig. 8. Control of substrate concentration S_2 of nonlinear model (1)

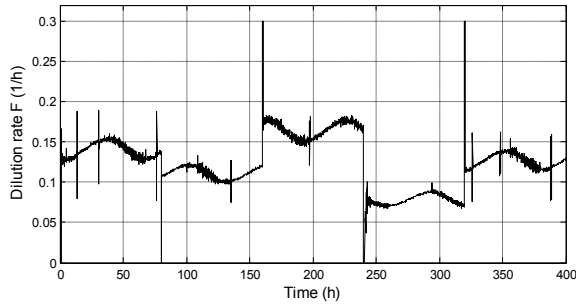


Fig. 9. The applied command (dilution rate F)

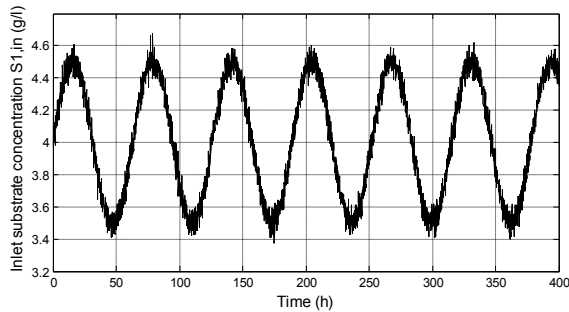


Fig. 10. Inlet substrate concentration $S_{1,in}$

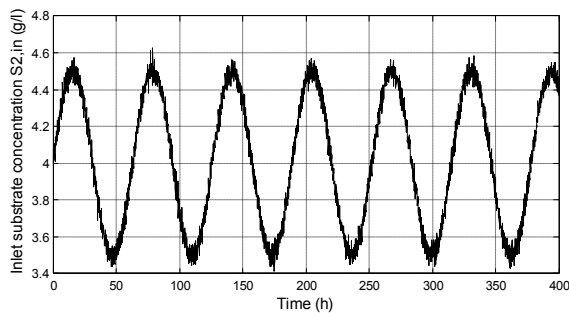


Fig. 11. Inlet substrate concentration $S_{2,in}$

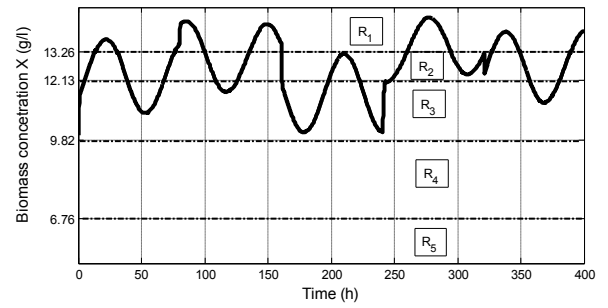


Fig. 12. Biomass concentration X of nonlinear model (1)

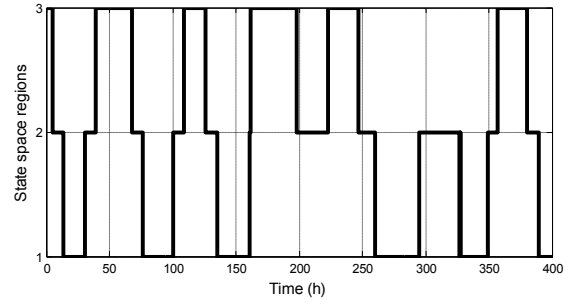


Fig. 13. State space partitions

REFERENCES

- [1] C. Komives, R.S. Parker, "Bioreactor state estimation and control." in *Current Opinion in Biotechnology*, 14, pp. 468-474, 2003.
- [2] D. Jingjing, S. Chunyue, L. Peng, "Modeling and control of a continuous stirred reactor based on a mixed logical dynamical model" *Chinese J. Chem. Eng.*, 15(4), pp. 533-538, 2007.
- [3] O. Roeva, T. Pencheva, S. Tyonkov, M. Arndt, B. Hitzmann, S. Kleist, G. Miksch, K. Friehs, "Multiple model approach to modelling of *Escherichia coli* fed-batch cultivation extracellular production of bacterial phytase", in *Electronic Journal of Biotechnology*, vol. 10, no.4, 2007.
- [4] M. Morari, J.H. Lee, "Model predictive control: past, present and future." in *Computers and Chemical Engineering*, 23(4-5), pp. 667-682, 1999.
- [5] F. Borelli, "Constrained optimal control of linear and hybrid systems", *Lecture Notes in Control and Information Sciences*, Springer, 2003.
- [6] K. Turki, O. Boubaker, " H_∞ gain scheduling control of a non linear bioprocess: a multiple model approach" in *AIML Journal*, vol. 5, 2005.
- [7] A. Bemporad, M. Morari "Control of system integrating logics, dynamics, and constraints" *Automatica*, 35(3):407-427, 1999.
- [8] F. D. Torrisi, A. Bemporad, G. Bertini, P. Hertach, D. Jost, D. Mignone, *Hysdel 2.0.5 - User manual*.
- [9] M. Lazar. "Model Predictive Control of Hybrid Systems: Stability and Robustness", Phd. Thesis, 2006
- [10] A. Bemporad, *Hybrid toolbox - User's guide*, 2010

Hybrid Control for a Hyper-Redundant Robot

Stoian Viorel, Nițulescu Mircea, Mănoiu – Olaru Sorin

Abstract—A new control algorithm for a hyper-redundant manipulator, which moves in a labyrinth and can touch the boundaries of the niche, is presented in this paper. The collisions or contacts are part of the task. The support-points from the walls help with the moving process. Initially, a study of the hyper-redundant locomotion by creeping and climbing is achieved.

Also, the dynamic model by Lagrange method of the hyper-redundant robotic structure is established. Afterwards, a hybrid position/force controller for the hyper-redundant locomotion is proposed. Finally, the simulations of the hyper-redundant locomotion by climbing are presented.

I. INTRODUCTION

A Hyper – Redundant Manipulator (HRM) has the potential ability to perform non-conventional tasks, such as moving in highly constrained environment or grasping a various sizes and shapes of objects [1].

The term hyper-redundant refers to redundant manipulators with a very large or infinite number of degree of freedom. These manipulators are analogous in morphology to snakes, elephant trunks, and tentacles [2]. Is used the term “wave” to describe the mechanism deformations associated with these hyper-redundant locomotion gaits (repetitive sequences of mechanism deformations that cause net motion of the robot). Two highly idealized forms of biological locomotion are stationary wave locomotion and traveling wave locomotion.

The HRMs are categorized into 3 types of mechanisms: a large number of rigid links in series (a chain of discrete links) [3], [4], [5] etc., a cascade of modules composes of redundantly actuated parallel rigid platforms [6], [7] and truly flexible physical structures tentacle-like [6], [8]. This paper presents a serial rigid link robot manipulator with hyper degrees of freedom.

One of the cardinal goals of the robotic researchers is to provide control algorithms that allow robots to move in an

unknown environment without collisions with the unstructured obstacles. Generally, in specific literature this problem is addressed as the collision avoidance problem [6], [9], [10], [11]. In this paper the hyper-redundant manipulator is analyzed in following situation: his job is to infiltrate, to creep through dropped walls after a cataclysm in a humanitarian action or for inspection in highly constrained environments.

The mechanics are for a snake-like structure but with serial rigid links. In this case is necessary a compliant motion by touching of objects in many points of the manipulator like a human been in a very narrow cave which press with his shoulder the wall for his stability and with his hand push up a block (gravestone, manhole) or for climbing. The collisions or contacts are a part of the task.

The support points from the walls help to the moving process. For example, in fig. 1 (for simplicity, a planar structure) the robot fixes its i -link on the walls pressing the link's ends by F_i and F_{i+1} forces. In this situation the upper part (links $i+1, i+2 \dots n$) can be relaxed and inferior part (links $1, 2 \dots i-1$) can be constricted. This process allows a creeping in horizontal plane or a climbing in vertical plane.

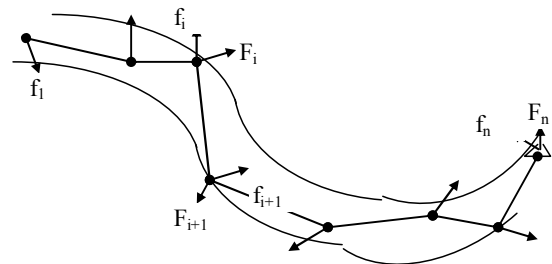


Fig. 1. The HRM in a labyrinth

II. HYPER-REDUNDANT LOCOMOTION

Into a spatial and complex labyrinth is necessary to combine two kinds of actions: creeping and climbing. These actions, called “the hyper-redundant locomotion” are defined to mean motions resulting in net displacements of hyper-redundant robots due to the internally induced bending of the mechanism (Chirikjian and Burdick 1991). For simplicity, in

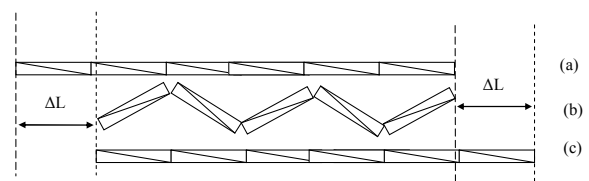


Fig. 2. The hyper-redundant locomotion

Manuscript received June 17, 2011. This work was supported in part by the strategic grant POSDRU/88/1.5/S/50783, Project ID 50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resource Development 2007 – 2013

Mănoiu Olaru Sorin is with the University of Craiova, Faculty of Automation, Computers and Electronics, Bulevardul Decebal, Nr. 107, 200440, Craiova, ROMANIA Tel./Fax: +40.251.436999 (e-mail: manoiusorin2006@yahoo.com).

Mircea Nițulescu is with the University of Craiova, Faculty of Automation, Computers and Electronics, Bulevardul Decebal, Nr. 107, 200440, Craiova, ROMANIA Tel./Fax: +40.251.436999 (e-mail: nitulescu@robotics.ucv.ro).

Stoian Viorel is with the University of Craiova, Faculty of Automation, Computers and Electronics, Bulevardul Decebal, Nr. 107, 200440, Craiova, ROMANIA Tel./Fax: +40.251.436999 (e-mail: stoian@robotics.ucv.ro).

this paper is studied a planar structure, but the ideas are generalized to the spatial case. Figure 2 presents the hyper-redundant locomotion with amplitude varying wave that remains stationary with respect to body coordinates. ΔL represents the step for one cycle of motion. In (a) the robot there is in the initial position, in (b) it has the upper end strongly fixed in terrain, the inferior end is discharged and it

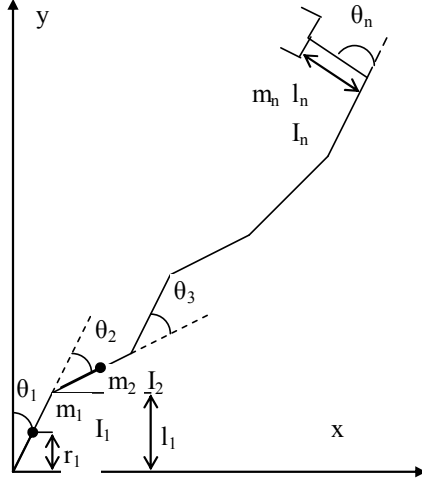


Fig. 4. The HRM structure.

moves ahead. In (c) the inferior end is strongly fixed in terrain and the upper end in discharged. Now, the hyper-redundant robot can relax ahead and it has again the initial position, but moved with the displacement ΔL .

In [12] is presented the hyper-redundant locomotion with traveling wave with constant amplitude. The two ends of the robot have the same functions like the ones from Figure 2. Here there is only one wave that travels the length of the body from the bottom of the mechanism to the upper end.

In fig. 3 is presented a climbing process by the same hyper-redundant robotic structure.

III. DYNAMIC MODEL

In fig. 4 is presented a hyper-redundant robotic structure for establish a dynamic model by Lagrange method. With the notations from that figure are passed get through all phases of the algorithm. The coordinates $x(q)$ and $y(q)$ are:

$$x_k = \sum_{i=1}^n l_i \cos \left(\sum_{j=1}^i \theta_j \right) \quad (1)$$

$$y_k = \sum_{i=1}^n l_i \sin \left(\sum_{j=1}^i \theta_j \right) \quad (2)$$

The velocities of the x and y are:

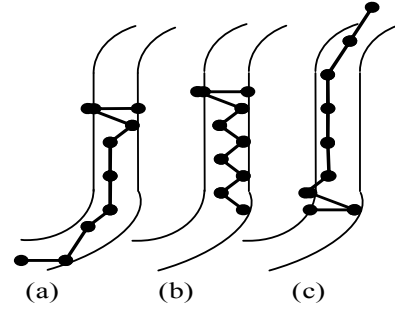


Fig. 3. The climbing process.

$$\dot{x}_k = - \sum_{i=1}^k \left\{ l_i \left[\sin \left(\sum_{j=1}^i \theta_j \right) \right] \left(\sum_{j=i}^k \dot{\theta}_j \right) \right\} \quad (3)$$

$$\dot{y}_k = \sum_{i=1}^k \left\{ l_i \left[\cos \left(\sum_{j=1}^i \theta_j \right) \right] \left(\sum_{j=i}^k \dot{\theta}_j \right) \right\} \quad (4)$$

The kinetic energy and the potential energy are:

$$E_c(q, \dot{q}) = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (5)$$

$$E_p(q) = \sum_{i=1}^n E_{p_i} = g \sum_{i=1}^n m_i \left[\sum_{j=1}^i l_j \sin \left(\sum_{k=1}^j \theta_k \right) \right] \quad (6)$$

where $M(q)$ is the inertial matrix and

$$q = [\theta_1, \theta_2, \dots, \theta_n]^T \quad (7)$$

The Lagrange function is:

$$L(q, \dot{q}) = E_c(q, \dot{q}) - E_p(q) = \frac{1}{2} \sum_{i,j=1}^n M_{ij}(q) \dot{q}_i \dot{q}_j - E_p(q) \quad (8)$$

Now, we can apply Lagrange's equation are:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = T_i \quad (9)$$

The two terms of the Lagrange's equation are:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) = \frac{d}{dt} \left(\sum_{j=1}^n M_{ij} \dot{q}_j \right) = \sum_{j=1}^n (M_{ij} \ddot{q}_j + \dot{M}_{ij} \dot{q}_j) \quad (10)$$

$$\frac{\partial L}{\partial q_i} = \frac{1}{2} \sum_{j,k=1}^n \frac{\partial M_{kj}}{\partial q_i} \dot{q}_k \dot{q}_j - \frac{\partial E_p}{\partial q_i} \quad (11)$$

Now, the Lagrange's equation is:

$$\sum_{j=1}^n M_{ij}(q) \ddot{q}_j + \sum_{j,k=1}^n \Gamma_{ijk} \dot{q}_j \dot{q}_k + \frac{\partial E_p}{\partial q_i}(q) = T_i \quad (12)$$

where:

$$\Gamma_{ijk} = \frac{1}{2} \left(\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ik}(q)}{\partial q_j} + \frac{\partial M_{kj}(q)}{\partial q_i} \right) \quad (13)$$

If we define the Coriolis matrix as:

$$\begin{aligned} C_{ij}(q, \dot{q}) &= \sum_{k=1}^n \Gamma_{ijk} \dot{q}_k = \\ &= \frac{1}{2} \sum_{k=1}^n \left(\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ik}(q)}{\partial q_j} + \frac{\partial M_{kj}(q)}{\partial q_i} \right) \dot{q}_k \end{aligned} \quad (14)$$

Then $C(q, \dot{q})\dot{q}$ is the Coriolis and centrifugal forces vector. The third term is a position term representing loading due to gravity.

$$\frac{\partial E_p}{\partial q_i}(q) = G(q) \quad (15)$$

T_i represents the outer forces:

$$T_i = T_i - B(\dot{q}) - \tau \quad (16)$$

where T_i are the actuators forces (inner generalized forces), B represents the friction forces and τ represents the forces from joints due to the forces acting on the tool tip or on the another points from the robotic structure due to a load or contact with the environment.

With the above-mentioned notations we have the dynamic model of the hyper-redundant robotic structure from Figure 4:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + B(\dot{q}) + G(q) + \tau = T \quad (17)$$

In this paper, by “position” we mean “position and orientation”, and by “force” we mean “force and torque”.

IV. HYBRID CONTROL

During creeping in tunnels or labyrinths the robot comes in contact with environment; therefore, interaction forces develop between the robot and environment. Consequently, these interaction forces, as well as the position of the points from the robot, must be controlled. A number of methods for obtaining force information exist: motor currents may be measured or programmed, motor output torques may be measured and joint-mounted sensors may be used. There are two primary methods for producing compliant motion: a

passive mechanical compliance built into the manipulator, or an active compliance implemented in the software control loop, force control. Passive compliance offers some performance advantages, but the force control method offers the advantage of programmability. This allows the manipulator to use the particular form of compliance necessary for the particular application. Throughout this paper is assumed that the desired velocity and force trajectories, which are commanded by the controllers, are consistent with the model of the environment [13]. If this is not the case, it may be possible to modify the desired velocity and force trajectories to be consistent with the model of the environment. The compliant motion occurs when the position of a robot is constrained by the task. The so-called hybrid position/force controller [14], [15] can be used for tracking position and force trajectories simultaneously. The basic concept of the hybrid position/force controller is to decouple the position and force/control problems into subtasks via a task space formulation because the task space formulation is valuable in determining which directions should be force or position controlled. Figure 5 presents a hybrid position/force controller.

$$x_d = \begin{bmatrix} x_d \\ y_d \\ z_d \end{bmatrix}; x = \begin{bmatrix} x \\ y \\ z \end{bmatrix}; \Delta x = x_d - x; \quad (18)$$

$$F_D = \begin{bmatrix} F_{xd} \\ F_{yd} \\ F_{zd} \end{bmatrix}; F = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}; \Delta F = F_D - F$$

$$q = [q_1, q_2, \dots, q_n]^T; f = [f_1, f_2, \dots, f_n]^T \quad (19)$$

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; S = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & S_z \end{bmatrix} \quad (20)$$

where S defines the compliance selection matrix.

$S_i=0$ for position control

$S_i=1$ for force control

$$T_{Ff} = J^T; T_{Ff}^{-1} = (J^{-1})^T \quad (21)$$

Where $J(q)$ is the Jacobian matrix and T_{Ff} is a metrical operator which make the transformation between the forces F and the forces F . $x=f(q)$ is a relation which represent the forward kinematics. For position controller is advantageously to use a PID controller and force a PD controller.

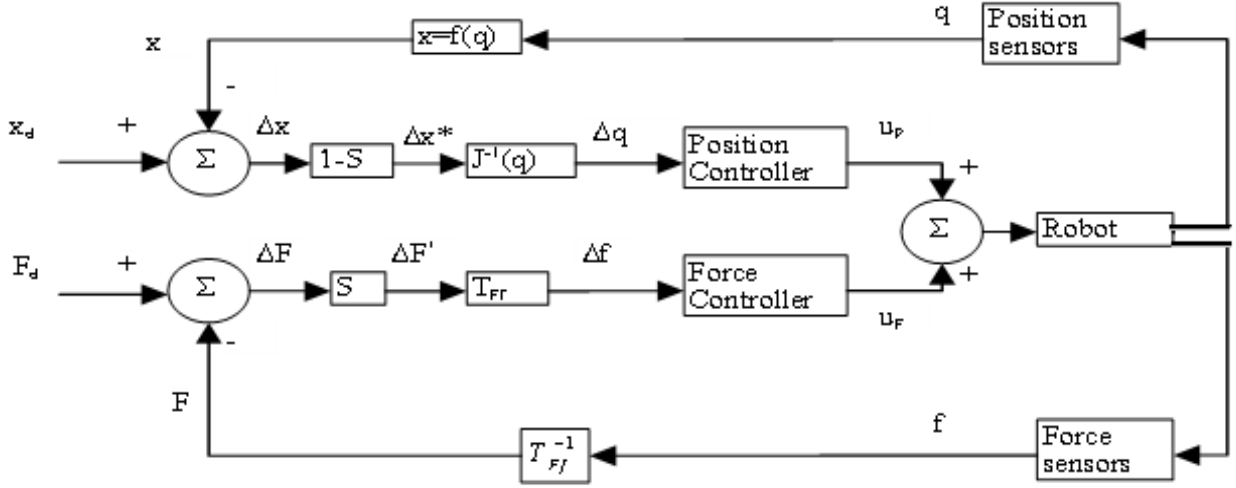


Fig. 5. The Hybrid Position / Force Controller.

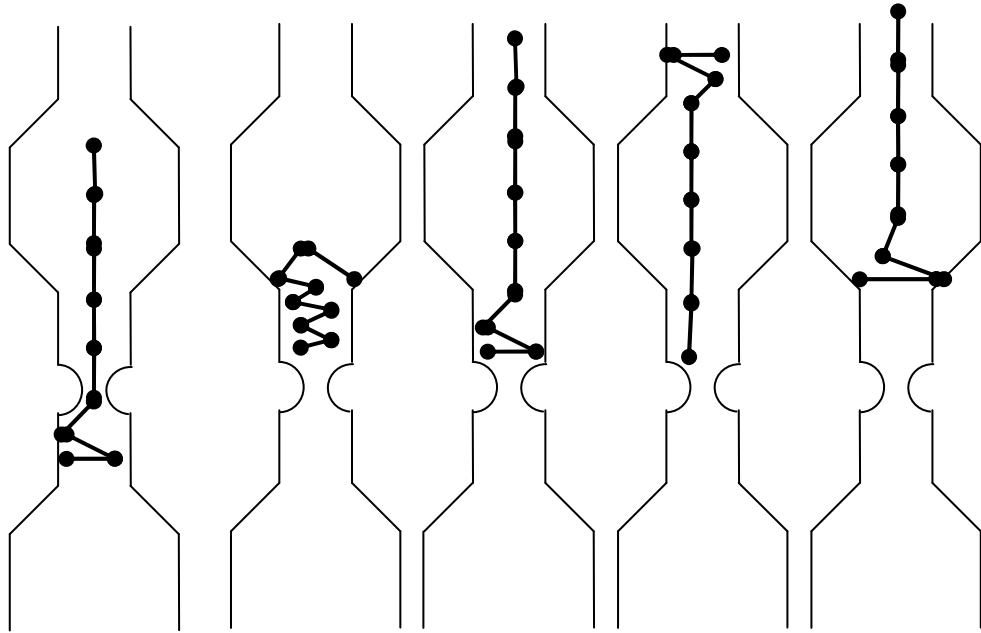


Fig. 6. Simulation for a climbing action.

$$u_p = k_1 \Delta q + k_2 \Delta \dot{q} + k_3 \int_0^t \Delta q dt \quad (22)$$

$$u_F = k_4 \Delta f + k_5 \Delta \dot{f} \quad (23)$$

The general control law u_i is:

$$u_i = \sum_j \left[\Psi_{ij} S_j \Delta f_i + \Xi_{ij} (1 - S_j) \Delta q_i \right] \quad (24)$$

where Ψ_{ij} is a force control law and Ξ_{ij} is a position control law.

Figure 6 presents a simulation for the climbing operation through a vertical labyrinth.

Any segments of the hyper-redundant robot are used for

the anchorage of the structure on the walls of the vertical labyrinth while the others segments are moving up.

The hyper-redundant locomotion by creeping is analogous.

V. CONCLUSIONS

A new control algorithm for a hyper-redundant manipulator, which moves in a labyrinth and can touch the boundaries of the niche is presented in this paper. The collisions or contacts are part of the task. The support-points from the walls help with the moving process. Initially, a study of the hyper-redundant locomotion by creeping and climbing is achieved. Also, the dynamic model by Lagrange method of the hyper-redundant robotic structure is established. Afterwards, a hybrid position/force controller for the hyper-redundant locomotion is

proposed. Finally, the simulations of the hyper-redundant locomotion by climbing are presented.

ACKNOWLEDGMENT

This work was partially supported by strategic grant POSDRU/88/1.5/S/50783, Project ID 50783 (2009), co-financed by the European Social Fund – Investing in People, within the Sectoral Operational Programme Human Resource Development 2007 – 2013.

REFERENCES

- [1] S. Hirose, "Biologically Inspired Robots: Snake-like Locomotors and Manipulators", Oxford Univ. Press, 1993.
- [2] Chirikjian, G.S. and J.W. Burdick, 1992. "A Geometric Approach to Hyper-Redundant Manipulator Obstacle Avoidance", Transactions of the ASME, Journal of Mechanical Design, vol 114, pp 580-585.
- [3] Kobayashi, H. and S. Ohtake, 1995. "Shape Control of Hyper-Redundant Manipulator", Proceedings of the 1995 IEEE International Conference on Robotics and Automation, (ICRA'95), Nagoya, Japan, vol 2, pp. 2803-2808.
- [4] Mochiyama, H.; E. Shimemura; and H. Kobayashi. 1996. "Control of Serial Rigid Link Manipulators with Hyper Degrees of Freedom: Shape Control by a Homogeneously Decentralized Scheme and its Experiment", Proceedings of the 1995 IEEE International Conference on Robotics and Automation (ICRA'96), Minneapolis, Minnesota, vol. 2, pp. 2877-2882
- [5] Paljug, E.; T. Ohm; and S. Hayati. 1995. "The JPL Serpentine Robot: a 12 DOF System for Inspection", Proceedings of the 1995 IEEE International Conference on Robotics and Automation, (ICRA'95), Nagoya, Japan, vol. 3, pp. 3143-3148.
- [6] Chirikjian, G.S. and J.W. Burdick, 1993. "Kinematically Optimal Hyper-Redundant Manipulator Configurations", IEEE Transactions on Robotics and Automat., vol II, No.6, pp. 794-805.
- [7] Zanganeh, K. E. and J. Angeles. 1995. "The Inverse Kinematics of Hyper-Redundant Manipulators Using Splines", Proceedings of the 1995 IEEE International Conference on Robotics and Automation (ICRA'95), Nagoya, Japan, vol 2, pp. 2797-2902.
- [8] Ivanescu, M. and V. Stoian. 1998. "A Control System for Cooperating Tentacle Robots", Proceedings of the 1998 IEEE International Conference on Robotics and Automation (ICRA'98), Katholieke Universiteit, Leuven, Belgium, vol 2, pp. 1540-1545.
- [9] Colbaugh, R.; H. Seraji; and K.L. Glass. 1989. "Obstacle Avoidance for Redundant Robots Using Configuration Control", J. of Robotic Systems, vol. 6, pp 721 – 744.
- [10] Khatib, O. 1986. "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", Int. J. of Robotic Research, vol. 5, pp. 90 – 98.
- [11] Zghal, H.; R.V. Dubey; and J.A. Euler. 1992. "Collision Avoidance of a Multiple Degree of Redundancy Manipulator Operating Through a Window", Transaction of the ASME, Journal of Dynamic Systems, Measur., and Control, vol 114, pp. 717-721
- [12] Chirikjian, G.S. and J.W. Burdick, 1991. "Kinematics of Hyper-Redundant Robot Locomotion with Applications to Grasping", Proceedings of the 1991 IEEE International Conference on Robotics and Automation (ICRA'91), Sacramento, California, vol I, pp 720-725.
- [13] Lipkin, H. and J. Duffy. 1988. "Hybrid Twist and Wrench Control for a Robot Manipulator", Trans. ASME J. Mechan. Transmissions Autom. Design, vol 110, pp. 138-144.
- [14] Chae, A.; C. Atkenson; and Hollerbach J. 1988. "Model-Based Control of a Robot Manipulator", Cambridge, MA, MIT Press.
- [15] Raibert, M. and J. Craig. 1981. "Hybrid Position/Force Control of Manipulators", J. Dyn. Syst. Meas. Control, vol. 102, pp. 126-133.

System Identification for Nonlinear FOPDT Model with Input-Dependent Dead-Time

Zhen Sun and Zhenyu Yang

Abstract—An on-line iterative method of system identification for a kind of nonlinear FOPDT system is proposed in the paper. The considered nonlinear FOPDT model is an extension of the standard FOPDT model by means that its dead time depends on the input signal and the other parameters are time dependent. In order to identify these parameters in an on-line manner, the considered system is discretized at first. Then, the nonlinear FOPDT identification problem is formulated as a stochastic Mixed Integer Non-Linear Programming problem, and an identification algorithm is proposed by combining the Branch-and-Bound method and Mean Least Square technique. The proposed method is tested through a number of numerical examples. As shown in the case studies, the proposed method can provide a reasonable performance in the parameters identification with regard to the reasonable accuracy. The result also showed that the dead time change according to the input can also be reasonable tracked.

I. INTRODUCTION

To develop a mathematical model for a system or process is always the first step need to be taken in the design of the system with controllers. It is well known that a First Order Plus Dead Time (FOPDT) system can model many industrial processes [6], [8], [11].

The FOPDT model has three different parameters, named system gain, time constant and dead time (time delay). These parameters are often set as constants in the whole system running for the standard model. In reality, during the system running, the system may not stay unchanged but vary according to the time. Thereby, in order to still keep the simplicity of FOPDT model, meanwhile be able to cope with the time-varying characteristics, a kind of nonlinear FOPDT model in which the time varying parameters of the system can be describe is proposed in [5], [7], [10]. The considered nonlinear FOPDT model is an extension of the standard FOPDT model by means that both system's gain and time constant can be changed during the system running. This nonlinear FOPDT model is generated by using a linearized method to a nonlinear model. In our previous work [12], a new type of explicit nonlinear FOPDT model is proposed as well, named Time-Varying FOPDT (TV-FOPDT) model, which is used to model the superheat dynamic in a super-market refrigeration system. The TV-FOPDT model is an extension of the standard FOPDT by allowing the system parameters (system gain, time constant and time delay) to be time dependent variables.

Sometimes, in the practical system, the parameters may depend on the other variables besides the time. For example,

considering the system of the superheat in a refrigeration system, the time that used in the process of changing the evaporation temperature depends on some condition, such as the refringent filling of the evaporator. The more the refringent is filled in the evaporator, the more time needed. If this refringent filling is taken as the system input, the time that used in the temperature change process, which could be taken as the time delay for the system changing, may depend on this input refringent filling [6], [11], [13]. In order to express this property of system, this paper will extend the TV-FOPDT model to a more general one where the dead time (time delay) is also input dependent and the model in the paper can be called as a kind of TV-FOPDT model with input dependent dead time.

It is observed that the nonlinear FOPDT models have been widely used in a number of nonlinear control applications [1], [5], [7]. Some methods have been already adopted to make the system identification for this kind of model. In [7], [10], a nonlinear FOPDT model is proposed by linearizing the nonlinear system at a number of different operating points, so the parameters of the obtained FOPDT are operating-point dependent [7]. But this method is not recommended for online identification if a precise nonlinear model can not be obtained beforehand. An on-line nonlinear FOPDT identification method is proposed in [5] by using the so-called longrange predictive identification method. However, the formulated system identification leads to a nonlinear optimization problem due to the unknown time-dependent time delay. Therefore, four different potential time delay scenarios are assumed, before converting the nonlinear optimization problem into a Least-Square (LS) problem using the spectral factorization technique. The assumption of time delays limits the proposed method in [5] to be applied for any other systems except for these two specific patient cases they have studied.

In order to deal with these difficulty, an on-line system identification method for the TV-FOPDT model was proposed based on a formulated Mixed Integer Non-Linear Programming (MINLP) problem in the previous work [12]. The data of sampled input and output is adopted to make the system identification in this method which could be, to some extent, regarded as a kind of black box identification method. The method adopts the Branch-and-Bound(BB) method [4] and LS technique to cope with the formulated Mixed Integer Non-Linear Programming (MINLP) problem and takes the optimal value as the parameters estimation at each sampling step. Then, the a moving window technique is applied to update the parameters identification result. In this paper, as

Zhen Sun and Zhenyu Yang are with the Department of Energy Technology, Aalborg University, Esbjerg Campus, Niels Bohrs Vej 8, Esbjerg 6700, Denmark, zhen@et.aau.dk, yang@et.aau.dk

the model adopted is added with noise in the measurement, the previous identification method is extended to cope with the output signal with stochastic feature. The parameters identification for this TV-FOPDT model with input dependent dead time is reformulated as a stochastic MINLP problem and it is solved using BB method and Mean Least Square (MLS) method.

The proposed method is tested through a numerical example. The rest of the paper is organized in the following: Section II formulates the considered identification problem; Section III introduces the proposed method for IDTV-FOPDT model identification; Section IV discusses the testing result for numerical example; and finally, we conclude the paper in Section V.

II. PROBLEM FORMULATION

The system described in a Time-Varying FOPDT (TV-FOPDT) model with input depended dead time is defined in the following.

$$Y(s) = G^{u(t),t}(s)U(s), \quad (1)$$

with transfer function

$$G^{u(t),t}(s) = \frac{K^t}{T_p^t s + 1} \exp^{-T_d^{u(t),t} s}. \quad (2)$$

And the measurement is

$$x(s) = y(s) + \omega(s). \quad (3)$$

where $Y(s)/U(s)$ is the Laplace-transform of the system output/input $y(t)/u(t)$. K^t , T_p^t and $T_d^{u(t),t}$ are the system gain, time constant, and time delay (dead-time), respectively. Different with the standard FOPDT model, all these system parameters can be time-dependent, especially the time delay can also depend on the input signal. This dependence feature is represented by the corresponding subscript. $x(s)$ is the measured output of the system and $\omega(s)$ is the noise in the output measurement, which is assumed as a Gaussian process with 0 mean and variance Q .

The TV-FOPDT with input depended time delay system identification can be described as: *how to correctly estimate system parameters K^t , T_p^t and $T_d^{u(t),t}$ for a system modeled by (1) with (2) based on the data of system input and measured output (3) signals in an on-line and probabilistically optimal manner?*

III. MODEL IDENTIFICATION METHOD

The proposed identification method that is based on the MINLP problem applies what is used in our previous work [12], which consists of two steps, namely discretized problem [13] and iterative MLS Prediction algorithm. It is summarized in the following.

A. Discretized Problem

Following the similar procedure as in [13], the model (1) with (2) can be approximated by its discrete-time equivalence, i.e.,

$$Y(z) = G^{u(k),k}(z)U(z), \quad (4)$$

with

$$G^{u(k),k}(z) = \frac{K^k(1 - \alpha^k)}{z^{l^{u(k),k}}(z - \alpha^k)}.$$

Here $\alpha^k \triangleq \exp^{-\frac{T_s}{T_p^k}}$, and T_s is the sample interval. It should be noticed that K^k and T_p^k are not the same as K^t and T_p^t in (2). The former two are piecewise-constant (constant during every sampling period) timed functions, while the later two in (2) are real-timed functions. The relationship of these two description is that K^k is equal to K^t , T_p^k is equal to T_p^t at each sampling time, i.e., $K^k = K^t$ and $T_p^k = T_p^t$ when $t = kT_s$ for any k . Thereby, K^k , T_p^k are called as the k th sampled (time-varying) system gain, the k th sampled (time-varying) time constant[12]. Here $l^{u(k),k}$ is the discrete approximation of the k th sampled system delay $T_d^{u(k),k}$ (the k th sampled (time-varying) time delay, $T_d^{u(k),k} = T_d^{u(t),t}$ when $t = kT_s$ for any k), and it is defined as an integer with the property:

$$l^{u(k),k}T_s < T_d^{u(k),k} < (l^{u(k),k} + 1)T_s \quad (5)$$

Define $\beta^k \triangleq K^k(1 - \alpha^k)$, then TV-FOPDT model with input depended dead time (4) can be transferred into a difference equation model described as

$$y(k) = \alpha^k y(k-1) + \beta^k u(k - l^{u(k),k} - 1), \quad (6)$$

for $k = l^{u(k),k} + 1, l^{u(k),k} + 2, \dots \infty$.

The output measurement is not changed, but the measured output signal can only be obtained at each sampled time:

$$x(k) = y(k) + \omega(k). \quad (7)$$

Then, the original continuous-time model identification problem of (2) with parameters K^t , T_p^t and $T_d^{u(t),t}$ is converted to estimate parameter sequences of α^k , β^k and $l^{u(k),k}$ for a stochastic discrete-time system (6) based on a number of sampled input and measured output obtained by (7). This random discrete-time system identification problem is called the discreteized approximation of the original continuous-time identification problem [12].

B. Iterative MLSP method

The method is proposed in the following, which can be named as Mean Least Square Prediction (MLSP) identification method, in order to handle this system identification problem for the TV-FOPDT model with input depended dead time.

Suppose that the considered system (1) is running at k th sampling step and take N as the number of latest pair samples of the measured output and input into consideration, where N is the length of the moving data window used in each estimation step. Define $\theta^k = [\alpha^k \ \beta^k]^T$, then the parameters identification of the system (6) at the k th

sampling step can be formulated as a Stochastic Mixed Integer Non-Linear Programming (SMINLP) problem, which is defined as:

$$\min_{\substack{l^{u(k),k} : \text{positive integer} \\ \theta^k \in \Theta^k}} \mathbf{E}\{\|B_N^k - A_N^k(l^{u(k),k})\theta^k\|_2^2\}, \quad (8)$$

where B_N^k is a stack of N latest measured outputs with forgetting factor at the current k th sampling step, i.e.,

$$B_N^k \triangleq [x(k) \ \rho x(k-1) \ \dots \ \rho^{N-2}x(k-N+2) \ \rho^{N-1}x(k-N+1)]^T. \quad (9)$$

$A_N^k(l^{u(k),k})$ is a stack of N inputs and measured output with forgetting factor at the current k th sampling step which can generate B_N^k , depending on the delay parameter $l^{u(k),k}$, i.e.,

$$A_N^k(l^{u(k),k}) \triangleq \begin{pmatrix} x(k-1) & u(k-l^{u(k),k}-1) \\ \rho x(k-2) & \rho u(k-l^{u(k),k}-2) \\ \vdots & \vdots \\ \rho^{N-2}x(k-N+1) & \rho^{N-2}u(k-l^{u(k),k}-N+1) \\ \rho^{N-1}x(k-N) & \rho^{N-1}u(k-l^{u(k),k}-N) \end{pmatrix}. \quad (10)$$

Θ^k represents the possible range of θ^k , which is determined by the limits of the original system gain K^t and time constant T_p^t in (2) at the current sampling time kT_s . Here ρ is called as forgetting factor, which is used in order to decrease the effect of old data to the estimation at the current sampling time [2].

If the system has run with no time delay, or the time delay is prior knowledge, the optimization problem (8) can be simplified to a problem of the minimization of the Mean Squared Error (MSE). In general, to solve this SMINLP problem (8) may lead to some non-convex issue due to the unknown time delay $l^{u(k),k}$. But if some pre-knowledge about time delay in system can be obtained, such as the potential upper and lower limits of the time delay(s) for the entire system or each sampling step, an iterative numerical algorithm can be performed by combining the BB method [4], which is one typical method for MINLP problem, and the MLS technique for efficiently solving this SMINLP problem. The algorithm is called as an iterative MLSP algorithm, which is summarized in the following:

- **Preparation:** The upper and lower limits for system time delay(s) in terms of some integer number multiplying with sampling period need to be given. Without losing generality, we assume that $l_{\min}^{u(k),k} \leq l^{u(k),k} \leq l_{\max}^{u(k),k}$ and $l_{\min}^{u(k),k}$, $l_{\max}^{u(k),k}$ are known before the procedure. The

sampling rate T_s , sliding window length N and forgetting factor ρ need to be decided before the procedure.

- **Data collection period:** In the beginning, the algorithm only collects the sampled data until the process reaches a specific sampling step, denoted this step as k_{ini} , where $N + l_{\max}^{u(k_{ini}),k_{ini}} = k_{ini}$. It is to guarantee that there is enough data to construct matrix (9) and (10).
- **Iteration period:** The iterative identification starts from the k_{ini} step in an on-line manner. k is denoted as the sampling step and there is $k \geq k_{ini}$, a computing loop is constructed with regard to $l^{u(k),k}$ starting from $l_{\min}^{u(k),k}$ and ending at $l_{\max}^{u(k),k}$ by taking the unit increment.

- For each iteration (k) of $l^{u(k),k}$ ($l_{\min}^{u(k),k} \leq l^{u(k),k} \leq l_{\max}^{u(k),k}$), solve the LMS problem (8) and record the corresponding prediction error. The MLS method with forgetting factor is adopted in this paper. The analytical solution has the format as:

$$\begin{aligned} \hat{\theta}^k(l^{u(k),k}) &= ((A_N^k(l^{u(k),k}))^T Q^{-1} A_N^k(l^{u(k),k}))^{-1} \\ &\quad (A_N^k(l^{u(k),k}))^T Q^{-1} B_N^k, \\ Cov(\hat{\theta}^k) &= ((A_N^k(l^{u(k),k}))^T Q^{-1} A_N^k(l^{u(k),k}))^{-1}, \end{aligned} \quad (11)$$

- where $Cov(\hat{\theta}^k)$ means the covariance of $\hat{\theta}^k$.
- $(\hat{\theta}^k(l^{u(k),k}), \hat{l}^{u(k),k})$ which leads to the minimal prediction error among all iterations moving from $l_{\min}^{u(k),k}$ to $l_{\max}^{u(k),k}$, denoted as $(\theta^{k*}(l^{u(k*),k*}), l^{u(k*),k*})$, is chosen as the optimal solution for (8) at the current step.
- The estimation of the k th sampled system parameters of (4) for the current sample, T_p^k and K^k , can be obtained from $\theta^{k*}(l^{u(k*),k*}) = [\alpha^{k*} \ \beta^{k*}]^T$ by

$$T_p^k = -\frac{T_s}{\ln \alpha^{k*}} \text{ and } K^k = \frac{\beta^{k*}}{1 - \alpha^{k*}}, \quad (12)$$

and the sampled time delay $T_d^{u(k),k}$ is estimated as $l^{u(k*),k*}T_s$.

- Repeat the above steps when a (couple) new data of input and measured output is obtained.

According to the former procedure, the system identification for TV-FOPDT model with input depended dead time can be executed in an on-line manner. Note that in this system identification, only the result of the parameters estimation is focused on, so the covariance of estimated parameter which can be calculated by the second part of (11) is not recorded.

IV. NUMERICAL SIMULATION AND TEST

A number of the numerical simulations are applied to make the test of the proposed system identification method

for TV-FOPDT model with input depended time delay.

The system considered is a switching FOPDT model with input depended time delay. The time delay of the system is dependent on the input signal $u(t)$ in the manner that $T_d^{u(t),t} = 0.5u(t)$. Other parameters are set as:

$$\begin{cases} T_p^t = 1, K^t = 3, & \text{when } t < 30 \text{ seconds;} \\ T_p^t = 2, K^t = 4, & \text{when } t \geq 30 \text{ seconds.} \end{cases}$$

Here t is the system running time. It means that the system has a switching at 30th second. The noise in the measurement of the output follows the distribution $\mathcal{N}(0, 0.001)$.

The test condition in the first case is set as $T_s = 0.1$ second, the sample number for the estimation $N = 40$, forgetting factor $\rho = 0.95$ and the pre-knowledge of the sampled time delay is assumed as $l_{\max}^{u(k),k} = 30$, $l_{\min}^{u(k),k} = 0$. According to the proposed method, it need to wait more than 7 seconds (the data collection period is $(40 + 30) \times 0.1 = 7$ seconds) to start the identification procedure in the beginning. In the test, the identification begins at 100th sampling time. The system is simulated in the simulink with the step input signal. Fig. 1 shows the input signal and measured output signal.

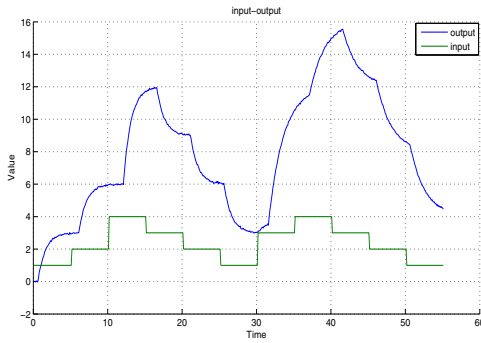


Fig. 1. The input and output data for the first test

Fig. 2 and Fig. 3 display the results of the system identification. And in order to investigate the relation between the input and time delay, the rate between the time delay and input is calculated at each estimation step. From the simulation results, the following discussion could be made:

- **Time delay estimation:** The result of the estimation for time delay is showed in Fig. 2. Before the 10th second, it is the data collection period. From 10th second, the estimation procedure begins. It need some time to attain the steady time delay estimation. This time is about 2 seconds, which is less than the time of one moving window length ($40 \times 0.1 = 4$ seconds). When the time delay changed in the system, the estimation need a

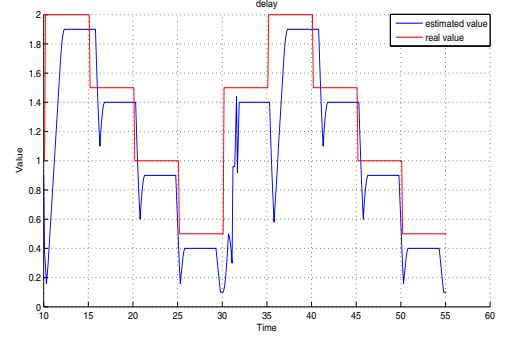


Fig. 2. The time delay estimation for the first test

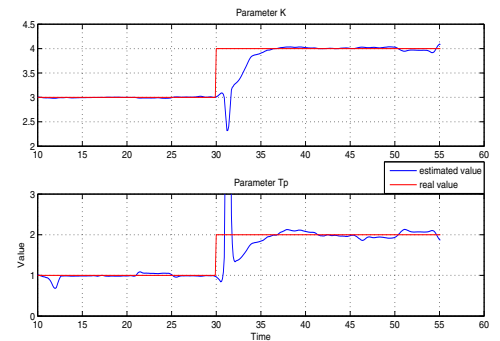


Fig. 3. The parameters identification for the first test

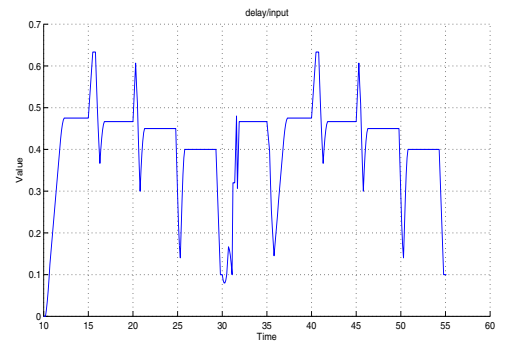


Fig. 4. The estimated delay to input relationship for the first test

little time (less than 1 second) to react to this change. Then the steady estimation was disturbed and a peak appeared. After a short time, the estimation value will be stabilized to a new value that is quite close to the real time delay again. In each time period when the time delay changes, the same phenomenon will emerge to the estimated value. But at 30th second, the system switched to a completely different system, in which not only time delay changed but also the other parameters changed. Unlike the other time when the time delay changed, before the changing time, the estimated value has been already different with the former steps. And in a period of about 2 seconds, it arises more than 2 peaks before it is back to the steady estimated value. Two different factors, both time delay change and system switching, work together to lead to this estimated value fluctuating more than before. It is observed that except for the time period when the system has a change, only small steady estimation errors (about 0.1 second) to the real time delay can be observed. This small estimation error is due to the fact that this identification solution is determined by T_s (see (5)).

- Parameters identification:** The result of identification to the other two parameters expect time delay could be seen in Fig. 3. Regarding the estimation of system gain K^t , unlike the time delay estimation, it does not need time to attain the steady estimated value. From the beginning at 10th second, it showed a quite good performance to this parameter identification. The estimated value is nearly the same to the real value. When K^t changed at 30th second, the estimated value would be away from the original steady estimated value. The estimated value had a large peak before it returns to another new steady value. But the time it needed (about 5 seconds) is much more than the time used in the time delay estimation. At all sample time where the estimation value is steady, the error for the estimation of K^t is below 1%. For another parameter T_p^t , in the beginning, it has a small deviation and quickly returns back to the steady estimation value which is quite close to the real value 1. At the time the system changed, it can be observed a rather large peak appeared (in order to show most estimation result in detail, the value is omitted in the figure). In more than 5 seconds, it recovered to the steady value. But it seems not so steadier than the estimation for the parameter K^t . It is believed that the unavoidable error of the time delay estimation affects much more on this time constant than the system gain.

- Time delay and input:** In order to show the relation between time delay and input signal, the rate between estimated time delay and measured input signal at each sampling time is calculated, which can be seen in Fig. 4. It can be observed that except for the time period the time delay changed, the rate is in the range of 0.4-0.5 at each sample points showed steady estimation. The result approximately shows that how the time delay depended on the input.

In order to show the moving widow length can affect the estimation result, a number of other tests are conducted. In each test, only N is changed with the first test to make the estimation. The results could be seen from Fig. 5 to Fig. 8.

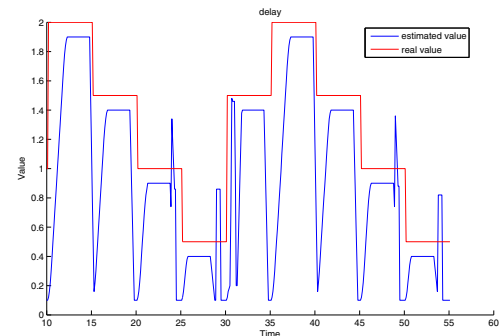


Fig. 5. The time delay estimation based on 30 samples

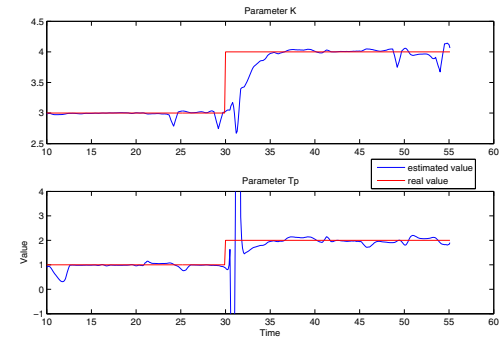


Fig. 6. The parameters identification based on 30 samples

From these tests, it can be observed:

- Sample number:** From Fig. 5 to Fig. 8, different sample numbers are adopted. According to the results, estimation using 50 samples could provide the smoothest results at the cost of the delay to detect the parameters change in the system. But the result using

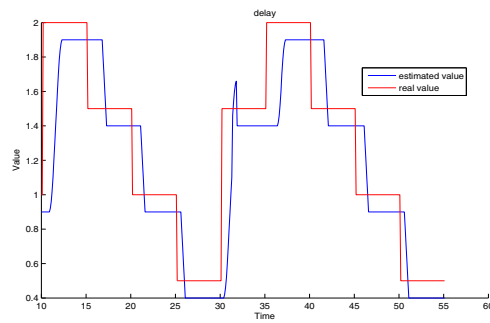


Fig. 7. The time delay estimation based on 50 samples

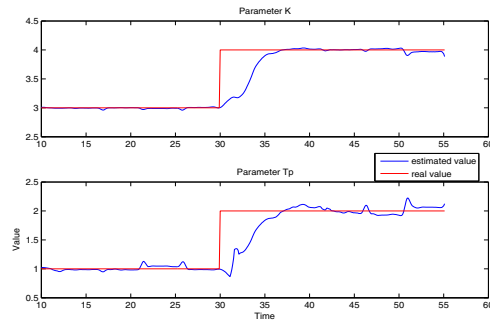


Fig. 8. The parameters identification based on 50 samples

30 samples not only is less smooth, but also had an obvious decreasing in the accuracy.

Conclusion from the simulation test:

From a number of the simulation, for a TV-FOPDT modeled system, in which the time delay depended on the input and other parameters of the system have a sudden change at some time, the proposed method, using SMINLP programming based on the BB and LMS method, can provide an accurate and prompt estimation for the time delay and parameters. The choosing of length of moving widow could affect the performance of the estimation. Less data leads that the estimation can not capture the property of the system and results in an unsatisfied performance. In general, the more sample number is used for estimation in each step, the much smoother estimation result could be obtained at the cost of the estimation delay. But too many sample points can decrease the estimation to reflect time varying property of the parameters. It need to look for a balanced sample number in the estimation.

V. CONCLUSION AND FUTURE WORK

This paper considered a system identification problem for a kind of TV-FOPDT model. This model is a extension of

the standard FOPDT model, in which the parameters are both time dependent, especially the time delay depends on the input signal as well. A Stochastic Mixed Integer Non-Linear Programming problem is formulated to make the system identification of the model. Then based on BB and LMS method, an iterative scheme is applied to solve the problem and take the optimal solution of the problem as the estimation of the parameters for the system. The simulation tests showed that this method could provide an accurate and prompt estimation for the parameters of the system model although it has an avoided estimation error based on sample interval. The performance can be affected by sample number.

In the future, to determine the optimal sample number to apply this model and method to the real system are parts of our work.

REFERENCES

- [1] K. Brazauskas and D. Levisauskas, Adaptive transfer function-based control of nonlinear process. Case study: Control of temperature in industrial methane tank, *Int. J. Adapt. Control Signal Process.*, Vol.21, 2007, pp. 871-884.
- [2] T. R. Fortescue, L. S. Kershenbaum and B. E. Ydstie, Implementation of self-tuning regulators with variable forgetting factors, *Automatica*, Vol. 17, No. 6, 1981, pp. 831-835.
- [3] E. W. Gralda and J. W. MacArthurb, A moving-boundary formulation for modeling time-dependent two-phase flows, *International Journal of Heat and Fluid Flow*, Vol.13, No.3, 1992, pp. 266-272.
- [4] I. E. Grossman and N. V. Sahinidis (eds), Special issue on mixed integer programming and its application to engineering, Part I, *Optim. Eng.*, Vol. 3, No. 4, Kluwer Academic Publishers, 2002.
- [5] K. E. Kwok, S. L. Shah, B. A. Finegan and G. K. Kwong, An observational trial of a computerized drug delivery system on two patients, *IEEE Trans. Cont. Sys. Tech.*, Vol.5, No.4, 1997, pp. 385-393.
- [6] L. F. S. Larsen, R. Izdi-Zamanabadi and R. Wisniewski, Supermarket Refrigeration System - Benchmark for Hybrid System Control, *Proc. of ECC07*, Kos, Greece, 2007.
- [7] J. Lee, W. Cho and T. Edgar, Control system design based on a nonlinear first-order plus time delay model, *J. Proc. Cont.*, Vol.7, No.1, 1997, pp. 65-73.
- [8] H. Li, S. Jeong, J. Yoon and S. You, An empirical model for independent control of variable speed refrigeration system, *Applied Thermal Engineering*, Vol.28, 2008, pp. 1918-1924.
- [9] L. Ljung, *System Identification - Theory for the User*, 2nd edition, PTR Prentice Hall, Upper Saddle River, 1999.
- [10] H. Ouarti and T. F. Edgar, The Use of Approximate Models and Exact Linearization for Control of Nonlinear processes, *Proc. Amer. Cont. Conf.*, 1993, pp. 2268.
- [11] H. Rasmussen and L. F. S. Larsen, Nonlinear superheat and capacity control of a refrigeration plant, *17th Mediterranean Conference on Control and Automation*, Thessaloniki, Greece, Jun. 24-26, 2009, pp.1072-1077.
- [12] Z. Yang, Z. Sun and C. Anderson, Nonlinear FOPDT Model Identification for the Superheat Dynamic in a Refrigeration System, submitted paper, *IECON 2011*.
- [13] Z. Yang, C. Andersen and R. Izadi-Zamanabadi, On-line Auto-Tuning of PI control of the superheat for a supermarket refrigeration system, *IEEE Multi-Conference on Systems and Control*, Denver, USA, Sept. 28-30, 2011.

On Defining and Following Robot Paths in an RFID Augmented Environment

Ioan Susnea, Grigore Vasiliu

Abstract— Unlike the vast majority of the applications of the RFID technology for robot navigation, which refer to problems related to robot localization and mapping, this paper explores the possibilities to use RFID tags deployed in the environment for defining and following robot paths. The experiments described here prove that an RFID augmented environment can contribute to a drastic simplification of the design of some service robots.

Index Terms— RFID, real time robot control, path following.

I. INTRODUCTION

The antropomorphic paradigm we use when designing robots is sometimes counterproductive. We tend to expect too much from the robots. They are not necessarily supposed to sense the world the way we do, and their environment is not always unknown, or even hostile. In fact, most service robots operate in an environment that is not only known/predictable, but also can be manipulated to include sensors, actuators, computing and communication equipment.

One convenient way to manipulate the environment is to deploy a number of RFID devices, capable to store a variety of digital data, aimed to provide autonomous robots with valuable navigation information. As a result, the on-board equipment can be significantly reduced, along with the power requirements, and the overall cost of the robots.

An RFID (Radio Frequency Identification) system ([1]) typically consists in a reader, equipped with an antenna, and at least one “RFID tag”, which is a small ROM/EEPROM memory, and additional circuits for providing power and access to the digital data stored in the tag (see fig. 1).

One major advantage of the RFID technology is that the tags (in most situations) are powered by the electromagnetic field generated by the reader’s antenna.

It is also important to note that the price of the commonly used tags is only a few cents per unit.

The actual data transfer between the tag and the reader is

based on “load modulation”, obtained by connecting a variable load to the resonant circuit of the receiver’s antenna.

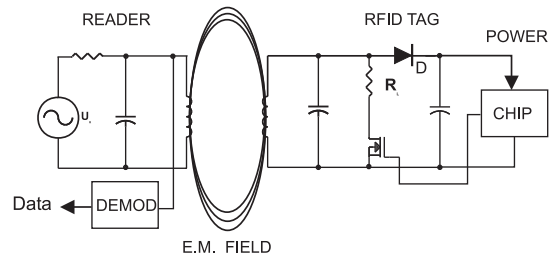


Fig. 1 A typical, passive RFID system

This results in an amplitude modulation (A.S.K. – Amplitude Shift Key) of the RF carrier, which can be sensed and decoded by the reader.

Depending on the maximum range where data transfer between reader and tags is possible, RFID devices are called:

- proximity tags (when the operating range is between 0 and 100mm)
- vicinity tags (when the operating range is between 100 mm and 1000 mm)
- long range tags (when the operating range is between 1m and 15 m. These tags are usually active – they need their own power supply, and are much more expensive).

Typical data storage capacity of the low cost tags range from a few bytes of ROM, to a few kilobytes of EEPROM.

The A.S.K modulation data transfer may be affected by collisions, if more than one tag is in the recognition area of the reader. (see figure 2). In this case, no tag will be read.

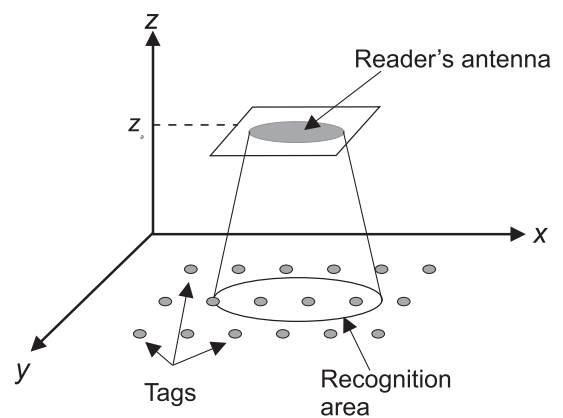


Fig. 2 A placement of tags leading to data collision

Manuscript received Mat 30, 2011.

Ioan Susnea is with the Computer and Applied Informatics Department, University Dunarea de Jos of Galati, Romania. Corresponding author. (e-mail: isusnea@yahoo.com)

Grigore Vasiliu is with the Automation and Industrial Informatics Department, University Dunarea de Jos of Galati, Romania (e-mail: vasiliugrigore3@yahoo.com)

The solution is to adjust the tag density and the distance between the reader's antenna and the tags, so that only one tag is in the recognition area at any given moment (see figure 3). Another solution is to use a modern reader, having collision avoidance capabilities, but even in this case, tags are read one by one, in sequence.

This aspect is essential for understanding the applicability and the limitations of using RFID tags in robot navigation. For example, if a tag distribution as depicted in figure 3 is used for robot localization, with the reader carried by the mobile robot, and the tags embedded in the floor, wherein the tags store the absolute coordinates (x,y) of their location, the position estimation error will be comparable to the radius of the recognition area, and there is no direct means to determine the robot orientation, based on the information stored by the tags.

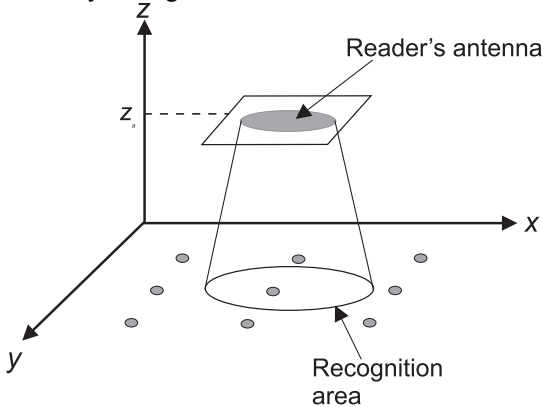


Fig. 3. Correct placement of tags for collision-free data transfer

To conclude, we should keep in mind that the following major constraints must be considered in any application of RFID for robot control:

- The memory capacity of most passive tags is not enough to store a whole map of the environment.
- Only one tag can be read at a time.

With these constraints, the possibility to define paths, as sequences of points, defined by information stored in different tags, seems, at a first glance, very limited.

This explains why the majority of the solutions proposed in the literature for using the RFID technology in robot navigation is limited to robot localization and mapping.

This paper presents two possible solutions to the problem of defining and following paths for real time control of mobile robots, by means of passive RFID tags deployed in the environment.

Besides this introduction this article is structured as follows:

Section 2 is a brief overview of the related work.

Section 3 presents the general issues related to defining and following paths by means of RFID tags, and suggests two possible solutions.

Section 4 presents the experimental results obtained by applying a new method for defining paths in a RFID augmented environment.

Finally, section 5 is reserved for discussion and conclusions.

II. RFID FOR ROBOT LOCALIZATION

Although the principles of the RFID are known since 1973 ([2]), recent advances in this technology led to a significant drop of the tag prices, along with an increase of their performances.

After 2004, a great number of research articles address the problem of using RFID technology for robot control. Most of them ([3], [4],...[10]) propose various solutions for robot self-localization and mapping.

Haehnel et al. in ([3]) use a reduced set of wall mounted vicinity tags, and a probabilistic method to determine the robot's position.

Kulyukin et al. ([4]) also propose wall mounted vicinity tags as landmarks in a system aimed to guide visually impaired persons. Later, they propose an improved solution ([5]), based on proximity tags embedded in a "smart floor".

Park and Hashimoto ([6]) also use a network of floor mounted proximity tags, and propose a method to determine the robot orientation by measuring the response time of the tags.

In ([7]) and ([8]), the authors determine the position of the robot with respect to the RFID tags, based on considerations about the propagation of the electromagnetic waves.

The solutions proposed in ([9]) and ([10]) rely on fusing the RFID and odometry information to obtain more accurate position estimations.

III. DEFINING ROBOT PATHS WITH RFID

A. The Digital Pheromones Approach

An important step towards using RFID in path following was made by Mamei and Zambonelli ([11]) by defining a model for "digital pheromones", stored in read/write RFID tags "densely deployed in the environment".

According to this model, digital pheromones are data structures, stored in RFID tags, consisting in one integer indicating the pheromone source intensity, and a time stamp.

The pheromone intensity is implemented as a counter C_n , which is incremented every time an agent accesses the tag, or decremented, when the time elapsed from the last access was greater than an evaporation constant.

$$C_{n+1} = C_n + Qf(T_{n+1} - T_n) \quad (1)$$

where

$$f(t) = \begin{cases} +1 & \text{for } t < \Delta t \\ -\text{int}(\frac{T_{n+1} - T_n}{\Delta t}) & \text{for } t > \Delta t \end{cases} \quad (2)$$

In (1) and (2), Q is a positive constant, representing the quantum of reinforcement/evaporation of the pheromone, $\text{int}(x)$ is the integer part of x , and Δt is the evaporation constant. T_{n+1} is the current time according to the real time clock of the agent, and T_n is the time stamp of the previous record, read from the tag.

This model elegantly solves the problem of pheromone evaporation, but it still cannot describe the spatial diffusion,

because only one tag can be read at a time.

To overcome this issue, we proposed in ([12]) a solution that uses two lateral RFID readers for sensing and modifying the pheromone intensities. Since the two readers are located some distance apart from each other, this enables the agent to actually sense spatial gradients of the digital pheromone distribution, and we used the differential data provided by the two readers as positioning error, relative to the pheromone trail. A fuzzy logic controller was used for the actual task of path following, but any other control algorithm (PID, sliding mode, etc.) is equally applicable.

B. The Virtual Relief Approach

Consider a smart floor, as shown in figure 3. This structures the space into a 2D grid, where each cell has a RFID tag in the center, and a size (a) comparable with the diameter of the recognition area of the RFID readers (see fig. 4).

For the usual proximity tags, the recognition area can be considered circular, with a diameter of 15-20cm. Commonly used research robots, like Pioneer3-DX and PeopleBot (www.mobilerobots.com) have a bias (b) of 38-45cm. If the robots are equipped with two readers, placed laterally on the left and right sides of the robot, these will read tags located 3-4 cells apart from each other.

Now, let's assume that each tag contains a one byte unsigned integer, which is interpreted as the "virtual altitude" of the corresponding cell.

Fig. 5 shows an example of representing the map of the RFID augmented environment, including virtual altitude information. In this representation, darker shades of gray indicate cells with higher altitude, while zero altitude cells are depicted in white.

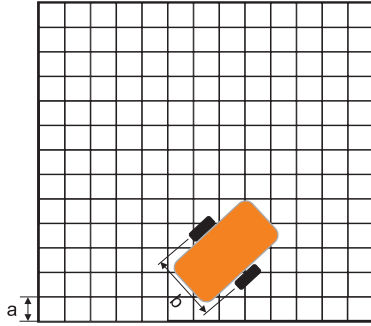


Fig. 4. An example of grid map associated with the RFID smart floor

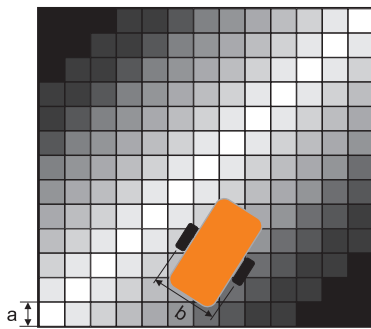


Fig. 5. Defining a path by means of virtual altitude information stored

in RFID tags

With this approach, the path is extended to the size of the smart floor. By sensing the altitude of the cells with two readers, the robot can be easily controlled to follow the path defined as adjacent cells with the lowest altitude.

The positioning error in this case is simply:

$$e = A_L - A_R \quad (3)$$

Where A_L and A_R are the altitudes of the cells corresponding to the left and right drive wheel of the robot. If (x_0, y_0, θ_0) is the current position of the robot, and b is the distance between the planes of the drive wheels, it is assumed that the coordinates of the two readers that read the altitude are:

$$x_R = x_0 - \frac{b}{2} \sin \theta_0 \quad (4)$$

$$y_R = y_0 - \frac{b}{2} \cos \theta_0 \quad (5)$$

$$x_L = x_0 + \frac{b}{2} \sin \theta_0 \quad (6)$$

$$y_L = y_0 + \frac{b}{2} \cos \theta_0 \quad (7)$$

The whole process is similar to the flow of a river. That's why we would call this method of defining paths "the riverbed model".

IV. EXPERIMENTS WITH RFID DEFINED PATHS

In the first stage of the experiment, both the smart floor and the robot were simulated by software applications running on two computers (fig. 6).

We have used MobileSim, provided by MobileRobots, (www.mobilerobots.com) as the robot simulator. This offers a pretty good simulation of the kinematic behaviour of the robots Pioneer3-DX and PeopleBot, used in the second stage of the experiment.

According to the experimental protocol, the robot/simulated robot periodically sends to the control application data packets containing the current position of the robot (x_0, y_0, θ_0) , as reported by the robot's odometric system. The software computes the positions of the readers with (4)..(7) and extracts the altitudes A_L, A_R , then computes the positioning error with (3).

The actual control application, running on a second computer, was a simple fuzzy logic controller (FLC), like that described in ([13]).

The FLC receives as input the difference of altitude sensed by the two RFID readers/extracted from the map simulating the smart floor, computed with (3), and generates references for the speeds of the drive wheels (v_R, v_L).

The smart floor, was represented as a 100x100 matrix, stored in a separate text file.

In the second stage of the experiment, we have used off the shelf Pioneer3-DX and PeopleBot robots, with no modifications to the control application (fig. 7). The smart

floor was still simulated.

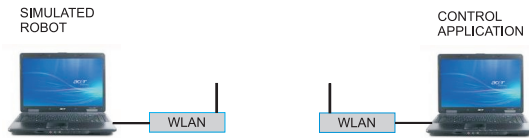


Fig. 6. Simulation setup for stage 1 of the experiments

Figures 8 and 9 present the path of the simulated robot, recorded with MobileSim, in two typical control situations.

No significant differences were noticed when working with real robots and simulated smart floor.

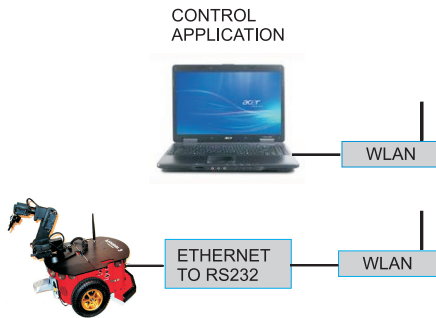


Fig. 7. Simulation setup for stage 2 of the experiments

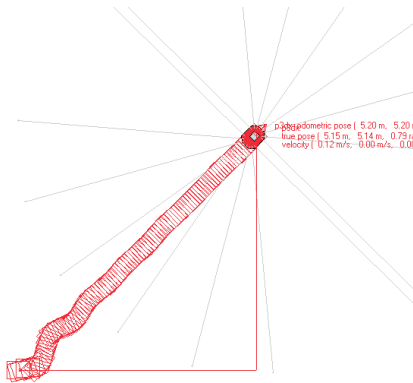


Fig. 8. Recorded simulation result for a 45 degrees turn of the target path

V. DISCUSSIONS AND CONCLUSION

The method described has some important advantages:

- Since the storage memory requirements for the tags are very low, it is possible to define hundreds of distinct paths, even when using the cheapest tags.
- The resulting path following algorithm is also very simple, and can be implemented using low cost, low power microcontrollers. This will reduce implementation time and the overall cost of the system.

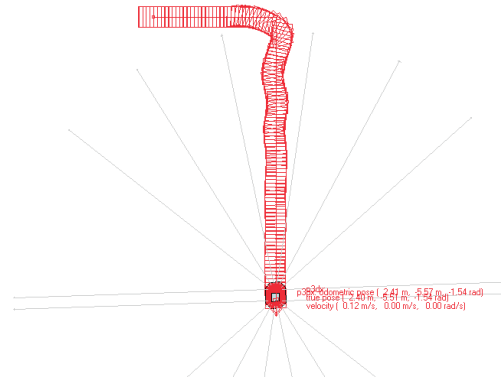


Fig. 9. Recorded simulation result for a 90 degrees turn of the target path

- In some applications, this method will eliminate the need of a localization system for the robots, thus leading to further cost reduction.
- It is compatible with some simple, reactive, obstacle avoidance algorithms, like the one described in ([14]).
- It is also compatible with several localization systems.
- Preparation of the smart floor is relatively simple and can be automated. Some “smart carpets” are already commercially available, and it is possible to embed the RFID tags in floor tiles.
- The method described can lead to a drastic cost reduction of some service robots (e.g. like intelligent wheelchairs, or intelligent walkers to carry or guide patients in hospitals, nursing homes, etc.) and allows the design of intelligent shopping carts, able to guide persons with various disabilities in supermarkets.

The only drawback of the method is that the accuracy of defining paths is limited by the size of the recognition area of the RFID readers (15-20cm). Some oscillations are unavoidable, but they can be reduced through a proper design of the controller.

The described system is also very robust to data loss. If, for some reason, the data stored in some RFID tags is altered or lost, given the inherent distributed nature of the system, the effect on the overall operation of the system would be barely perceptible.

This work was conducted in the context of exploring the possibilities to reduce the cost of some service robots by using a distributed control system of low cost embedded microcontrollers in a manipulated smart environment.

The experiments described here seem to demonstrate that such simple solutions are not only possible, but also highly desirable.

We truly believe that using the RFID technology in robot navigation can be a valuable tool for creating affordable service robots.

REFERENCES

- [1] Finkenzeller K, *RFID Handbook, Second Edition*, John Wiley and Sons, 2003
- [2] Cardullo et al. *Transponder apparatus and system*, U.S. Patent 3,713,148, 1973
- [3] Haehnel, D., Philipose M. et. al., *Mapping and Localization with RFID Technology*, in Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on, 2004, Vol. 1, pp: 1015- 1020
- [4] Kulyukin V., Gharpure C., Nicholson J., Pavithran S., *RFID in Robot-Assisted Indoor Navigation for the Visually Impaired*, Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on Volume 2, Issue , 28 Sept.-2 Oct. 2004 pp: 1979 -1984 vol.2
- [5] Kulyukin V., Kutianawala A., Jiang M., *Surface-embedded Passive RF Exteroception: Kepler, Greed, and Buffon's Needle*, in Ubiquitous Intelligence and Computing, Springer, 2007, pp: 33-42
- [6] Park S., Hashimoto S., *Indoor localization for autonomous mobile robot based on passive RFID*, Proceedings of the 2008 IEEE International Conference on Robotics and Biomimetics Bangkok, Thailand, February 21 - 26, 2009
- [7] Koutsou A. D., Seco F., Jimenez A. R., Roa J. O., Ealo J. L., Prieto C., Guevara J., *Preliminary Localization Results With An RFID Based Indoor Guiding System*, in IEEE International Symposium on Intelligent Signal Processing Alcala de Henares, Spain, October, 2007.
- [8] Zhang Y. , Amin M.G., Kaushik S., *Localization and Tracking of Passive RFID Tags Based on Direction Estimation*, International Journal of Antennas and Propagation , 2007
- [9] Byoung-Suk Choi, Joon-Woo Lee and Ju-Jang Lee *Localization and Map-building of Mobile Robot Based on RFID Sensor Fusion System*, in IEEE International Conference on Industrial Informatics, Daejeon Korea, INDIN 2008
- [10] Lim Hyung Soo, Choi Byoung Suk, Lee Jang Myung, *An Efficient Localization Algorithm for Mobile Robots based on RFID System*, SICE-ICASE International Joint Conference 2006 Bexco, Busan, Korea
- [11] Mamei, M. Zambonelli, F. - *Physical deployment of digital pheromones through RFID technology*, Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE
- [12] Susnea I. , Vasiliu G, Filipescu A, *RFID Digital Pheromones for Generating Stigmergic Behaviour to Autonomous Mobile Robots*, 4th WSEAS Int. Conf. on Dynamic Systems and Control (Control '08), CORFU ISLAND, GREECE, October 26-28, 2008, pp.20-24
- [13] Susnea I, Vasiliu G, Filipescu A, *Real-Time, Embedded Fuzzy Control of the Pioneer3-DX Robot for Path Following*, Proceedings of 12th WSEAS International Conference on SYSTEMS, Heraklion, Greece, July 22-24, 2008, pp.334-338
- [14] Susnea, I.; Filipescu, A., Vasiliu, G., Coman, G., Radaschin, A., *The bubble rebound obstacle avoidance algorithm for mobile robots*, ICCA 2010, The 8'th International Conference on Control and Automation, Xiamen, China, 9-11 June, 2010, pp: 540-545

Use of “Petri Nets System” Concept in Modeling Dynamics with Increased Complexity

Oleg Tărăbuță

Abstract—Relying on the Petri net formalism, the paper develops new modeling strategies for systems with increased complexity that coordinate several types of workflows and different classes of resources, within a context of specific requirements (such as priorities, deadlines etc). At the lower level of model construction, the individual behavior of the constituent workflows is captured by classical Petri nets. For the higher level of coordination, the modeling strategy is based on the concept of “system with Petri nets”, or, equivalently “Petri nets system” which is able to accommodate both the Petri net descriptions (commented above) and specialized structures ensuring adequate interfaces and optimal services (relative to a set of operation criteria). The “Petri nets system” models proposed in this paper are rigorously defined as topology and token dynamics. Finally, we present an example that illustrates the construction of a Petri nets system with self-adaptive management and effective resource allocation.

I. INTRODUCTION AND MOTIVATION

THE problem of efficient allocation of resources was and will continue to be a problem. Optimization and finding effective methods can lead to solve financial issues and performance index. These solutions can be applied in all areas, starting with resource management, real-time scheduling and planning activities areas, and ending in the industry area, depending on the problem details and solutions found.

Implementation of *Petri nets* [1] in computer science and automation has led to the creation of simulation programs for activities that use resources [2]. Following these tests could decide whether or not deadlock problems appear in execution of our activities.

Implementing the concept of *timed Petri networks* has brought new advantages in solving the problem of resource allocation and verification of allocation effectiveness [3, 4]. A disadvantage of timed Petri nets is the execution of a single series of activities at the same time as the network’s construction.

More advanced are *nested Petri networks* [5, 6, 7, 8, 9]. Their research targeted networks that include a set of Petri nets on two interconnected levels and implementing rules and dependencies of transitions. First level networks are activities with a part in producing a particular type of resource in the process. The second level conveys the overall

process state transitions and the network nodes. Each node can be connected to multiple networks from the first level.

Based on nested Petri networks *dynamic interorganizational workflows* (DIWF-nets) have been created [10, 11]. DIWF-nets check network properties to respect the availability of nodes, connections and completion success. But they do not relate to timed networks and not solve the problem completely.

One of the solutions used in industry are *holonic manufacturing systems* based on Petri net [12, 13, 14, 15]. These systems are used for reconstruction of production lines in the most efficient way [16]. The goal is to select a solution that meets the deadline time for the production line needed. The weaknesses of this solution are inefficient and exclusively use of resources. Giebels et al. as well as Wullink et al. have proposed a system architecture for a flexible planning of production and control system, named EtoPlan, for concurrent manufacturing planning and control of activities and groups of resources in a manufacture- to-order environment based on multiple and holarchies[17, 18].

This article will describe a method to solve a complex problem of allocating resources to parallel performed processes. Additionally, each action from each process will have deadline, and each process will have a priority attribute. This complex system will be simulated using Petri nets. Initially, we will make an overview of Petri nets in chapter 2. In chapter 3 we will present a *Low Complexity and High Complexity Petri nets System*, and then will describe the theory for new type of *self-adaptive Petri nets systems* in chapter 4. A short practical example for high complexity Petri net system and self-adaptive system will be show in chapter 5.

II. BRIEF OVERVIEW OF CLASSIC PETRI NETS

A. Petri Nets

Definition 1. A **Petri net** (called *marked Petri net*) is a 5-tuple (P, T, F, W, M_0) where:

- P and T are two finite nonempty sets (of *places* and *transitions*, respectively) such that $P \cap T = \emptyset$ and $P \cup T \neq \emptyset$, where $\emptyset$ denotes the empty set;
- F is included or equal with $(P \times T) \cup (T \times P)$ - a set of arcs;
- $W : (P \times T) \cup (T \times P) \rightarrow N$ is the weight function of N verifying $W(x, y) = 0$ if $(x, y) \notin F$;
- $M_0 : P \rightarrow N$ is the initial marking;

Manuscript received June 15, 2011.

The author is with the Department of Automatic Control and Applied Informatics from the Technical University “Gh. Asachi” of Iasi, Blvd. Mangeron 27, 700050 Iasi, Romania (e-mails: oleg.tarabuta@gmail.com).

Token Dynamics in Petri Nets

Petri Nets transitions are validated when in predecessor positions of transition t are available $W(p,t)$ tokens. During firing $W(p,t)$ tokens are removed from position p and $W(t,p)$ tokens are moved to successors of transition t .

B. Timed Petri Nets

In 1974 P.M. Merlin has introduced timed Petri nets [19]. Merlin timed Petri nets extend classical Petri nets by associating a time interval $(I_1(t), I_2(t))$ to each transition t .

Definition 2. A **timed Petri net** (TPN) is a pair $\gamma = (N, I)$, where:

- $N = (P, T, F, W, M_0)$ is a marked Petri net, called *the underlying net* of γ ;
- $I : T \rightarrow \mathbf{R}^+ \times \mathbf{R}^+$ is the *time function* of γ , where for each $t \in T$, $I(t) = (I_1(t), I_2(t))$ with $I_1(t) < I_2(t)$.

$I_1(t)$ is called the *static interval of t* . $I_1(t)$ and $I_2(t)$ are called *static earliest firing time of t* ($\text{seft}(t)$) and *static latest firing time of t* ($\text{slft}(t)$), respectively, and are relative to the moment at which transition t was last enabled.

Token Dynamics in timed Petri Nets

In timed Petri Nets, transitions are validated when in t transition's predecessor positions of transition t are available $W(p,t)$ tokens and time frame is in the frame between $I_1(t)$ and $I_2(t)$. Tokens are removed from predecessor place and moved to successor places identically as in the Petri Nets.

C. Nested Petri Nets

Bellow it's presented a short definition of this networks type. A part of nested nets structure will be used to compare this category with our Petri nets system.

Definition 3. A **nested Petri net** is a tuple $\text{NPN} = (\text{Expr}, \text{Lab}, \text{SN}, (\text{EN}_1, m_1^0), (\text{EN}_2, m_2^0), \dots, (\text{EN}_k, m_k^0), \Lambda)$, with:

- Expr is a language of expressions defined before.
- $\text{Lab} = \text{Lab}_v \cup \text{Lab}_h$ is a set of labels.
- $(\text{EN}_1, m_1^0), (\text{EN}_2, m_2^0), \dots, (\text{EN}_k, m_k^0)$ are ordinary marked Petri nets, called object-nets.
- $\text{SN} = (N, U, W, M_0)$ is a *high level Petri net*, called a system-net of NPN, where $N = (P, T, F)$ is a Petri net, $U = (A, I)$ is a model for Expr, $W : F \rightarrow \text{Expr}$ is a function such that $W(x, y) = \theta = (\theta_1, \dots, \theta_n)$ where $\theta_i \in \text{Expr}$ ($1 \leq i \leq n$) and M_0 is the initial marking of the net.
- Λ is a partial function which assigns to the transitions of the system-net SN and of the object-nets EN_i ($i \in \{1, \dots, k\}$), labels from Lab.

Modeling Strategy

A nested Petri net refers to a single process only. The object-nets EN_i are an Petri net and their role are to produce some tokens or to make some actions. This solution can be applied to many processes when are used resources in exclusive mode. In case of shared resources, multiple nested Petri net must be interconnected. System complexity of these multiple nested nets increases and dependencies between transitions from different processes needs additional conditions and rules.

If Petri nets are replaced with timed Petri nets then timing fire rules for transitions must be implemented in this multiple nets system. The system, obtained using multiple nested timed Petri nets with shared resources, is too complex to be implemented and verified. When the priority attribute is added for each nested net from entire system as well as the execution time attribute for each transition, the resulted workflow verification may be impossible.

In our proposed network type, a complex Petri net system will replace the system with multiple nested nets because of nested net complexity.

III. NEW TYPES OF MODELS BASED ON PETRI NET SYSTEMS

We introduce two new types of Petri net systems: Low and High Complexity. These types will solve problem's translation to workflow and their easy representation. Only Petri Nets or nested Petri nets can not represent timed networks and networks with different priorities.

A. Low Complexity Petri Net Systems

Definition 4. A **Low Complexity Petri net system** is a tuple $\text{LC-PNS} = (\text{TPNS}, \text{RS}, \text{FM}, \text{ETS})$, where:

- TPNS is a *set of timed Petri nets* ($\text{TPN}_1, \text{TPN}_2, \dots, \text{TPN}_k$), where k is index of network ($1 \leq k \leq n$, n is number of networks).
- RS is a *tokens set* $((\text{res}_0, \text{type}_0, \text{status}_0), \dots, (\text{res}_k, \text{type}_k, \text{status}_k))$, where res_i is token name, type_i is token type and status_i is the availability status;
- FM is the *set of final marking* ($M_f(\text{TPNS})$) for all nets.
- ETS is a *extended time system* with role to assign $\text{slft}(t_k)$ and calculate $\text{seft}(t_k)$ using fire times $d(t_k)$ for all transition from each TPN_i from TPNS .

Token Dynamics in LC-PNS

Token status is defined as enumeration type $\{\text{available}, \text{reserved}\}$. When token is required by transition ($\bullet t$) and token is available, status is changed in reserved. When transition is done ($t \bullet$) all tokens used by transition t are released and their statuses are set to available (Figure 1).

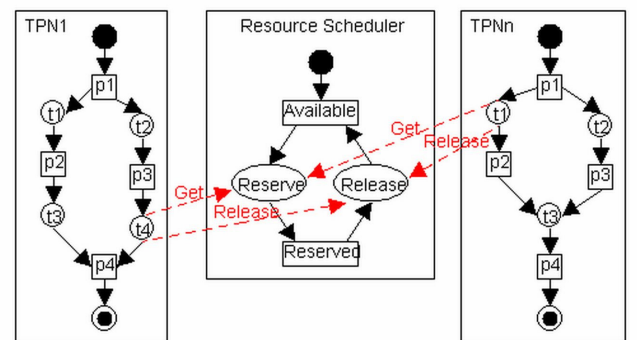


Fig. 1. Tokens status updating method in Resource Scheduler

Additional for transitions from Petri net set there will be added two new attributes: execution time and transition deadline. Execution time is a positive number greater than zero. Execution time (*delay time* - $d(t_k)$) for each transition

must be less than difference between static latest firing time and static earliest firing time.

The static earliest firing time is calculated using the marking of Petri net from initial marking M_0 to the predecessor position of the current transition t_k .

$seft(t_k) = \sum_i (d(t_i))$, where $t_i \in t^*$, $M_0[t^*]m_k$ and m_k is predecessor position of t_k .

The execution time and the static latest firing time are constant values and are read from the initial configuration.

The resource scheduler from Petri net system is updated with time option in order to be compatible with new type of systems with deadlines. Execution of the entire system will be performed using a timer. In each time unit, all Petri may require tokens from resource scheduler.

Modeling Strategy

Each TPN_k Petri net from TPNS uses resources in exclusive and shared mode. All used resources are tokens. A TPN_k will represent a process. All processes will be run in parallel mode. Some processes use common (shared) tokens, in which case it is necessary to allocate this token category in dynamic mode. Transitions in networks can be activated only if shared tokens are available. Status update of shared tokens and allocation must be executed by the resource scheduler. Allocation of all tokens will be done when a transition requires tokens. Executions of transitions in networks are made identical as in the Petri nets. A similar technique is used in workflows [20].

B. High Complexity Petri Net Systems

Definition 5. A High Complexity Petri nets system is a tuple **HC-PNS** = (LC-PNS, TTS, PS, ACR, LS), where:

- LC-PNS is a Low Complexity Petri Net System;
- TTS is a transition tokens configuration system who specify number and types of tokens used for transitions nodes.
- PS is a priority system for Petri nets from system ($Prio(TPN_i)$);
- ACR is the set of rules for system (RAOM, AcR0, AcR1, AcR2 and AcR3);
- LS is a system log with role to save data for static firing start time ($sfst(t_i)$), time exceeding events and deadlocks.

Token Dynamics in HC-PNS

Allocation algorithm for tokens can be selected from FIFO mode (first in – first out) or from optimized mode to avoid tokens deadlocks. In our model of Petri nets system the tokens for a specific transition (action) from a process (Petri net) will be allocated only if the number of required tokens is available (*resource allocation in optimized mode – RAOM*) (Figure 2).

Using this allocation mode tokens will not be blocked while the number of available tokens is less than the weight of arc from Petri net ($W(x,y)$). The weight of arcs to a transition t_i will be equal with arcs weight from transition t_i .

The resulted rule for tokens allocation is: number of

reserved tokens to firing a transition node is the same with number of released tokens. Tokens are reserved by a transition t_i during its static firing time ($d(t_i)$). After this time tokens are released and become available.

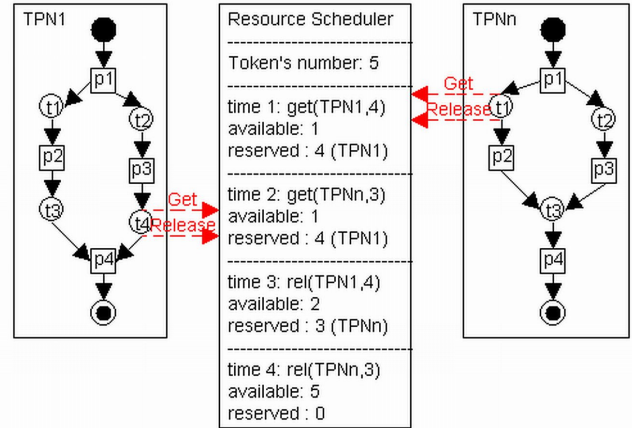


Fig. 2. Optimized resource allocation mode (Transition t_1 from TPN_n is not possible until tokens are released by transition t_4 from TPN_1).

When two or more transitions from different Petri nets in the same time can obtain the required tokens, the problem of allocation may be not decidable. FIFO and optimized algorithm can not be applied when the sum of required tokens is greater than total number of possible available tokens (Figure 3). In these cases it is needed to define a rule for tokens allocation.

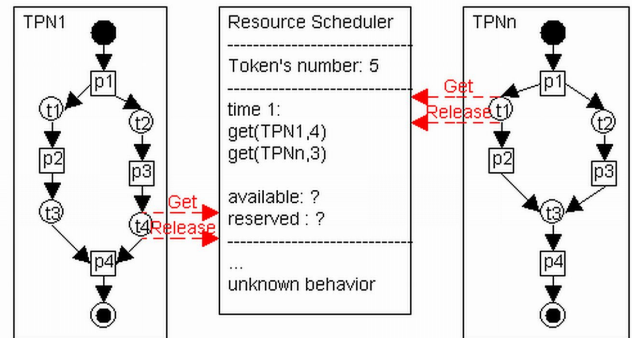


Fig. 3. Deadlock situation and not decidable case (TPN_1 and TPN_n requires tokens in the same time. Number of tokens is less than sum of required tokens).

Times for transitions in HC-PNS

During system execution, transitions activation times are triggered for deadline verification. *Static firing start time* ($sfst(t_i)$) is defined as time when all tokens for transition are available and transition is activated. The next conditions must be satisfied:

- $seft(t_i) \leq sfst(t_i) \leq slft(t_i)$ – firing start time is allowed only in the firing time period for transition t_i ;
- $sfst(t_i) + d(t_i) \leq slft(t_i)$ – end of execution shall not exceed deadline.

A *time exceeding event* happens when firing start time or end of execution exceeds deadline. Firing start time can not

occur before static earliest firing time because earliest firing time is calculated using execution times of transitions to current transition from initial marking.

Reasons for time exceeding are the unavailable tokens for some Petri net from managed system. A solution to avoid deadlock is the use of execution priority provided by the execution ratio. *Execution ratio* is defined as: $ER(t_i) = d(t_i) / (slft(t_i) - seft(t_i))$. The value of this parameter must be less than or equal with 1. When execution time can not be done within the firing period, deadlock of Petri net appears and it may be a system deadlock.

Token Dynamics in HC-PNS

For two or more transitions able to be activated in the same time the activation rules (AcR) of HC-PNS are:

- firstly transitions with the earliest deadline are executed (**AcR1**);
- for transitions with equal deadline, the greater execution ratio ($ER(t_i)$) has high priority (**AcR2**);
- for transitions with equal deadline and execution ratio the execution will be done by Petri net index (**AcR3**).

Activation rules are applied in order: firstly AcR1, then AcR2 and finally AcR3.

Using activation rules a priority system is elaborated. These priorities are set depending on transitions proprieties and modifying the execution order for any Petri net will not be possible without modifying a lot of transition proprieties.

A solution to execution priority is the *priority attribute* for each Petri net from HC-PNS (**Prio(TPN_i)**). This attribute will add a new activation rule:

- if two or more transitions send request for the same token then the transition from Petri net with high priority obtain token (**AcR0**).

Rule AcR0 will be applied before AcR1. In case of transitions from networks with equal priority first set of activation rules is applied (AcR1, AcR2 and AcR3).

Modeling Strategy

In order to create a HC-PNS, data for the Petri nets, execution times and deadlines and used tokens for all transitions, priority for Petri Nets and number and types of available tokens shall be provided. System scheduler will start execution of Petri nets, will dynamically assign tokens using system rules and will monitor execution times.

Before system simulation, the integration of execution time into the allowed firing period for transitions will be verified. If an exception appears, then the system had a global time exceeded event. Another verification is checking of initial availability of the tokens number used by transition nodes. If initial tokens number from a specific type is less than the quantity required by transition, then system had a global deadlock and some Petri nets will not reach the final position. After the system simulation, the data stored by system log will be used to create a set of TPN from initial networks set. The Petri nets with global system deadlock will

be simulated until deadlock appears. For all initial networks the static earliest firing time ($seft(t_i)$) will be replaced with static firing start time ($sfst(t_i)$).

The result of system simulation is a set of independent Timed Petri Nets. For each of them there all the required tokens will be allocated. Time firing constrains is the static interval of t ($I(t)$), where $I1(t) = sfst(t)$ and $I2(t) = slft(t) = \text{deadline}$. In this mode we obtain a standard timed Petri net simulation for multiple nets from a complex Petri nets system. Delay time for transitions is the execution time ($d(t_i)$) and delay for the places is the difference:

$d(p_i) = sfst(t_i) - sfst(t_{i-1}) - d(t_{i-1})$, where $t_{i-1} \succ p_i \succ t_i$ is a network sequence.

A complex Petri nets system is *fully executed* if all Petri nets reach the final place (M_f) from the initial place (M_0). An *initial place* M_0 is a place M_i if (not $\exists$) $M[t_k]M_i$ (arcs are only outputs for place). A *final place* M_f is a place p_i if (not $\exists$) $M_i \succ t_i$ (arcs are only inputs for place).

Petri nets from HC-PNS are not live and reversible, but are bounded and have a coverability tree. Using a HC-PNS it is possible to simulate multiple processes with different priorities and shared tokens. The obtained effective execution times can be compared to verify deadlines and overall delay for each process.

IV. SELF-ADAPTION FOR HC-PNC

We will introduce a self-adaptive system for high complexity Petri nets systems because the obtained results are not the optimal solutions for proposed problem in the same cases.

A. Optimization criteria for self adaptation

The optimization methods for efficient resources allocation for many processes with parallel execution must find the way to not exceed deadlines. Solutions for HC-PNS start with the activation rules for transitions.

A solution to avoid a deadlock for the current network and for other nets from the system while the RAOM rule is applied may be the additional token allocation. This option is efficient when number of required tokens is less than the global tokens number. It may be applied in other cases only if other solutions are not available.

An activation rule based on processes priority may generate time exceeded events for a process with less priority. A solution may be increasing the priority for the process with many problems or decreasing the priority of the process that generates time exceeded events for others. The decision to increase priority must be done only for processes with ratio of time exceeds more than a static constant ($\text{time exceed transitions number} / \text{total transitions number} > \text{increase_constant}$). The decision to decrease priority must be applied for a process who generate time exceed ratio more than a constant value ($\text{blocked transitions number from entire system} / \text{total system transitions number} > \text{decrease_constant}$). The increase and the decrease constants

must be set before possible optimizations and must also be less than or equal to increase/decrease processes indexes calculated in the execution time.

The activation rule for the earliest deadline can block the next executions for transitions from the process. If sequent to the exceeding time or deadlocks analysis the reason is a concurrent transition with the earliest deadline, then the deadline will be modified under the condition that the change will solve all the affected transitions problems and will not affect execution of the problem generating transition (in relation to old deadline).

In case of more transitions with equal priorities and deadlines, the activation rule for greater execution ratio can generates many time exceed events. A way to solve this problem can be the deadline change for one transition. The deadline increase will change the activation priority to a lower value. The solution is accepted only when this change will not affect the execution of the next transitions. Another solution can be to decrease deadline. In this case the activation priority will be changed to higher value and it means than other transition will have an exceeded time. If a decrease deadline is applied the increase is needed for other transitions.

The last rule sets a higher priority to a lower index network. The activation sequence can be modified if network indexes are modified. This change must not affect more transitions than those that it solves.

B. Self-Adaptive Petri Nets Systems

We propose to find a lot of solution for efficient token allocation using the HC-PNS and their optimization criteria. The processes are represented as Petri Nets.

Definition 6. A *Self-Adaptive Petri nets system* is a tuple **SA-PNS** = (HC-PNS, SAS, TPNS), where:

- HC-PNS is the *initial high complexity Petri nets system*;
- SAS is the *self-adaptive system applied to HC-PNS*;
- TPNS is the *resulted set of Timed Petri nets* executed in parallel mode;

Self-adaptive system is composed from simulation and adaptive system based on optimization criteria.

The first step is the simulation of the initial complex Petri nets system. During this simulation start and delay transitions times are monitored. Additionally, the process time exceeding ratio and process generated time exceeding ratio are calculated. Collected data are used to apply the optimization criteria depending on the detected problem. After the optimization methods were applied, new obtained HC-PNS is simulated. The system configuration is saved if a solution is found is better than the previous. Usage of self-adaptive algorithm generates many possible solutions. A solution can be the combination of two or more optimization methods. One method provides usually a solution for a local deadlock in the system and it is not guaranteed than the found solution solves problems for HC-PNS.

Different methods applied in different steps have the

possibility to find a solution for all the system nets. In some cases found better solution may exceeds deadlines or need additional tokens. A self-adaptive system may not provide an ideal solution. Priority change generates the execution delays for processes with high priority. This event may affect workflow if methods are combined. It is needed to decide the better solution after running SA-PNS. We must choice a variant acceptable for us.

V. PRACTICAL EXAMPLE

HC-PNS is a High Complexity Petri Nets System with 3 Petri Nets (TPNS = (TPN₁, TPN₂, TPN₃)). Each Petri net represents a process.

Processes have activities and these are represented by transitions. Petri nets have are described in Table1 (TPN₁ with M₀ as P₀ and M_f as P₃, TPN₂ with M₀ as P₀ and M_f as P₃, TPN₃ with M₀ as P₀ and M_f as P₂). Priorities for Petri Nets of complex system are PS = ((TPN₁, 1), (TPN₂, 1), (TPN₃, 2)).

F	TPN ₁			TPN ₂			TPN ₃	
	t ₀	t ₁	t ₂	t ₀	t ₁	t ₂	t ₀	t ₁
Pred(t)	P ₀	P ₁	P ₂	P ₀	P ₁	P ₂	P ₀	P ₁
Succ(t)	P ₁	P ₂	P ₃	P ₁	P ₂	P ₃	P ₁	P ₂

Tab. 1. Petri Nets arcs from HC-PNS.

Available tokens of system are RS = ({1, type1}, {4, type2}, {3, type3}, {1, type4}, {1, type5}, {1, type6}). Transitions from each Petri net are fired with specific tokens.

Transitions tokens configurations system is:

TTS	TPN ₁			TPN ₂			TPN ₃	
	t ₀	t ₁	t ₂	t ₀	t ₁	t ₂	t ₀	t ₁
Type1	0	1	1	0	0	1	0	1
Type2	3	3	0	1	2	0	2	2
Type3	3	0	0	0	0	0	0	0
Type4	0	0	1	0	0	1	0	0
Type5	0	0	0	1	0	0	0	0
Type6	0	0	0	0	1	0	0	0

Tab. 2. Tokens types used for transition from HC-PNS.

Tokens from types 3, 5 and 6 are specific used resources and types 1, 2 and 4 are shared resources.

Execution times (d(t)), deadlines used as static latest firing time slft(t) and calculated static earliest firing time (seft(t)) are presented in Table 3. Execution start times of Petri nets are (0,0,3).

At time unit 0: begin simulation of TPN₁ and TPN₂. Both t₀ transitions are activated because numbers of required tokens are available (4 of type2, 3 of type3, 1 of type5).

At time unit 3: t₀ from TPN₂ release used tokens and their statuses are changed to *available*. Start simulation of TPN₃. Transitions t₀ from TPN₃ and t₁ from TPN₂ requires 4 shared tokens of type2, but only 1 token is available and transitions wait for available resources.

At time unit 5: t₀ from TPN₁ release used tokens. 2 tokens of type2 are reserved to t₀ from TPN₃, because Petri net has bigger priority (*AcR0*). Transition t₁ from TPN₁ require more tokens of type2 that are available (3 required and only 2 available) and will wait (*RAOM*). Transition t₁ from TPN₂ is activated.

At time unit 7: t₁ from TPN₂ release tokens, and t₂ is

activated (all required tokens are available). Transition t_1 from TPN_1 still wait.

At time unit 9: simulation of TPN_2 is finished. Transitions t_0 from TPN_3 and t_2 from TPN_2 release tokens. Transition with high priority t_1 from TPN_3 is activated (2 tokens of type2 and 1 of type1 are reserved). Transition t_1 from TPN_1 waits because not enough tokens are available.

At time unit 11: simulation of TPN_3 is finished. Transition t_1 from TPN_1 requires one shared token of type1. Transition t_1 is activated.

At time unit 18: Transition t_2 from TPN_1 is activated.

At time unit 21: simulation of TPN_1 is finished.

Obtained results are presented in Table 3 as static firing start time (sfst(t)) and times for transitions finish. Three transitions exceed deadline, and deadline of TPN_1 is exceeded.

	TPN ₁			TPN ₂			TPN ₃	
	t_0	t_1	t_2	t_0	t_1	t_2	t_0	t_1
Sfct(t)	0	5	12	0	3	5	3	7
Slft(t)	7	15	20	8	9	14	8	11
d(t)	5	7	3	3	2	3	4	2
Sfst(t)	0	11	18	0	5	7	5	9
End at	5	18	21	3	7	9	9	11

Tab. 3. Simulation results of HC-PNS.

If the current complex Petri nets system is included in a self-adaptive system, then there can be found more solutions to avoid deadlines. One solution is to add a token of type2 and one of type1. Deadline of transition t_1 from TPN_2 is increased to 15 and then the transition t_1 from TPN_1 will be executed first (AcR1 and AcR2). In this mode after first run of self-adaptive system, a new configuration derives.

	TPN ₁			TPN ₂			TPN ₃	
	t_0	t_1	t_2	t_0	t_1	t_2	t_0	t_1
Sfct(t)	0	5	12	0	3	5	3	7
Slft(t)	7	15	20	8	9	15	8	11
d(t)	5	7	3	3	2	3	4	2
Sfst(t)	0	7	14	0	5	9	3	7
End at	5	14	17	3	7	12	7	9

Tab. 3. Simulation results of SA-PNS.

VI. CONCLUSION

Optimization options can be upgraded with new features. Until now we proposed a self-adaptive complex Petri nets system that searches the better solutions for a system with concurrent processes. The cost function for tokens can be implemented with the lowest cost solution, accepted delay ratio or the maximum additional costs and maximum additional tokens.

The advanced problems are based on complex systems. Because the complex systems are extensions of ordinary systems, the solutions for these problems are available in solutions for ordinary systems. The proposed problem depicted concurrent processes organized in a system with shared resources. Solutions obtained from HC-PNS are solutions for standard Petri nets.

Self-adaptive algorithms applied to complex systems provide the possibility to optimize current solutions. This

optimization can be applied in problems from many domains like programming, real time systems, economic, human and resource management in organizations or industry.

REFERENCES

- [1] Murata, T., 1989. Petri nets: properties, analysis and applications. Proceedings of the IEEE 77 (4), 541–580.
- [2] W.M.P. van der Aalst, Three Good Reasons for Using a Petri-net-based Workflow Management System, Proceeding of the International Working Conference on Information and Process Integration in Enterprises (IPIC96), 179-201, 1996.
- [3] P.M. Merlin, A Study of the Recoverability of Computing Systems, Ph.D. Thesis, Computer Science Department, University of California, Irvine, 1974.
- [4] P.M. Merlin, D.J. Farber, Recoverability of Communications Protocols - Implications of a Theoretical Study, IEEE Transactions on Communications, vol. 24, no. 9, 1036-1043, 1976.
- [5] I. A. Lomazova, Ph. Schnoebelen. Some Decidability Results for Nested Petri Nets. Ershov Memorial Conference: pp. 208-220, 1999.
- [6] I.A. Lomazova: Nested Petri Nets – a Formalism for Specification and Verification of Multi – Agent Distributed Systems. Fundamenta Informaticae 43 pp. 195-214, 2000.
- [7] I. A. Lomazova. Nested Petri nets: Multi level and recursive systems. Fundamenta Informaticae vol 47(3-4), pp. 283-293, 2001.
- [8] I. A. Lomazova. Modelling Dynamic Objects in Distributed Systems with Nested Petri nets. Fundamenta Informaticae 51(1-2), pp. 121-133, 2002.
- [9] K. Van Hee, I. Lomazova, O. Oanea, A.Serebrenik, N. Sidorova and M. Voorhoeve. Nested Nets for Adaptive Systems. In ICATPN '06, vol. 4024 of LNCS, pp. 241-260, 2006.
- [10] O. Captarencu(Prisecaru), T.Jucan. Interorganizational Workflows – an approach based on level Petri Nets. In Scientific Annals of the "Al. I. Cuza" University, Iasi, Computer Science Section, vol XIII, Iasi, pp. 17-38 2003.
- [11] O. Prisecaru and T. Jucan. Interorganizational Workflow Nets: a Petri Net Based Approach for Modelling and Analyzing Interorganizational Workflows. In: Proc. of the 4th International Workshop on Enterprise & Organizational Modeling and Simulation (EOMAS-2008), Montpellier, France, June 16-17, 2008, CEUR Workshop Proceedings, ISSN 1613 0073.
- [12] Balasubramanian, S., Brennan, R.W., Norrie, D.H., 2001. An architecture for metamorphic control of holonic manufacturing systems. Computers in Industry 46, 13–31.
- [13] Wyns, J., 1999. Reference Architecture for Holonic Manufacturing, PhD Dissertation, PMA Division, Katholieke Universiteit, Leuven, Belgium.
- [14] Brussel, H.V., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P., 1998. Reference architecture for holonic manufacturing systems : PROSA. Computers in Industry 37, 255–274.
- [15] Christensen, J., 1994. Holonic Manufacturing systems — initial architecture and standards directions. In: Proceedings of the First European Conference on Holonic Manufacturing Systems, Hannover, December 1994.
- [16] Koestler, A., 1967. The Ghost in the Machine. Hutchinson, London.
- [17] Giebels, M.M.T., Kals, H.J.J., Zijm, W.H.M., 2001. Building holarchies for concurrent manufacturing planning and control in EtoPlan. Computers in Industry 46 (3), 301–314.
- [18] Wullink, G., Giebels, M.M.T., Kals, H.J.J., 2002. A system architecture for holonic manufacturing planning and control (EtoPlan). Robotics and Computer- Integrated Manufacturing 18 (3–4), 313–318.
- [19] P.M. Merlin, A Study of the Recoverability of Computing Systems, Ph.D. Thesis, Computer Science Department, University of California, Irvine, 1974.
- [20] WPMC, Workflow Management Coalition Terminology and Glossary (WPMC-TC-1011), Technical Report, The Workflow Management Coalition, Brussels 1999.

Issues in Modeling and Control of Electro-Pneumatic EGR Valve Actuator for Automotive Engineering

C.-S. Teodorescu, S. Olaru, *Member, IEEE*

Abstract—Exhaust Gas Recirculation (EGR) is widely used in automotive industry. This paper describes a model responsible for conventional electro-pneumatic EGR valve actuator, part of an EGR system. It is intended to keep relatively simple, decoupled with respect to some engine phenomena, better oriented towards validating control laws. Special attention is paid to solenoid-actuator servomechanism subpart, by actually comparing, in simulation, a variety of linear and proposed nonlinear feedback control laws. The goal is to find full-state (locally) optimal controllers, in terms of performance, robustness, complexity and cost.

Index Terms—EGR, automotive engine airpath modeling, linear and nonlinear control, backstepping, LQR, solenoid actuators.

I. INTRODUCTION

Californian exhaust gas regulations have become a standard, worldwide accepted, concerning modern passenger cars, powered by internal combustion engines (see, e.g., [28]). The *EGR device* is used to address stricter government emission regulations, primarily concerning NO_x (see, e.g., [14]), but also customer demands for better fuel economy, plus more efficient drivability (see, e.g., [11]).

Basically, any EGR device is placed in the *airpath* chain of an automotive engine, and has the purpose to route some of exhaust gas (or mixture), from *exhaust manifold*, back into *intake manifold*. This phenomenon is addressed in literature by *recycling* or *recirculation* (see, e.g., [11], [13]).

In this paper, we focus solely on the cheapest solution of EGR device found on automotive market, that any manufacturer can employ nowadays, namely, conventional electro-pneumatic EGR. This device is servo-actuated by solenoid actuator, and has the main advantages in reduced costs, high drive power, light weight, while major disadvantages include instability, poor performance compared to other types found on automotive market, e.g., based on stepper motors or linear solenoid EGR (see, e.g., [30]).

Technically speaking, it is rather difficult to design a relatively simple model of EGR system, due to many nonlinear terms involved (like hysteresis of gas flow) and also, because of many *coupled* physical phenomena. This leads to a large and complex system with parameters having non-constant evolution over time, sometimes difficult to be measured or even estimated. In this paper, we present a simplified model, that we assume to be better suited for validating

control laws, by *decoupling* dynamics of EGR from Variable Geometry Turbocharger (VGT) and disregarding some engine parameters behavior (like fresh air cooling, recycled gas cooling, etc.). More complex phenomena, dealing with modeling coupled dynamics of EGR and VGT have been studied, e.g., in [22], while a controller based on passivation design was given in [27].

We anticipate a little bit further in-depth description of EGR model, by saying that, a subpart is the so-called *EGR poppet*. Actually, at this level, *regulation* of recycled exhaust gas takes place, by physically limiting *mass flow*. The primary goal of our research work is to find some efficient method for controlling recycled exhaust gas passing through EGR device. This topic naturally leads to controlling efficiently *EGR poppet* position. We point out the difficulty of this problem, due to increased complexity of overall nonlinear (hysteretical) system, as one will see hereafter. In this paper, we approach a more modest task, prior to approaching the harder, earlier mentioned one. Therefore, we focus solely on servo-actuator mechanism of EGR system, and effectuate an analysis of different linear and nonlinear control techniques. An ‘a priori’ requirement for control design, is that proposed solutions should be *robust* with respect to mechanical natural phenomena, like *wear* and *fouling* (see [18], [21]).

Unlike cheaper open-loop control solutions based on *static maps* (see, e.g., [22], [25]), used by automotive constructors, in this paper we analyze potential interest of closed-loop *state-feedback* linear and nonlinear control. The choice for each *feedback law* to be studied, has been made according to our knowledge in control engineering, and with respect to other solutions existing in nowadays automotive Research and Development (R&D) market.

The paper is structured as follows: during the first part we present in details the *EGR model*, with its seven subcomponents, while in the second part, we analyze different proposed controllers actuating the solenoid-actuator subcomponent. Validation follows, by running a set of numerical simulations.

II. EGR SYSTEM MODELING

The purpose of this section is to give in-depth explanations on the actual nonlinear *EGR system*, that we currently work with, formally translated into equations (1a)-(1h). For simplicity purposes, we separated *EGR system* dynamics from VGT behavior, assuming the two processes are independent (which, in reality, is not true). Our model is suitable only for naturally aspirated engines with external EGR.

C.-S. Teodorescu is with STITS, Univ. Paris-Sud, F-91405 Orsay, France
Catalin-Stefan.Teodorescu@u-psud.fr

S. Olaru is with SUPELEC Systems Sciences (E3S)-Automatic Control Department, F-91192 Gif-sur-Yvette, France
Sorin.Olaru@supelec.fr

$$\dot{x}_1 = x_2 \quad (1a)$$

$$\dot{x}_2 = \frac{1}{m_1} \left[-\beta_1 x_1 + \beta_2 - m_1 g \cos(\zeta_1) - b_{1v} x_2 - \beta_3 \frac{x_3^2}{x_1^2} - c_{1U}(x_1 - x_{1max})\sigma_1(x_1) + c_{1L}(x_{1min} - x_1)\sigma_2(x_1) \right] \quad (1b)$$

$$\dot{x}_3 = \frac{1}{L(x_1)} \left[-R_S x_3 - \frac{\partial L(x_1)}{\partial x_1} x_2 x_3 + U \right] \quad (1c)$$

$$\dot{x}_4 = \frac{RT_{dbf}}{V_{dbf}} \left[\frac{C_{D1}(x_1)A_{env}(x_1)p_0}{\sqrt{RT_0}} \Psi\left(\frac{x_4}{p_0}\right) - \frac{C_{D2}(x_1)A_{vac}(x_1)x_4}{\sqrt{RT_{dbf}}} \Psi\left(\frac{p_{vac}}{x_4}\right) \right] \quad (1d)$$

$$\dot{x}_5 = x_6 \quad (1e)$$

$$\dot{x}_6 = \frac{1}{m_5} \left[A_D(p_0 - x_4) - \beta_5 x_5 - \beta_6 - m_5 g \cos(\zeta_5) - b_{5v} x_6 - c_{5U}(x_5 - x_{5max})\sigma_5(x_5) + c_{5L}(x_{5min} - x_5)\sigma_6(x_5) + F_{exh}(x_8) \right] \quad (1f)$$

$$\dot{x}_7 = C_D(x_5) \left[\frac{A_{ppt1}(x_5)p_{exh}(x_8)}{\sqrt{RT_{exh}(x_8)}} \Psi\left(\frac{p_{int}}{p_{exh}(x_8)}\right) - \frac{A_{ppt2}(x_5)p_{int}}{\sqrt{RT_{int}}} \Psi\left(\frac{p_{exh}(x_8)}{p_{int}}\right) \right] \quad (1g)$$

$$\dot{x}_8 = r_1(t) \quad (1h)$$

To facilitate explanations, let us split EGR dynamics (1a)-(1h), represented schematically in Fig. 1, into seven different subparts, each to be analyzed separately, in the sequel. Naming conventions are according to literature and state-of-the-art. The experimented reader, familiar to this subject, will easily find his way through. We will briefly introduce the overall working principles in subsection II-B, and come back with more details, further on, in subsection II-C. Without loss of generality, and to simplify further explanations, let us assume that components of Fig. 1, as represented, are all spatially and geometrically placed perpendicular to the ground.

For convenience, Table I contains a list of all *state variables* from (1a)-(1h). Information is split into three columns: on the left we put the state-space variable, followed by the associated meaning, while on the third column are Standard International (SI) measurement units.

TABLE I

NOMENCLATURE OF STATE-SPACE VARIABLES IN MODEL (1).

x_1	position of solenoid plunger	m
x_2	velocity of solenoid plunger	m/s
x_3	current through solenoid coil	A
x_4	pressure inside diaphragm buffer	N/m ²
x_5	position of EGR poppet	m
x_6	velocity of EGR poppet	m/s
x_7	mass of recirculated exhaust gas	kg
x_8	crankshaft rotation angle	rad

A. Simplification hypotheses

Henceforth, let us define major simplification hypothesis, used to build model (1). Let us start with *buffer*, naming

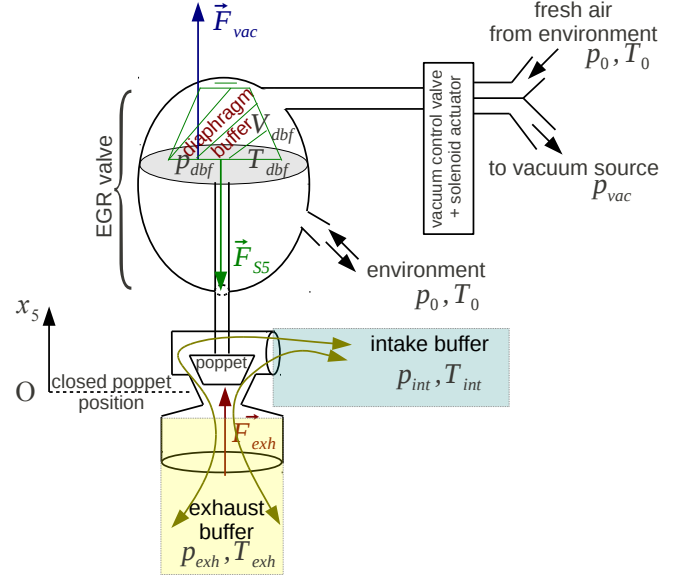


Fig. 1. Simplified schematics with working principle of EGR system.

convention used to refer a single-zoned constant volume, filled with gas, where inner pressure and temperature are uniformly distributed. Second simplification hypothesis consists of assuming that, exhaust gas passing through EGR has similar properties of ambient air (see, e.g., [11], [13]).

B. Overall EGR system working principle

We refer the reader to Fig. 1 and Table. I, as we would like to explain overall behavior of *EGR system* (1a)-(1h). In this subsection, we develop step-by-step a simple scenario divided into three phases, that will serve us as basis to

validate our model with feedback controllers in section IV. A *scenario* is a set of conditions defining a simulation or experimentation.

Although we anticipate a little bit further explicit developments, altogether with using new terminology not presented until now, let us start.

Phase I. We assume that initially, no *exhaust gas* is being recirculated from *exhaust buffer* towards *intake buffer* (see Fig. 1), due to quasi-closed *EGR poppet* position i.e. $x_5(t_0) \simeq 0$ (see Table I). As explained further on, during subsection II-C.1, *solenoid plunger* position $x_1(t)$ determines the quantities of vacuum and fresh air, eligible of passing through the *vacuum control valve*: when $x_1(t) \simeq x_{1min}$, meaning that plunger almost hits lower bumper (see Fig. 7), some air exits *diaphragm buffer* (naming conventionally used to designate the upper subpart of *EGR valve*), on its way towards *vacuum source* (see Fig. 1). As a result, pressure $x_4(t)$ inside *diaphragm buffer* falls within a certain time, reaching eventually the value $p_{vac} \simeq const$ (i.e. vacuum source pressure). The lower subpart of *EGR valve* is made up of a non-isolated region with respect to *ambient*, such that, inside it, constant pressure p_0 is always ensured. Pressure difference $\Delta p(t)_2 = x_4(t) - p_0$ gives rise to *pressure force* F_{vac} , overcoming spring force F_{S5} and pulling upwards the rod and *EGR poppet* (see Fig. 1). Now that *EGR poppet* is opened, as long as $p_{exh} > p_{int}$, exhaust gas can pass through it, on its way towards *intake manifold* (see Fig. 10).

Phase II. By efficiently controlling *solenoid plunger* position, such that it reaches $x_1(t) \simeq x_{1max}$, we try to avoid hitting the upper bumper (see Fig. 7). Fresh air from *ambient* passes through *vacuum control valve*, on its way to *diaphragm buffer*; pressure $x_4(t)$ rises to p_0 , while force balance $|F_{S5}| > |F_{vac}|$, leads to *EGR poppet* closing its *air gap*. Exhaust gas mixture is no longer being recirculated (see Fig. 10).

Phase III. Simulations end as we set desired plunger position back to its original value (see Fig. 7 and Fig. 10).

C. EGR system components

1) *EVR. Solenoid-actuator*: The EGR Vacuum Regulator (EVR) is used to servo-actuate the pneumatic *EGR valve* by controlling the ratio of vacuum/fresh air towards *EGR valve* (details in subsections II-C.5 and II-B). EVR mechanism can be divided into two subparts: *vacuum control valve* and *solenoid actuator*.

Working principle for solenoid actuators detailed, e.g., in [12, pp. 227–231], can be synthesized as follows: the *plunger*, term used to name a linearly moving part, made of steel, is actuated due to the joint action of magnetic force (pushing in one direction) and spring force (pulling in the opposite direction). Newton's second law applied on plunger of lumped mass m_1 , assumed to be rigid and having a single degree of freedom (DOF) – see [16], [17], namely linear motion, gives rise to equations (1a)-(1b), where x_1 is plunger position expressed with respect to an (imaginary) Ox_1 axis perpendicular to ground; x_2 is plunger velocity;

spring force is equal to $-\beta_1 x_1 + \beta_2$ with β_1 positive-value *stiffness*; β_2 positive-value spring *preload force*; gravity force $G_1 = m_1 g \cos(\zeta_1)$ with g gravitational acceleration, ζ_1 angle between any perpendicular to Earth's ground and solenoid plunger's longitudinal axis; dynamic frictional force $F_{1f} = b_{1v} x_2$ with b_{1v} kinetic friction (or viscous damping) coefficient (see [16]); static friction, stick slip are neglected; magnetic force $F_{mag} = \beta_3 \frac{x_2^2}{x_1^2}$ with $\beta_3 = \frac{\mu_0 N^2 S}{8}$, μ_0 permeability of free space or air, N number of solenoid coil turns, S transverse surface of plunger (see [15]); elasto-plastic bumpers positioned at x_{1min} and x_{1max} are modeled as virtual springs of first-order, with very high coefficient constants c_{1U} (upper bumper) and c_{1L} (lower bumper); two discontinuous functions are being used:

$$\sigma_1(x_1) = \begin{cases} 0 & \text{if } x_1 < x_{1max} \\ 1 & \text{elsewhere} \end{cases}$$

$$\sigma_2(x_1) = \begin{cases} 0 & \text{if } x_1 > x_{1min} \\ 1 & \text{elsewhere} \end{cases}$$

where x_{1max} and x_{1min} are real positive constants.

Although behavior of subsystem (1a)-(1b) is continuous, since we have introduced both continuous and discontinuous elements, we eventually get a dynamical nonlinear *hybrid system* (see, e.g., [5], [26]).

State-space equation (1c) results by applying equivalent electrical circuit law on solenoid actuator: $U = R_S x_3 + \frac{d\lambda}{dt}$ with U input control voltage, R_S is equivalent resistance of solenoid; state-space variable x_3 is current through windings; $\lambda \triangleq L(x_1) \cdot x_3$ is flux linkage (see [15]); $L(x_1)$ is solenoid self-inductance, the choice including, e.g., first order polynomial (see [8, Ch. 11]) $L(x_1) = \frac{L_0}{L_1 + x_1}$, or second order polynomial (see [15]) $L(x_1) = L_0 + \frac{L_1}{x_1^2}$, with L_0 and L_1 constant-value parameters, to be determined experimentally. Explicitly, we get eventually differential equation (1c).

2) *Vacuum source*: This subcomponent of *EGR system*, introduces parameter p_{vac} in (1d), assumed to be quasi-constant.

3) *Environment (or ambient)*: Fresh air from *environment* is assumed to be at standard National Institute of Standards and Technology (NIST) conditions: temperature T_0 , pressure p_0 in (1d) and adiabatic constant γ in (2) (see [11]).

4) *Crankshaft*: Roughly speaking, automotive engines have a certain number of cylinders, 4 in most European passenger cars, situation focused during present paper. Each cylinder has a piston which produces linear motion, converted into rotation by *crankshaft*. Let us define in (1h) *crankshaft rotation angle* x_8 , with angular velocity $\dot{x}_8$ assumed to be constant throughout simulations, equal to r_1 .

5) *EGR valve*: The *EGR valve* subsystem consists of an inner diaphragm (assumed of zero mass), dividing the former into two isolated regions. The upper region encloses gas mass m_{dbf} , has inner pressure x_4 , temperature T_{dbf} , volume V_{dbf} , all appearing in (1d), and a weightless spring fixed on one side, generating force F_{S5} in Fig. 1. This linear motion is transferred to a rod situated on the second, lower region side. This rod ends with *EGR poppet*.

Imposed regulations ask for *non-choked* gas flow conditions (see [14]). Therefore, inner pressure is modeled using *standard orifice flow function* (see [13, Appendix C], [23]), concerning only sub-critical flow of gas:

$$\Psi(\alpha) = \alpha^{1/\gamma} \sqrt{\frac{2\gamma}{\gamma+1} \left(1 - \alpha^{\frac{\gamma-1}{\gamma}}\right)} \quad (2)$$

$$\text{with } \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}} \leq \alpha < 1$$

represented visually in Fig. 4(a). The empirical equation of one-dimensional steady compressible flow through an orifice is (see, [13, pp. 55 and Appendix C]):

$$\dot{m} = \frac{C_D A_{orif} p_{res}}{\sqrt{RT_{res}}} \Psi\left(\frac{p_{orif}}{p_{res}}\right) \quad (3)$$

with m mass of gas; C_D discharge coefficient (see [13]), which is non-constant for any gas that is not ideal. In reality, *exhaust mixture* does not behave as an ideal gas; since it exhibits pretty different properties, discharge coefficient functions should be experimentally identified: $C_{D1}(x_1)$ and $C_{D2}(x_1)$ in (1d), $C_D(x_5)$ in (1g). For the purpose of our study, we assume ideal gas always being used, and thus we set all discharge coefficients to 1. p_{orif} is pressure at orifice level; p_{res} is pressure inside a *large* reservoir where an orifice has been drilled, of opening area A_{orif} , which in our case corresponds to experimentally identified functions $A_{env}(x_1)$ and $A_{vac}(x_1)$ appearing in equation (1d). For simulation purpose, we modeled both $A_{env}(x_1)$ and $A_{vac}(x_1)$ as first order functions (see Fig. 3).

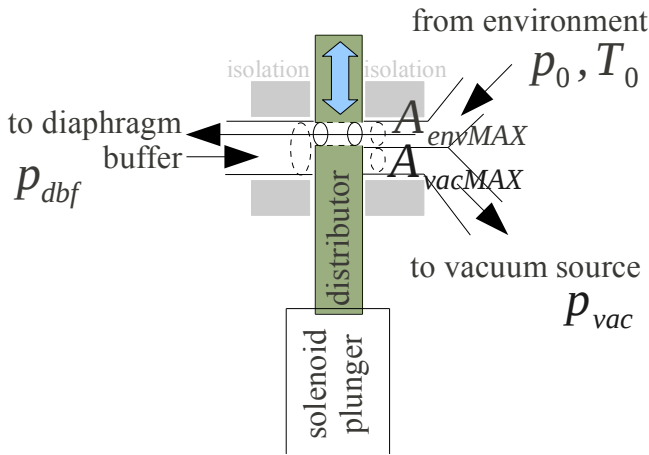


Fig. 2. Vacuum control valve. Shifting of air through opening areas $A_{env}(x_1)$ and $A_{vac}(x_1)$; $x_1 = x_{1max}$.

Remark 2.1: In practice, both curves in Fig. 3 should be experimentally adjusted. $\square$

Briefly speaking, equation (1d) is derived from *perfect gas equation* (see [13, Appendix B], [20], [21])

$$\dot{p}_{dbf} V_{dbf} = \dot{m}_{dbf} R T_{dbf} \quad (4)$$

with

$$\dot{m}_{dbf} = \dot{m}_{in} - \dot{m}_{out} \quad (5)$$

where m_{in} is the mass of gas coming in, while m_{out} is mass of gas exiting; the *specific (or individual) gas constant* $R \triangleq \frac{R_*}{M}$, where R_* is (molar) gas constant ($8.314 \text{ J K}^{-1} \text{ mol}^{-1}$) and M is molar mass of gas/mixture. This notation for R is consistent with (3).

Briefly speaking, equation (1d) contains two terms: the left one is responsible for pressure x_4 rise (since it is always positive definite), due to fresh ambient air coming in, while the right term acts on pressure x_4 drop (notice minus sign), as air flows towards *vacuum source*.

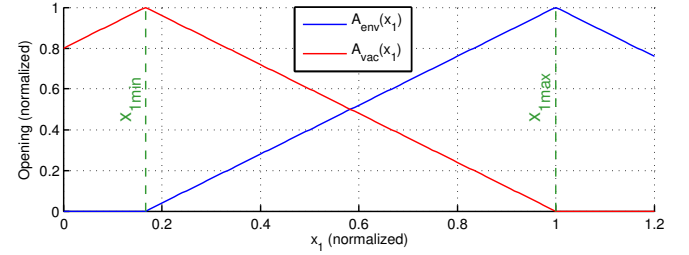


Fig. 3. Normalized opening areas functions $A_{env}(x_1)$ and $A_{vac}(x_1)$.

One may notice that, in Fig. 3, we have assumed (for simplicity) that, maximum opening areas for $A_{env}(x_1)$ and $A_{vac}(x_1)$ are equal; the opening area of *distributor* (see Fig. 2) is assumed to be equal to this maximum, also. In reality, depending on actual EGR geometry, these values might be slightly different.

In Fig. 4(b) we simulate x_4 from (1d) behavior: notice different values for *settling time*, higher for falling than rising, mainly due to nonlinear function $\Psi(\alpha)$ in (2) and Fig. 4(a). This is classic behavior of any nonlinear dynamical system (see, e.g., [19]). To draw these curves, we maintained $A_{env}(x_1) = A_{vac}(x_1)$; $T_{dbf} = T_0$; $p_{vac} = 59 \text{ (kPa)}$.

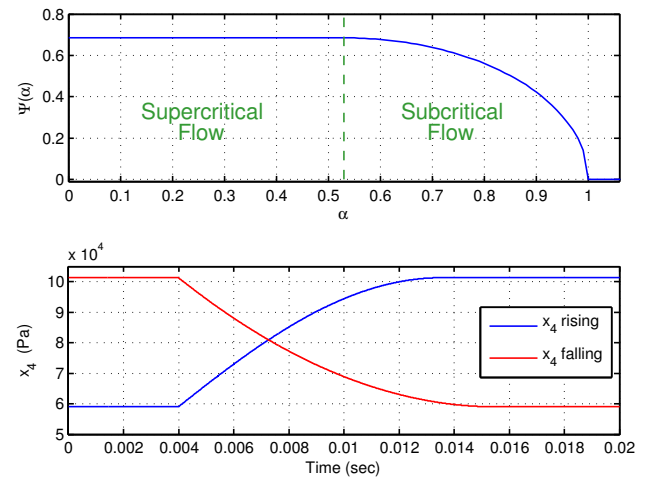


Fig. 4. (a) Standard orifice flow function $\Psi(\alpha)$; (b) Pressure x_4 .

In Fig. 1, black arrows were used to indicate the flow of fresh air, to and from *diaphragm buffer*. Depending on

solenoid plunger position x_1 , two possibilities arise: whether fresh air coming from *environment*, passes through EVR and enters *diaphragm buffer*, otherwise it exits *diaphragm buffer* on its way towards *vacuum source*.

In the sequel, let us consider the subsystem made up of *rod* plus *poppet*; assume both are rigid, of total mass m_5 and have a single DOF, namely linear motion (see [16]). By applying Newton's second law of motion on m_5 (or d'Alembert principle, as in [16], [17]), with respect to (imaginary) Ox_5 axis oriented perpendicular to ground, one gets equations (1e)-(1f); vacuum force $F_{vac} = A_D(p_0 - p_4)$, with A_D surface of diaphragm; first-order spring force $F_{S5}(x_5) = -\beta_5 x_5 - \beta_6$, with β_5 positive-value stiffness and β_6 positive-value spring preload force; dynamic frictional force $b_{5v}x_6$, with $b_{5v} > 0$ kinetic friction coefficient (or viscous damping, see [16]); gravity force $m_5 g \cos(\zeta_5)$, with ζ_5 angle between any perpendicular to ground and EGR *rod's* longitudinal axis. High-temperature resistant bumpers positioned at x_{5min} and x_{5max} , are modeled as virtual springs of first-order, with very high coefficient constants c_{5U} (upper bumper) and c_{5L} (lower bumper); two discontinuous functions are used:

$$\sigma_5(x_5) = \begin{cases} 0 & \text{if } x_5 < x_{5min} \\ 1 & \text{elsewhere} \end{cases}$$

$$\sigma_6(x_5) = \begin{cases} 0 & \text{if } x_5 > x_{5max} \\ 1 & \text{elsewhere} \end{cases}$$

Last term in (1f), the exhaust force $F_{exh}(x_8)$ should be experimentally identified. It depends primarily on pressure difference $\Delta p_1 = p_{exh}(x_8) - p_{int}$. For simulation purposes, we have used a cubic interpolation function.

6) *Intake and Exhaust buffers*: Let us introduce two virtual models: *intake buffer*, corresponding to *intake manifold* and containing a mixture of *fresh air*, more than 70% of total, and some *recirculated exhaust gas*; *exhaust buffer*, corresponds to *exhaust manifold*, which virtually contains only *exhaust gas* exiting engine's cylinders (see, e.g., details in [11], [13]).

Intake buffer introduces in (1g), constant parameters p_{int} and T_{int} , which stand for uniformly distributed pressure and temperature. *Exhaust buffer* adds to (1g), $p_{exh}(x_8)$ and $T_{exh}(x_8)$, representing uniformly distributed pressure and temperature, respectively.

Both $p_{exh}(x_8)$ and $T_{exh}(x_8)$ were modeled using a sum of Gaussian curves. We hope to count this aspect for originality of this paper. Experimental data from [13] was employed to shape $p_{exh}(x_8)$; for $T_{exh}(x_8)$, only the main peaks were modeled due to our lack of experimental results. In Fig. 5, we represent visually both curves evolution for two complete rotations of crankshaft i.e. $x_8 \in [0, 4\pi](\text{rad})$ with $4\pi = 720^\circ(\text{deg})$.

Each curve of modified Gaussian function is according to formula (see [10])

$$g(x_8, \xi_O, \xi_A, \xi_W, \xi_C) = \xi_O + \xi_A \exp\left(-\frac{(\xi_C - x_8)^2}{0.3606 \cdot \xi_W}\right)$$

with ξ_O horizontal offset, ξ_A amplitude, ξ_W width, ξ_C center of the Gaussian.

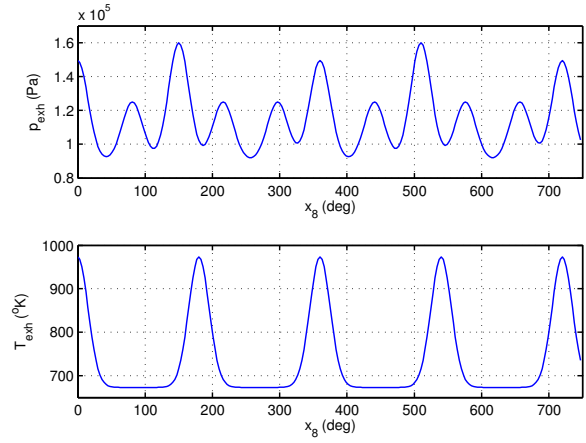


Fig. 5. Exhaust buffer: (a) pressure $p_{exh}(x_8)$ and (b) $T_{exh}(x_8)$. We assume $\dot{x}_8 = 1200$ rev/min.

7) *EGR poppet*: This subpart, solely, is responsible for equation (1g). In normal working engine conditions, the actual mass of recirculated gas mixture x_7 passes from *exhaust buffer*, through poppet, towards *intake buffer*, phenomenon which is ensured by the higher pressure p_{exh} with respect to p_{int} . To model it, an equation similar to (3) would be enough. However, in practice, unavoidable *valve overlap* occurs i.e. for short periods of time, during each running cylinder cycle, pressure difference reverses sign, as $p_{exh} < p_{int}$. Hence, fresh air from *intake buffer* flows towards *exhaust buffer* (see, e.g., [11]). Because of this ubiquitous phenomenon, we have chosen to model it using similar relations to (3) and (5), eventually resulting equation (1g).

In (1g), opening area parameters $A_{ppt1}(x_5)$ and $A_{ppt2}(x_5)$ were calculated according to [11]:

$$A_{ppt1}(x_5) = \pi d_{ppt1} x_5$$

$$A_{ppt2}(x_5) = \pi d_{ppt2} x_5$$

with d_{ppt1} outer diameter and d_{ppt2} inner diameter of poppet (see Fig. 6).

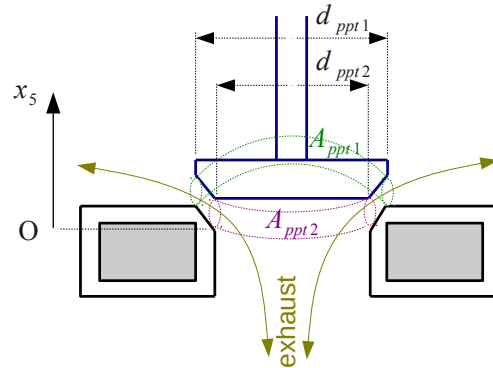


Fig. 6. Scheme of EGR poppet geometry.

Double-sided dark-yellow arrows in Fig. 6 and Fig. 1

indicate bidirectional *exhaust gas* flow ability, modeled in equation (1g). Let us point out that, equation (1g) contains briefly two terms, the first one including $\Psi(\alpha_1)$, while the second one has $\Psi(1/\alpha_1)$, with $\alpha_1 = p_{int}/p_{exh}(x_8)$. Consequently, since $\Psi(\alpha_1)$ is zero for any $\alpha_1 \geq 1$ (see Fig. 4), only one of these terms is operative at a time.

Remark 2.2: If it proves out that, in practice, modeling *mass flow* $\dot{x}_7$ as in (1g), is not accurate enough with respect to experimental data (e.g., due to properties of exhaust gas mixture), one might try using *van der Waals equation* instead of *perfect gas equation* (4) (see, e.g., [20], [24]). This choice might also affect (2). $\square$

III. CONTROLLER SYNTHESIS

Let us recall the purpose of *EGR system* (1), namely to be able to regulate the quantity of *gas mixture*, routed back into *intake buffer*, process that takes place at *EGR poppet* level. In order to servo-actuate it, one should control efficiently the solenoid-actuator's position x_1 .

In this paper, we propose a different approach with respect to cheaper open-loop control or pre-configured *static maps* (see, e.g., [22], [25]), by proposing full state-feedback closed-loop controllers. For this reason, let us introduce the following acronyms, responsible for the different control design methods to be analyzed hereafter: LQR (Linear Quadratic Regulator) and PoP (Pole Placement) [3]; SMC (Sliding Modes Continuous control) and SMD (Sliding Modes Discontinuous control) [1]; BBC (Backstepping-Based Continuous control) and BBD (Backstepping-Based Discontinuous control).

In [29], an experimental platform, having an electropneumatic system, is efficiently controlled using backstepping.

Linear control (like LQR, PoP) is more appealing to actual needs of automotive industry, while nonlinear control (e.g., SMC, SMD, BBC, BBD) might prove its interest, in terms of increased reliability, accuracy, and applicability to modern passenger cars, in the near-future, once advanced technology of micro-controller with sustained reliability becomes cheaper and widely accessible for industrial purposes.

Let us rewrite solenoid actuator subsystem equations (1a)-(1c), by ignoring bumpers:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{1}{m_1} \left[-\beta_1 x_1 + \beta_2 - m_1 g \cos(\zeta_1) - b_{1v} x_2 - \beta_3 \frac{x_3^2}{x_1^2} \right] \\ \dot{x}_3 = \frac{1}{L(x_1)} \left[-R_S x_3 - \frac{\partial L}{\partial x_1} x_3 + U \right] \end{cases} \quad (7)$$

A. Linearized LTI Model

A linearized model of system (7), will serve us as basis for calculating linear controllers. The simplest Linear Time-Invariant (LTI) model can be calculated, by applying linearization method at a specified, fixed *operating point* (see, e.g., [4], [5], [7, pp. 81–83]); this method is also known as first order Taylor approximation [2], [3]. Let us write

$$\begin{cases} x = x_e + \tilde{x} \\ U = U_e + \tilde{U} \end{cases}$$

with $x = [x_1 \ x_2 \ x_3]^T$ being the column vector of *state variables* (the symbol T denotes transpose); $x_e \triangleq [x_{1e} \ x_{2e} \ x_{3e}]^T$ and U_e form together, the equilibrium (x_e, U_e) , (see, e.g., [9, Def. 3.1, Ch. 3]); $\tilde{x}$ and $\tilde{U}$ are *small variations* around equilibrium.

By setting up all functions on the right side of *equal sign* in (7), to zero, one gets equilibrium $x_{1e} = \frac{x_{1max} - x_{1min}}{2}$; $x_{2e} = 0$; $U_e = R_S x_{3e}$; $x_{3e} = x_{1e} \sqrt{\frac{-\beta_1 x_{1e} + \beta_2 - m_1 g \cos(\zeta_1)}{\beta_3}}$. Next, linearizing around (x_e, U_e) , using Jacobian matrix, one gets matrices A and B , and therefore linearized system $\dot{\tilde{x}} = A\tilde{x} + B\tilde{U}$.

B. Linear Control

Based on linearized model presented in subsection III-A, LQR and PoP control laws can be explicitly calculated, resulting simple feedback law (see [2]–[4]):

$$U(x) = -K_1^T(x - x_e) - k_2 x_{1ref} + U_e \quad (8)$$

with $K_1 \in \mathbb{R}^n$ constant gain vector calculated using *algebraic Riccati equation* (for LQR) or basic matrix operations like sum, multiplication, inversion (for PoP); $n = 3$; $k_2 \in \mathbb{R}$ is a constant value; $x_{1ref}(t)$ is desired reference trajectory for $x_1(t)$, e.g., square-wave signal, or a piecewise quasi-constant signal.

C. Nonlinear Control vs. Linear Control

In this paper, we point out a comparison between different control designs, guiding the reader to known literature, for full explicit calculations of nonlinear controllers. To offer an insight of nonlinear analytical control laws complexity, with respect to linear situation presented in subsection III-B, one may see BBC in (6), where $k_i > 0$, ($i \in \{1, 2, 3\}$).

Remark 3.1: All nonlinear control laws to be treated in this paper (i.e. SMC, SMD, BBC, BBD) have the same order of complexity.

IV. NUMERICAL SIMULATIONS

Simulation results given in this section were normalized, therefore offer only qualitative indicators for analyzing our results. The ODE solver was set to Runge-Kutta, fixed-step, with sampling time set to $T_s = 1(\text{ms})$. All closed loop behavior, with linear or nonlinear controller is validated (in simulation) on the nonlinear EGR model (1).

Fig. 7(a), Fig. 8(a) and Fig. 9(left column) refer to the same set of simulation scenario, running without perturbation. Similarly, Fig. 7(b), Fig. 8(b) and Fig. 9(right column) refer to a second simulation scenario, involving additive *uniformly distributed white-noise* ν_i , applied to each state-variable x_i ($i \in \{1, 2, 3\}$). Simulated *white-noise* has zero average, standard deviation of 3% of maximum nominal value $|x_{i,max} - x_{i,min}|$, and stands for actual electrical noise, plus faulty electrical equipment. In this framework, we have used noise as an indicator for *robust stability*. Simulations were reconducted by replacing noise ν_i with *normally distributed white noise*. Consequently, feedback system keeps *robust stability* property.

$$\begin{aligned}
U_{\text{BBC}}^{\text{EGR}} = & R_S x_3 + \frac{\partial L}{\partial x_1} x_3 - \frac{L(x_1)}{2} x_3 \left[(k_1 + k_2 + k_3) - \frac{b_1}{m_1} \right] + \frac{L(x_1)}{2x_3} \left[\frac{x_1}{\beta_3} \left(-\beta_1 x_1 + \beta_2 - m_1 g \cos(\zeta_1) - b_1 x_2 \right. \right. \\
& \left. \left. + m_1 \left((k_1 + k_2)(x_2 - \dot{x}_{1ref}) + (1 + k_1 k_2)(x_1 - x_{1ref}) - \ddot{x}_{1ref} \right) \right) (k_3 x_1 + 2x_2) \right. \\
& \left. + \frac{x_1^2}{\beta_3} \left[-\beta_1 x_2 - (-\beta_1 x_1 + \beta_2 - m_1 g \cos(\zeta_1) - b_1 x_2) \left(\frac{b_1}{m_1} - (k_1 + k_2) \right) - m_1 (k_1 + k_2) \ddot{x}_{1ref} \right] \right. \\
& \left. + \frac{m_1 x_1^2}{\beta_3} \left((1 + k_1 k_2)(x_2 - \dot{x}_{1ref}) - \ddot{x}_{1ref} \right) + \frac{\beta_3}{m_1 x_1^2} \left(x_2 - \dot{x}_{1ref} + k_1 (x_1 - x_{1ref}) \right) \right]
\end{aligned} \tag{6}$$

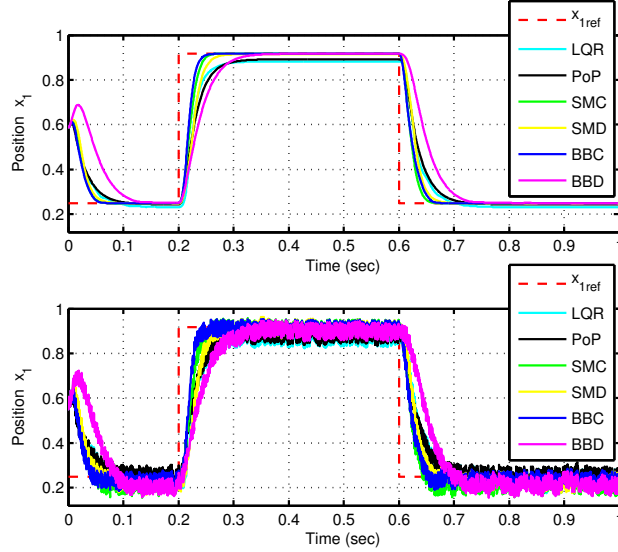


Fig. 7. Normalized regulation comparative results: plunger position for (a) unperturbed system; (b) system with additive noise on each state-variable.

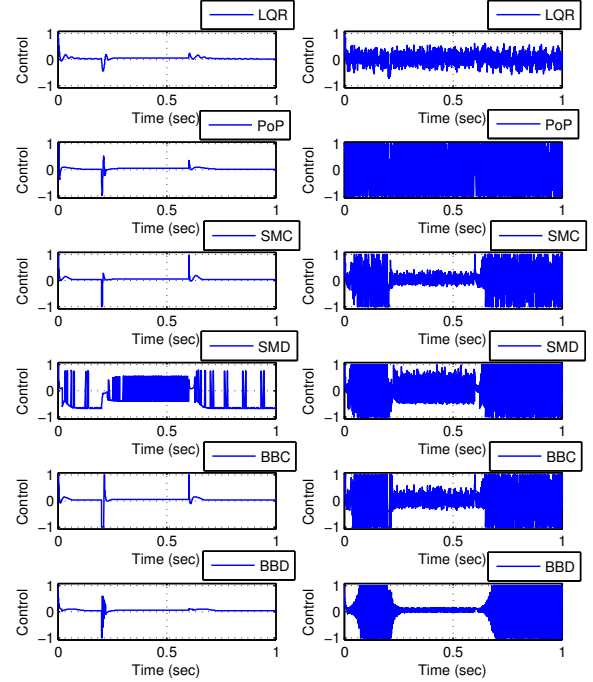


Fig. 9. Each column corresponds to normalized control law U responsible for results: in Fig.7(a) and 8(a) - (left column); in Fig.7(b) and 8(b) - (right column).

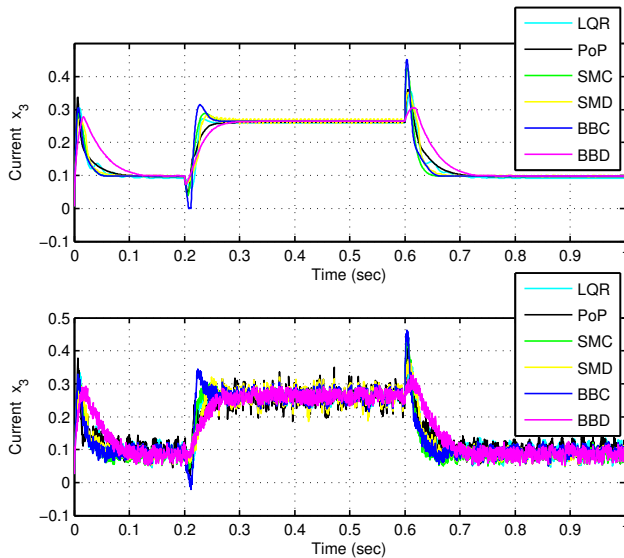


Fig. 8. Normalized current x_3 : (a) nominal system runs unperturbed; (b) additive white-noise is applied on each state-variable.

In Fig. 7 we compare results in terms of *regulation performance*. As expected, there is a ‘small’ *steady-state error* when applying linear with respect to any other nonlinear controllers, studied in this framework. This phenomenon is due to loss of accuracy when linearizing. However, it is possible to reduce this *steady-state error*, by choosing different weight matrices inside *objective function* for LQR, and place poles differently for PoP, respectively. All this, to the detriment of robustness: closed-loop system becomes more ‘sensitive’ to any structured uncertainties, sometimes resulting instability. Other methods for reducing the offset might include, e.g., adding an integrator to controller scheme or designing an appropriate (asymptotic or reduced order) observer (as in [2], [6]).

In Fig. 7(a) we analyze *time-responses* by comparing feed-

back loop results: notice *settling time* to within 5%, ranges from 50ms (for SMC and BBC) to 70ms (SMD), 80ms (LQR) and 100ms (BBD). All values are within automotive manufacturer demands for EGR solenoid-actuator.

In Fig. 8 we plot actual current x_3 time-evolution. There is a structural constraint for this particular type of actuator, demanding that current rests within normalized *admissible range* $[-1, 1]$. Values outside this domain, might damage the actuator, overheating, or even burn it. Both simulation scenarios (with or without white-noise) prove this constraint is guaranteed.

Control law U was plotted in Fig. 9. Notice, when adding white-noise to the system, regulation goal, robustness and stability are all ensured, although nonlinear controllers run to the saturation limit.

In Fig. 10, one can visualise simulated *recycled gas mass* x_7 and *mass flow* $\dot{x}_7$, passing through *EGR poppet*, with LQR driving EVR mechanism. Results were not normalized.

We conclude upon this section by saying that, simulation results prove numerically stable, robust with respect to additive white-noise and efficient-enough to ensure automotive manufacturer demands.

V. CONCLUSIONS

This paper deals with controlling a solenoid servomechanism, subpart of an EVR subsystem, itself part of a larger *EGR system*. First, we present a simplified model dynamics and construct eight coupled *state-space equations*. Second, we pass to feedback control laws synthesis. Different designs of linear and nonlinear controllers were studied in this framework, validating results through an extensive set of simulations. We compared effects and interactions between *state variables*. *Control system* properties, like stability, performance and robustness, were tested to respect actual automotive industry needs.

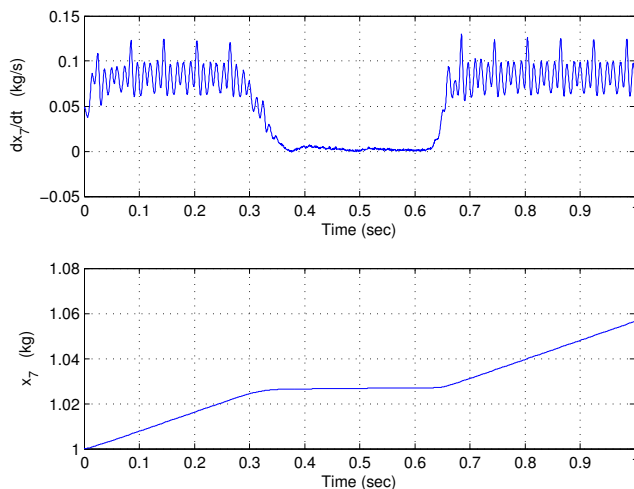


Fig. 10. (a) Simulated recycled gas mass flow $\dot{x}_7$; (b) recycled gas mass x_7 .

REFERENCES

- [1] V.I. Utkin. *Sliding Modes in Control and Optimization*. Springer-Verlag, Berlin, 1992.
- [2] W. Bolton. *Control Systems*. Elsevier, Oxford, 2002.
- [3] B.D.O. Anderson, J.B. Moore. *Optimal Control: Linear Quadratic Methods*. Prentice-Hall, New Jersey, 1990.
- [4] J. P. Hespanha. *Linear Systems Theory*. Princeton University Press, Princeton, New Jersey, 2009, ch. I.2 and VI.20.
- [5] A.J. van der Schaft, H. Schumacher. *An Introduction to Hybrid Dynamical Systems, Lecture Notes in Control and Information Sciences*. vol. 251, Springer, London, 1999.
- [6] F. Bonnans and P. Rouchon. *Commande et Optimisation de Systèmes Dynamiques*. Éditions de l'École Polytechnique, Palaiseau, France, 2005.
- [7] W.S. Levine (Ed.). *The Control Handbook*, 1st edition. CRC Press–IEEE Press, Boca Raton, Florida, 1996.
- [8] H. Khalil. *Nonlinear Systems*. Prentice-Hall, Upper Saddle River, NJ, 2002.
- [9] J.-M. Coron. *Control and Nonlinearity*. AMS, Providence, 2007.
- [10] A.D. Poularikas. *The Handbook of Formulas and Tables for Signal Processing*. CRC Press, Florida, 1999.
- [11] W.W. Pulkrabek. *Engineering Fundamentals of the Internal Combustion Engine*. Prentice Hall, New Jersey, 2004.
- [12] W.B. Ribbens. *Understanding Automotive Electronics*. 6th ed., Elsevier Science, New York, 2003.
- [13] J.B. Heywood. *Internal Combustion Engine Fundamentals*. McGraw-Hill, New York, 1988.
- [14] N. Kruse, A. Frennet, J.-M. Bastin (editors). *Catalysis and Automotive Pollution Control IV*. Elsevier, New York, 2005.
- [15] J.R. Brauer. *Magnetic Actuators and Sensors*. Wiley-IEEE Press, New Jersey, 2006.
- [16] R.W. Clough, J. Penzien. *Dynamics of Structures*. 3rd ed., Computers & Structures, Inc., Berkeley, 2003.
- [17] F. Axisa. *Modelling of mechanical systems: Discrete Systems, Volume 1*. Kogan Page Science, London, 2004.
- [18] R.G. Bayer. *Mechanical Wear Fundamentals and Testing*. Marcel Dekker, New York, 2nd ed., 2004.
- [19] P. Müllhaupt. *Introduction à l'analyse et à la commande des systèmes non linéaires*. PPUR, Lausanne, 2009.
- [20] R.D. Zucker, O. Biblarz. *Fundamentals of Gas Dynamics*. 2nd ed., J. Wiley & Sons, New Jersey, 2002.
- [21] E.J. Shaughnessy, I.M. Katz, J.P. Schaffer. *Introduction to Fluid Mechanics*. Oxford University Press, New York, 2005.
- [22] I. Kolmanovsky, P. Moraal, M. van Nieuwstadt, A. Stefanopoulou. Issues in modelling and control of intake flow in Variable Geometry Turbocharged engines. In Proc. European Control Conf., Brussels, Belgium, 1997.
- [23] P. Moraal, I. Kolmanovsky, M. van Nieuwstadt. Modeling and identification of a current to vacuum transducer and VNT actuator. In Proc. IEEE/ASME, Atlanta, USA, September 19-23, 1999.
- [24] C.A. Mugruza Vassallo. Van der Waals and ideal gas models for compressibility by means of pressure in pneumatic pipes from 1 to 100 lpm. In Proc. 26th Annual Int. Conf. of the IEEE EMBS, San Francisco, CA, USA, September 1-5, 2004.
- [25] J. Chauvin, G. Corde, N. Petit and P. Rouchon. Experimental motion planning in airpath control for HCCI engine. In Proc. American Control Conf., Minneapolis, Minnesota, USA, June 14-16, 2006.
- [26] A. Bemporad, M. Morari. Control of systems integrating logic, dynamics, and constraints. In *Automatica*, volume 35, 1999, pages 407–427.
- [27] P. Kokotović, M. Arcak. Constructive nonlinear control: a historical perspective. In *Automatica*, volume 37, 2001, pages 637–662.
- [28] O. Gassmann. Multicultural teams: Increasing creativity and innovation by diversity. In *Creativity and Innovation Management*, volume 10, no. 2, Jun. 2001, pages 88–95.
- [29] M. Smaoui, X. Brun, D. Thomasset. A study on tracking position control of an electropneumatic system using backstepping design. In *Control Engineering Practice*, volume 14, 2006, pages 923–933.
- [30] T. Miyake, H. Okada. An electrically operated EGR valve that reduces automotive emissions. In *J. Mitsubishi Electric Advance*, volume 87, 1999, pages 20–24.

Central Pattern Generator Control of a Differential Wheeled Robot

Ionuț Urziceanu, Fernando Herrero-Carrón, Juan González-Gómez, Mircea Nițulescu, *Member, IEEE*,
Francisco de Borja Rodríguez, Pablo Varona

Abstract—Bio-inspired central pattern generator approaches have been used for robotic autonomous locomotion mainly in legged, serpentine and worm like robots. In this paper we propose a new bio-inspired central pattern generator (CPG) neural circuit to control a differential wheeled robot. The CPG uses bursting neurons coupled through time evolving synapses, which endows them with rich rhythm negotiation capabilities. The overall circuit activity produces robust yet flexible rhythms. We show that this CPG can produce sequential command signals to rhythmically control the motion of the differential wheeled robot in a situation where the wheels can only partially rotate.

I. INTRODUCTION

CENTRAL pattern generators are neural circuits that autonomously generate signals that are used to produce rhythmic motion. Recent research has shown that their neurons have rich intrinsic oscillating dynamics and that their connection topology is built to negotiate their overall pattern of activity as a function of environmental conditions and external input from the central nervous system [1]. Inspiration from these biological neural circuits has been used to propose solutions for robot locomotion [2].

While CPG approaches have been used extensively to control legged, serpentine and worm like robots, it has not been widely explored in the context of wheeled robots. This is mainly because wheeled robots are usually thought to require a continuous control of the wheels [3] while CPG circuits produce rhythmic motion. In this paper we show how rhythmic motion can be an adequate strategy to control a wheeled robot in the case where the wheels can only partially rotate.

II. DIFFERENTIAL WHEELED ROBOT

A differential wheeled robot is a mobile robot whose locomotion is based on two (or more) wheels being controlled independently. The relationship between the rotation of each wheel determines the type of locomotion. In our case we use the SkyBot, which is a differential drive platform built for educational purposes. The customizable Skybot [4] (Fig. 1) is a differential drive robot composed of printable wheels

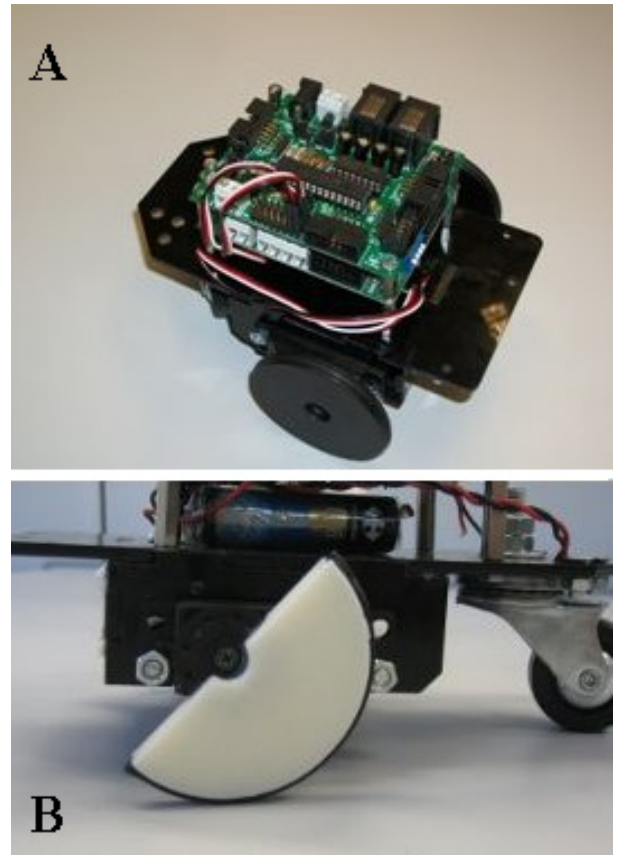


Fig. 1. A: Skybot: Overview of the differential drive robotic platform. B: SkyBot: Limited wheel detail (half-wheel).

and two hobby servos. The robot uses two electronic control boards called SkyPic [5] and Sky293 [6].

The SkyPic is a minimal design with only the necessary components for controlling the robot. It includes an 8-bit PIC16F876A microcontroller, headers for connecting the servos, an I2C bus for additional communication, serial connection to the PC, a test LED and a switch for resetting the circuit. The Sky293 board [6] is an extension board that was used to accommodate the sensors. We have considered the case in which the wheels of the robot cannot perform complete rotations: this limitation can be attributed to the actuators that are used (linear motors, SMAs, broken gearboxes, etc), to the environment (the wheel does not have enough space to rotate) or to the actual shape of the wheel (the wheel may be broken). Their movements are thus confined in an angular interval, let us say $[-\varphi_{\max}, \varphi_{\max}]$. In this case, a new locomotion principle can be applied such that the robot can still navigate, but with reduced mobility.

Manuscript received June 17, 2011. This work was supported by an Erasmus grant and MICINN BFU2009-08473, TIN-2010-19607.

FHC, FBR, PV. Authors are with Grupo de Neurocomputación Biológica (GNB), Departamento de Ingeniería Informática, Escuela Politécnica Superior, Universidad Autónoma de Madrid, 28049 Madrid. (email: pablo.varona@uam.es)

JGG is with the Robotics Lab, Universidad Carlos III de Madrid.

IU, MN are with the University of Craiova, Romania (email: urziceanuionut@yahoo.com)

To analyze the movement, we consider the kinematic model of a differential drive robot with a castor wheel [7]:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{R}{2}(\dot{\varphi}_1 + \dot{\varphi}_2) \cos\left(\frac{R}{L}(\varphi_1 - \varphi_2)\right) \\ \frac{R}{2}(\dot{\varphi}_1 + \dot{\varphi}_2) \sin\left(\frac{R}{L}(\varphi_1 - \varphi_2)\right) \\ \frac{R}{L}(\dot{\varphi}_1 - \dot{\varphi}_2) \end{bmatrix} \quad (1)$$

where terms x , y and θ are the robot coordinates and orientation relative to the global reference frame, φ_1 , φ_2 are wheel angles and, in our case, control input signals, R and L correspond to real robot parameters: R is the wheel radius and L is the distance between the two wheels (base width). The locomotion restriction consists in that each wheel can only perform an oscillatory movement between the above given angle values. By coordinating the oscillatory movements of the wheels, three possible movements can be found:

1. If the wheels oscillate in phase (phase difference is 0°), i.e., in the same direction, with the same frequency and at the same time, the robot will move forward and backward repeatedly.
2. If the wheels oscillate in anti-phase (phase difference is 180°), i.e. in opposite directions, with the same frequency and at the same time, the robot will pivot left and right around a vertical axis (Instantaneous Center of Rotation).
3. If the wheels oscillate with a phase difference between 0 and 180° , the movement will be a combination of four movements: turning to one side, go forward, turn to the other side, go backward. We call “step” the length in a straight line of the robot’s trajectory during a sequence of these four movements. It has been shown that the largest step value is obtained for a phase difference of 90° [7].

Other parameters that describe the motion are:

1. Initial phase (φ_0): determines the initial robot orientation relative to the path. It has no effect on the locomotion when the robot is travelling along its path.
2. Amplitude: the rotation angle interval for each wheel, it has an effect on the step size.
3. Offset (O): positive or negative angular value that’s added to the initial wheel angle such that the wheel’s resting position is not φ , but $\varphi + O$. The maximum rotation angle is $\varphi_{\max} - |O|$.
4. Direction (Γ) is the angle measured between the initial robot orientation and a new orientation, after changing the offset values. This depends on the physical robot parameters, wheel radius and base width. Let R and L be the wheel’s radius and robot’s base width, respectively. From the equations of the robot’s kinematic model, the orientation angle can be calculated as:

$$\Gamma = \frac{R}{L} \cdot O \quad (2)$$

A full description of the locomotion principle can be found in [7]. A video demonstration of this principle can be found at <http://goo.gl/2qv6V>.

III. CENTRAL PATTERN GENERATOR

Given the locomotion principle, the first step in designing a central pattern generator is to select the building blocks and the topology that are to be used for the neural circuit. Moreover, one must specify the means in which the information is encoded in the activity of the CPG. Three elements are needed in order to build a complete description: neurons, synapses and motoneurons to drive the servos [8].

A. Neuron model

We use a neuron model developed by Rulkov et al. [9] that mimics the activity of living bursting neurons. The model is computationally efficient and the possible set of behaviors can be controlled depending on the selection of a few parameters [8]. Three stable regimes may be selected by combination of its parameters: silent, in which the potential of the neuron remains in a constant resting state; tonic spiking, in which the neuron produces spikes at a constant rate, and tonic bursting, in which bursts of spikes are produced at a constant rate, with a silent interval in between. Furthermore, in the boundaries of the parametric regions of those regimes, chaotic behavior may be found [10]. For this study we will set the parameters of the neurons to work in tonic bursting regime. See figure 2 for an overall idea of the model working in this regime.

The bursting regime of the model presents a slow wave (slow time scale) with fast spikes of activity sitting on top of it (fast time scale). We use the slow time scale (y_n in (3)) to encode movement duration, i.e., the temporal length of the burst defines the temporal length of the movement, and the fast time scale (x_n in (3)) to define angular velocity of the servo (higher frequency of the spikes corresponds to faster servo movements). The mathematical description of Rulkov’s model as used in this work is as follows:

$$\begin{cases} x_{n+1} = f(x_n, y_n + \beta_e \cdot I_n) \\ y_{n+1} = y_n - \mu \cdot (x_n + 1) + \mu \cdot \sigma + \mu \cdot \sigma_e \cdot I_n \end{cases}$$

$$f(x_n, y_n) = \begin{cases} \frac{\alpha}{1-x_n} + y_n, & x_n < 0 \\ \alpha + y_n, & 0 \leq x_n < \alpha + y_n \\ -1, & \text{otherwise} \end{cases} \quad (3)$$

with $\mu = 0.001$ in all experiments. This is a bi-dimensional model, where variable x_n represents a neuron’s membrane voltage and y_n is a slow dynamics variable with no direct biological resemblance, but with similar meaning as gating variables in biological models that represent the fraction of open ion-channels in the cell. While x_n oscillates on a fast time scale, representing individual spikes of the neuron, y_n keeps track of the bursting cycle, a sort of context memory. Units are dimensionless, and can be rescaled to match the requirements of the robot. The combination of σ and α selects the working regime of the model: silent, tonic spiking or tonic bursting.

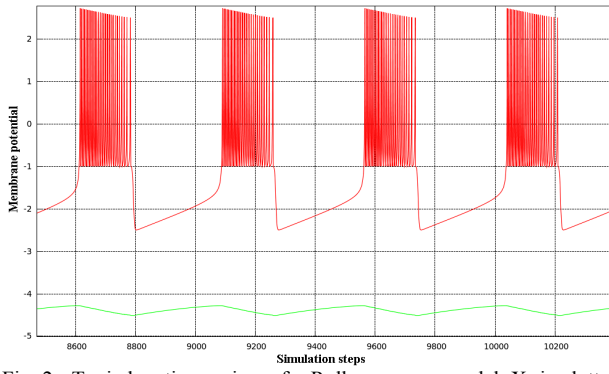


Fig. 2. Tonic bursting regime of a Rulkov neuron model. X_n is plotted in red, representing the fast subsystem; y_n is plotted in green, representing the slow subsystem. Parameters used in this simulation: $\alpha = 7$, $\beta_e = 0$, $\mu = 0.001$, $\sigma = -0.1$, $\sigma_e = 1$.

In the bursting regime, these parameters also control several properties of neural activity. Finally, external input is modeled through I_n . Depending on this value, a neuron will modify its behavior. For instance, an external stimulus, e.g. a sensory signal, may be input using this parameter. This property is essential for autonomous organization: processing units in the CPG must be able to negotiate the rhythm among them. Also, entrainment between the CPG and the physical robot can be achieved through I_n by adding an error term as external input to a neuron [11]. The total effect of this parameter will depend upon past history of events, the exact value of I_n and the phase within the burst cycle at which the neuron finds itself. In our work, I_n is the current flowing from one neuron to another: a periodic sampling of the continuous function described below in (5).

B. Kinetic synapse model

A key property of CPGs is that they are autonomous and the different units in the circuit talk to each other to negotiate the overall function. Here we present the model we have chosen to implement synapses, the communication channel of neurons. In this work we use a chemical synapse model [12]. Chemical synapses are unidirectional. When a potential spike arrives from the presynaptic neuron, the synapse releases a certain amount of neurotransmitter molecules that bind to the postsynaptic neuron's receptors. With time, neurotransmitter molecules begin to unbind. If a succession of spikes arrives within a short time, the synaptic response to each of them may overlap. Therefore the state of the synapse is dependent upon past events, a mechanism of context memory [8]. The additional time-scale provided by kinetic synapses in a CPG enriches synchronization between bursting neurons. For instance, we may choose to synchronize two bursting neurons upon the spike (fast) time scale or the burst (slow) time scale. We have selected the kinetics of the binding and unbinding processes such that synapses act as filters of the fast time scale and synchronization occurs at the slow time scale. That is, the basic unit of synchronization will be the burst as a whole, not every individual spike. Beyond this, synapses may introduce delays for finer control of phase difference between neurons. The mathematical description of the model follows:

$$\dot{r} = \begin{cases} \lambda[T] \cdot (1-r) - \beta \cdot r, & t_f < t < t_f + t_r \\ -\beta \cdot r, & \text{otherwise} \end{cases} \quad (4)$$

This equation defines the ratio of bound chemical receptors in the postsynaptic neuron, where r is the fraction of bound receptors, and are the forward and backward rate constants for transmitter binding and $[T]$ is neurotransmitter concentration. The equation is defined piecewise, depending on the specific times when the presynaptic neuron fires (t_f): during t_r units of time, the synapse is considered to be releasing neurotransmitters that bind to the postsynaptic neuron. After the release period, no more neurotransmitter is released and the only active process is that of unbinding, as described by the second part of the equation. Times t_f are determined as the times when the presynaptic neuron's membrane potential crosses a given threshold. Synaptic current is then calculated as follows:

$$I(t) = g_{syn} \cdot r(t) \cdot (x_{post}(t) - E_{syn}) \quad (5)$$

where $I(t)$ is postsynaptic current at time t , g_{syn} is synaptic conductance, $r(t)$ is the fraction of bound receptors at time t , $x_{post}(t)$ is the postsynaptic neuron's membrane potential and E_{syn} its reversal potential, the potential at which the net ionic flow through the membrane is zero. When coupling two Rulkov map neurons we will need to use a discrete synaptic function. We will build a sequence, let us call it I_n , by simulating $I(t)$ as a continuous function and then taking samples every 0.001 time units. We say that a synapse is excitatory when the probability of the postsynaptic neuron firing a spike increases after the presynaptic neuron has fired. If the probability decreases, the synapse is inhibitory. If the postsynaptic neuron rhythmically emits spikes, an excitatory synapse will generally increase its frequency while an inhibitory one will generally decrease it.

C. Motoneurons

Movement information is robustly encoded in the neurons' bursting episodes. A neuron called motoneuron is then responsible of decoding this information and translating it into the signal that will finally be sent to the servo controller. This signal tells the angle at which the servo should be positioned, in degrees. With a slightly modified version of the motoneuron model provided in [8], we have tried to mimic the real transformation occurring between living motoneurons and muscles. Motoneurons read the activity of other neurons using a simple threshold function that equals 1 if the membrane potentials of their corresponding neurons exceed the threshold values; otherwise the function equals 0. The role of this function is to detect individual spikes of neurons. By setting the threshold to, for example, $v = -1.5$ a.u., this function applied to the potential trace of one neuron will have value 1 during individual spikes and 0 otherwise. In this way, communication between neurons is event-based. That is, the actual shape of neural activity is not so important, only their timing is. This can be a mechanism that the

nervous system employs to lower the impact of noise [1], [13].

The role of motoneurons is now to integrate the individual events emitted by each one of the neurons. If a neuron emits a spike, the motoneuron will move the servo a little bit in a positive or a negative angle, depending on the promotor or remotor effect that is intended (correspondently). If it emits a second spike close enough to the first one, the servo will be positioned a little bit further. If the neurons that are connected to the motoneuron are silent, the motoneuron will slowly drive the servo to a resting position of angle 0. This is accomplished through the following equation governing motoneurons in our CPG:

$$\dot{\phi}_{1,2} = \dot{m}_{1,2} = C(t) - m_{1,2}(t) + O \quad (6)$$

where $m_{1,2}(t)$ represents the output signal of the motoneurons, O is an offset value and $C(t)$ is a function described by:

$$C(t) = \gamma \cdot (a_1 \cdot s(t, v) + a_2 \cdot s(t, v) + \dots + a_n \cdot s(t, v)) \quad (7)$$

where γ is the amplitude of the signal, $s(t, v)$ is the threshold function described above and terms $a_1, a_2, \dots, a_n$ describe the effect of the neurons over the motoneuron: if $a_i = 1$, neuron i has a promotor effect on the corresponding motoneuron, if $a_i = -1$, neuron i will have a remotor effect on the corresponding motoneuron. Given the characteristics for this type of locomotion, the motion can be controlled by means of three parameters: amplitude - to modify the step, offset - to modify the direction of movement and frequency - to modify the speed. Because the locomotion is composed of coupled oscillatory movements, it is possible to design a CPG that can generate the appropriate motor commands to drive the robot.

The motion is split in four sequences:

1. The wheels start by rotating in opposite directions with the same speed, offset and amplitude;
2. Both wheels rotate forward;
3. Both wheels rotate in opposite directions, but contrary to sequence 1;
4. Both wheels rotate backward.

The next design step is to identify what information needs to be encoded in the activity of the CPG. After that, a dynamical invariant must be defined, i.e. a stable, reproducible, repeatable neural pattern that is based on an activation sequence. As mentioned above, offsets, amplitude and speed of movement must be encoded. Also, a constant phase difference between 0 and 180° must be maintained in order to achieve a steady locomotion.

The proposed CPG model comprised of a group of four neurons is considered (see figure 3). To drive the wheels, we need two motoneurons. The main idea of the model is that all four neurons will trigger in a specific non-overlapping sequence. The dynamical properties of neurons and synapses together with the topology of the circuit produces a coordinated alternating rhythm. Each of them will have a promotor or remotor effect on the motoneurons in such way

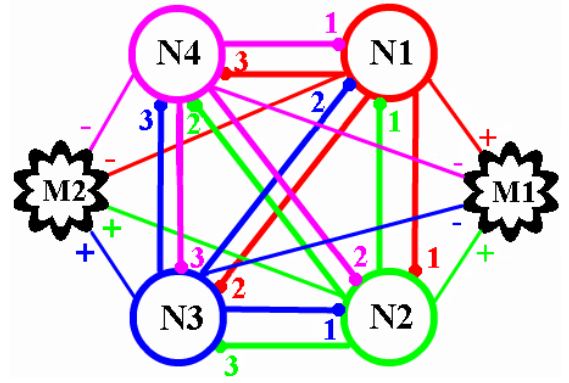


Fig. 3. CPG topology: the neural circuit is made of four neurons: N1, N2, N3, N4, two motoneurons: M1, M2 and twelve chemical synapses, each of them being identified by color and number. The arithmetic signs indicate the promotor or remotor effects that neurons have on the motoneurons. For forward locomotion, synapses red-1, green-3, blue-3, magenta-1 must have a smaller coupling strength compared to the others (See figure 5 for specific values).

that the motion sequences are obtained. The motoneurons are connected to all neurons of the model. Finally, the CPG will self-organize so that at any moment, each motoneuron will receive a signal from only one neuron to drive the wheels.

The angle values for the wheels will be encoded in the spiking activity of each neuron. Each spike represents a small increment in the wheel angle. The motoneurons then integrate the spikes and output the actual angle value, which is represented by $m(t)$ in (6). Therefore, amplitude can be controlled by changing the number of spikes per burst period. The larger the number, the greater the wheel angle.

Amplitude can also be tweaked by changing the γ parameter in (7). The speed of the movement is controlled by altering the burst periods. A longer burst will translate into a slower evolution of the motoneuron signal.

Offsets are also encoded in the motoneurons, using the “ O ” parameter in (6), such that the starting angle values are not zero, but O . By adjusting the offset values, one can change the trajectory of the robot. It is worth noting that in all cases, the robot will move in a straight line, regardless of the values of the offsets and the amplitudes. By changing the offset values, we change only the robot orientation.

Finally, the most important parameter encoded in the CPG is the phase difference that establishes the coordination between the wheels. In a more intuitive approach, phase difference is related with the time delay between the moment at which one wheel starts moving and the moment at which the second wheel starts moving. By measuring the angle difference between the wheels in this time interval, we can obtain the phase difference. In this CPG model, the desired phase difference is achieved through the activation sequence of the neurons which, in term, is obtained using inhibition and asymmetric coupling. Inhibition is closely related to the synaptic conductance (g_{syn} in (5)).

Depending on the sign of the synaptic current, the slow subsystem (see equation (3)) is either increased (case of an excitatory synapse), causing a burst, or decreased (case of an inhibitory synapse), triggering an early resting state of the neuron.

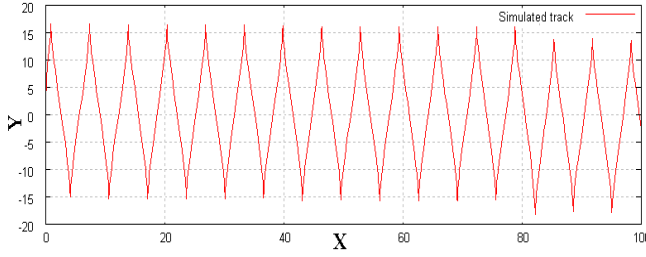


Fig. 4. Simulated trajectory of the differential drive using motoneuron-like motor signals (sine signals) that are phase-shifted by 90° . The simulation has been carried in Matlab/Simulink, with a signal amplitude of 90° . Wheel radius is 55 mm, base width is 103 mm.

For a clearer example of how phase difference is obtained, let us consider this activation sequence: $N1 \rightarrow N2 \rightarrow N3 \rightarrow N4$. In figure 3, the network is completely connected, i.e., each neuron is connected to all others. Also, each neuron is connected to both motoneurons and can simultaneously drive them. If all synapses have equal coupling strengths, the neurons will fire depending on which of them gets the chance to fire first. What we can do is encourage some neurons to fire before others in order to obtain the desired sequence and we do this by adjusting coupling strengths. For the above chain of activation, synapses from the forward path (red-1, green-3, blue-3, magenta-1 in figure 3) must have a weaker inhibitory effect than all other synapses. Correspondingly, neuron $N2$ will be favored to burst before others ($N1$, $N4$), then $N3$ and so on. Therefore, we will end up with synapses that have a powerful inhibitory effect and synapses that have a weaker inhibitory effect. If all neurons share the same parameters (except for initial starting conditions), all strong synapses share a certain coupling strength and all weak ones share another coupling strength, the period of the whole CPG will be composed of four equal regions in which each one is a burst period of a corresponding neuron.

As it can be seen in figure 5, during the activation period of $N1$, the motoneuron will be driven in order to increase the output signal (wheel angle), whereas an activation of $N3$ will decrease the motoneuron's output signal. Considering promotor and remotor neuron effects on motoneurons and given the fact that a complete wheel oscillation takes one CPG period and the delay between wheel movements are one burst region wide, a phase difference of 90° between the wheels is obtained. More intuitively, when one wheel has reached a quarter of its oscillation amplitude, the second wheel will begin to rotate. With this model, the phase difference cannot be changed due to the non-overlapping activation sequence; it is fixed to approximately 90° .

To change the direction of movement between moving forward and moving backwards, the activation sequence is changed by changing coupling strengths. For moving forward, the sequence is $N1 \rightarrow N2 \rightarrow N3 \rightarrow N4$. For moving backward, the sequence is $N1 \rightarrow N4 \rightarrow N3 \rightarrow N2$. To steer the robot, the wheel offsets and the amplitude have to be changed, so that $0 + A \leq |\phi_{\max}|$, where $\phi_{\max}$ is the maximum rotation angle (clockwise or anti-clockwise) that the joint can achieve.

IV. DRIVING THE ROBOT

Due to the relatively modest hardware resources of the PIC16F876A, the CPG software model has been implemented on a personal computer. The simulation of the CPG is done on this computer and the outputs (i.e. angle values of the wheels) are sent via serial interface to the robot which commands the two servos. No real entrainment has been implemented between the CPG and the actuators. In an ideal case, the CPG would wait for the actuators to position to the specified positions; in our case, due to the fact that we did not have feedback from the actuators (we used standard hobby servos), the CPG simulation is halted; by taking into consideration the number of iterations between two motor commands that are sent to the robot and the type of the servo, we estimate the servo response time (let us call it t_s). Therefore, the computer will wait t_s units of time before sending a new motor command to the robot.

A simplified control interface has been implemented to drive the robot forward, backwards, left and right and stop it. Forward and backward locomotion are obtained just by changing the coupling strengths of particular synapses which changes the activation sequence of the neurons as described above. Left and right locomotion is achieved by altering the offset values of the motoneurons according to (7). Due to the fact that the offset and the amplitude of the movement are related, such that $\phi_{\max} \leq \gamma + O$, one must adjust the amplitude value. The main control program that is running on the PC receives commands from the user and changes the CPG parameters accordingly.

A first approximation of the robot trajectory, using the above described locomotion principle has been done by implementing an idealized kinematic model of a differential drive robot. By numerically integrating (1), we can obtain the robot positions and orientation relative to a global reference frame (see figure 4). Figure 5.C illustrates the actual trajectory of the robot that has been extracted from the video tracking of the robot performing a forward locomotion. A video demonstration of the robot controlled by the proposed central pattern generator is available at <http://youtu.be/WhfcUNx3IQY>.

V. CONCLUSION

In this paper we have shown how a differential wheeled robot with limited wheels can be controlled with a bio-inspired CPG. The CPG provides a command signal for the rhythmic control of the limited wheels that overcomes their restriction and produces an effective locomotion. The CPG approach can also account for sensory feedback in the form of additional stimulus to all or some neurons in the circuit. Sensory feedback can also be implemented in the form of a modulation of the CPG connectivity. The wide range of rhythm negotiation properties of the proposed model leads to a rich variety of self-organized locomotion.

Advances in CPG research in recent years provide new bio-inspiration for robotic design. We believe that novel CPG control strategies can lead to more autonomous robot locomotion.

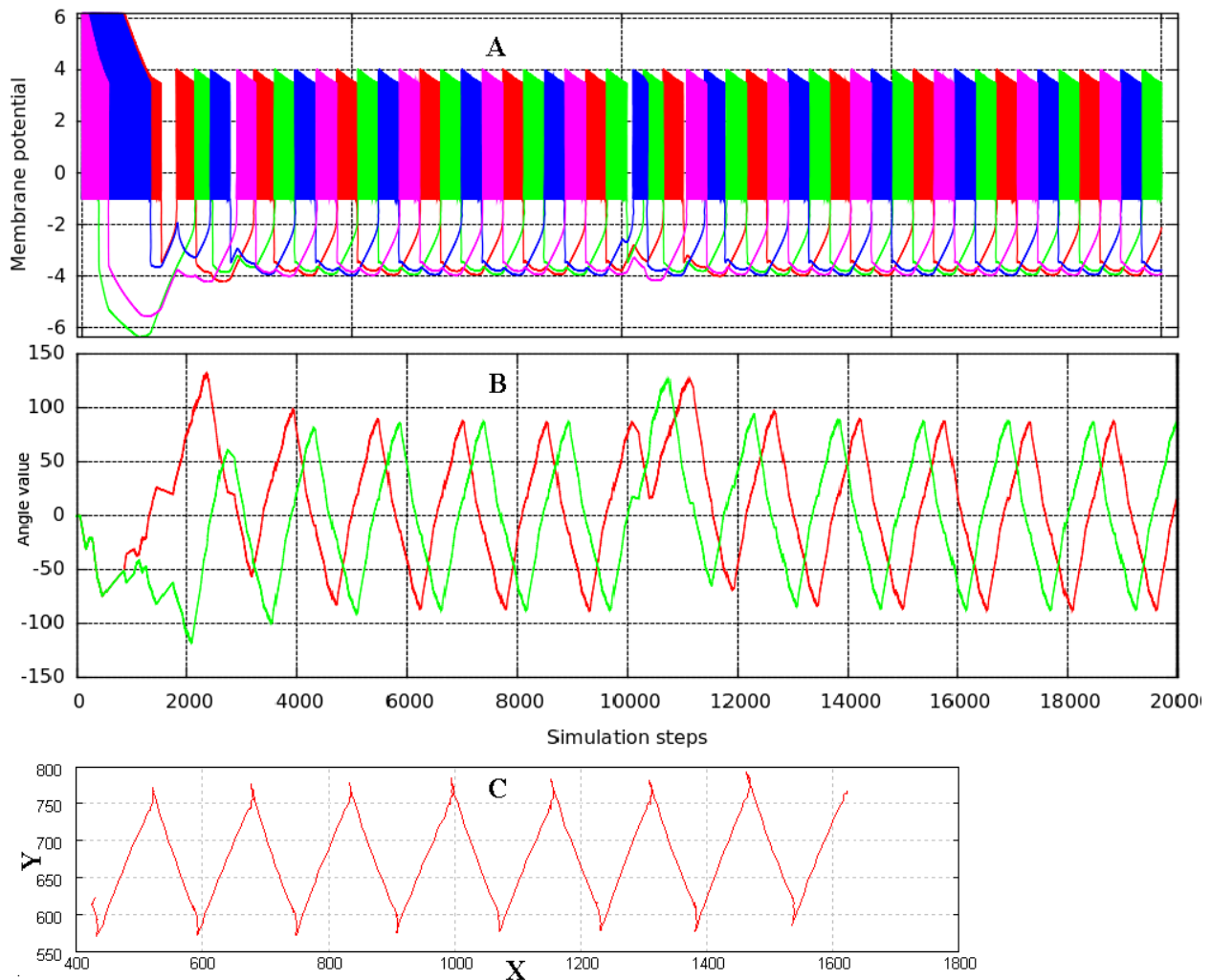


Fig. 5. A: Plot representing the activity of the CPG neurons (activation sequence). 20000 iterations have been simulated: 10000 representing a forward movement, 10000 representing a backward movement. B: the motoneurons output values (wheel angles). After an initial unstable transient period of negotiation, the CPG autonomously achieves a stable oscillation, with constant frequency and amplitude. C: Real robot trajectory extracted from video tracking of a forward movement (arbitrary pixel coordinates are used for Y axis, and number of frames is used for the X axis). Parameters used: neurons: $\alpha = 9$, $\sigma = 0.5$, $\sigma_c = 1$; synapses: $\alpha = 0.5$, $\beta =$, $E_{syn} = 9$, $g_{syn1,2} = 25$, $T = 1$, $release_time = 0.01$; motoneurons: $\gamma = 900$, $v = -1.5$, $O = 0$.

REFERENCES

- [1] M. Rabinovich, P. Varona, A. Selverston, and H. Abarbanel, "Dynamical principles in neuroscience". *Reviews of Modern Physics*, 78(4), 1213-1265, 2006. APS. doi: 10.1103/RevModPhys.78.1213.
- [2] A. J. Ijspeert. Central pattern generators for locomotion control in animals and robots: a review. *Neural Networks*, 21(4), pp. 642-653. Elsevier, 2008.
- [3] M. Năulescu, Theoretical aspects in wheeled mobile robot control, Proc. of International Conference on Automation, Quality and Testing, Robotics –AQTR 2008, Cluj-Napoca, Romania, ISBN 978-1-4244-2576-1.
- [4] WikiRobotics: Skybot repository. Available at next web address: <http://www.learobotics.com/wiki/index.php?title=Skybot>
- [5] WikiRobotics: SkyPic repository. Available at next web address: <http://www.learobotics.com/wiki/index.php?title=Skypic>
- [6] WikiRobotics: Sky293 repository. Available at next web address: <http://www.learobotics.com/proyectos/sky293/sky293.html>
- [7] J. Gonzalez-Gomez, J. G. Victores, A. Valero-Gomez, M. Abderrahim, "Motion Control of Differential Wheeled Robots with Joint Limit Constraints", in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2011, (to appear).
- [8] F. Herrero-Carrón, F. B. Rodríguez, and P. Varona (2011). Bio-inspired design strategies for central pattern generator control in modular robotics. *Bioinspir Biomim*, 6(1), 16006. doi: 10.1088/1748-3182/6/1/016006.
- [9] N. F. Rulkov, "Modeling of spiking-bursting neural behavior using two-dimensional map", *Physical Review*, vol. 65, 2002, pp. 1-9.
- [10] A. L. Shilnikov and N. F. Rulkov, "Origin of chaos in a two-dimensional map modeling spike-bursting neural activity", *Int. J. Bif. and Chaos*, 13:3325–3340, 2003.
- [11] F. Herrero-Carron, F.B. Rodriguez, P.Varona. Flexible entrainment in a bio-inspired modular oscillator for modular robot locomotion. *Lect Notes Comput. Sc* 6692: pp. 532-539.
- [12] A. Destexhe, Z. F. Mainen, and T. J. Sejnowski, "An Efficient Method for Computing Synaptic Conductances Based on a Kinetic Model of Receptor Binding". *Neural Computation*, 6(1), 14-18, 1994, MIT Press. doi: 10.1162/neco.1994.6.1.14.
- [13] F. Herrero-Carrón, F.B. Rodríguez, P. Varona. Studying robustness against noise in oscillators for robot control. *Proceedings of the ARCS Conference*, pp. 58-64, Orlando, 2011.
- [14] J. Gonzalez-Gomez, A. Valero-Gomez, A. Prieto-Moreno, M. Abderrahim, "A New Open Source 3D-printable Mobile Robotic Platform for Education". *Proc. of the 6th Int. Symposium on Autonomous Minirobots for Research and Edutainment*, 2011, May, 23-25. Bielefeld. Germany.

The Comparative Study of Two Analytical Methods for Detection and Elimination of the Static Hazard in Combinational Logic Circuits

Alexandru Valachi, Bogdan I. Aignătoaei, and Mihai G. Timiș

Abstract— In this paper, the authors do a comparative study of two methods to eliminate the static hazards from logical functions. The first method consists in determining the coupled terms and getting the term that masks the static hazards, and the second method consists in determining the masking additional term by solving a system of Boolean equations. From the presented examples, the authors deduced that the 2nd method (solving a Boolean equations system) is more advantageous because it eliminates the heuristic nature of the first method (the determination of coupled terms) and, by solving the equations system it determines all prime implicants that needs to be added to the function's expression.

I. INTRODUCTION

UNDER certain conditions, may occur unwanted transitions in the output logical signals. These transitions are known as glitches. A logic glitch is a kind of unwanted noise, its presence in the output signal can initiate an uncontrollable process, in the next level where that is an input signal [1].

We can distinguish three types of noise that is introduced in CLC (Combinational Logic Circuits), called hazards (Static, Dynamic and Function Hazards).

In the following we consider only the static hazard problem:

- Static 1 hazard, also called SOP (Sum of Products) hazard – a glitch that occurs in otherwise steady-state 1 output signal from SOP logic;

- Static 0 hazard, also called POS (Product of Sums) hazard – a glitch that occurs in an otherwise steady-state 0 output signal from POS logic.

II. STATIC HAZARDS IN TWO-LEVEL COMBINATIONAL LOGIC CIRCUITS (CONSENSUS METHOD [1])

We will initially define:

- Coupled variable; a variable input is complemented within a term of function and uncomplemented in another term of the same function.
- Coupled term; one of two terms containing only one coupled variable.
- Residue; the part of a coupled term that remains after removing the coupled variable.

- Hazard cover (or consensus term).

The RPI (Redundant Prime Implicant) required to eliminate the static hazards:

- AND the residues of coupled p-term to obtain the SOP hazard cover,
- OR the residues of coupled s-term to obtain the POS hazard cover.

Example 1. Lets consider the logic function $f(x_2x_1x_0) = R_1(2,3,5,7)$.

a) *SOP example:* will be determined the prime implicants using Veitch-Karnaugh or Quine-McCluskey methods, as:

$$\begin{aligned} A &= \overline{x_2} \cdot x_1 & (2,3) \\ B &= x_1 \cdot x_0 & (3,7) \\ C &= x_2 \cdot x_0 & (5,7) \end{aligned} \quad (1)$$

One of the minimal equations is:

$$y = A + C = \overline{x_2} \cdot x_1 + x_2 \cdot x_0 \quad (2)$$

where we have:

- coupled variable: x_2
- coupled terms: $\overline{x_2} \cdot x_1, x_2 \cdot x_0$
- residues: x_1, x_0
- consensus term: $x_1 \cdot x_0$

Therefore, the logic expression that has no static hazard in relation to x_2 variable is:

$$y = \overline{x_2} \cdot x_1 + x_2 \cdot x_0 + x_1 \cdot x_0 \quad (3)$$

b) *POS example:* will be determined the prime implicants using Veitch-Karnaugh or Quine-McCluskey methods, as:

$$\begin{aligned} a &= x_2 + x_1 & (0,1) \\ b &= x_1 + x_0 & (0,4) \\ c &= \overline{x_2} + x_0 & (4,6) \end{aligned} \quad (4)$$

Al. Valachi, B. I. Aignătoaei, and M. G. Timiș are with the Automatic Control and Computer Engineering Faculty of the Technical University "Gheorghe Asachi" of Iași, Bd. Mangeron 27, 700050 Iași, Romania (e-mails: avalachi@cs.tuiasi.ro, baignat@cs.tuiasi.ro, mtimis@cs.tuiasi.ro).

One of the minimal equations is:

$$y = (x_2 + x_1) \cdot (\overline{x_2} + x_0) \cdot (x_1 + x_0) \quad (5)$$

where we have:

- coupled variable: x_2
- coupled terms: $\overline{x_2} + x_0, x_2 + x_1$
- residues: x_1, x_0
- consensus term: $x_1 + x_0$

The equation (5) shows no static 0 hazard.

Example 2. Let's consider the function of four variables $y = f(x_3, x_2, x_1, x_0) = R_1(0, 1, 2, 5, 6, 7, 8, 9, 10, 14)$.

a) *SOP hazard*: will be determined the prime implicants using Quine-McCluskey method, as:

$$\begin{aligned} A &= \overline{x_3} \cdot \overline{x_1} \cdot x_0 & (1, 5) \\ B &= \overline{x_3} \cdot x_1 \cdot \overline{x_0} & (2, 6) \\ C &= \overline{x_3} \cdot x_2 \cdot x_0 & (5, 7) \\ D &= \overline{x_3} \cdot x_2 \cdot x_1 & (6, 7) \\ E &= x_3 \cdot x_1 \cdot \overline{x_0} & (10, 14) \\ F &= \overline{x_2} \cdot \overline{x_1} & (0, 1, 8, 9) \\ G &= \overline{x_2} \cdot x_0 & (0, 2, 8, 10) \\ H &= x_1 \cdot \overline{x_0} & (2, 6, 10, 14) \end{aligned} \quad (6)$$

Applying the Patrick method [2], going from prime implicants table will be determined all SOP solutions.

Let's consider the logical p_i variables attached to the prime implicants as follows: $p_0 \leftrightarrow A$, if $p_0 = 1$, the A prime implicant is present in the logical function expression, otherwise $p_0 = 0$ (A prime implicant is not present in the logical function expression), etc.

Therefore, considering the correspondence $p_1 \leftrightarrow B$, $p_2 \leftrightarrow C$, $p_3 \leftrightarrow D$, $p_4 \leftrightarrow E$, $p_5 \leftrightarrow F$, $p_6 \leftrightarrow G$, $p_7 \leftrightarrow H$, in the table illustrated in Figure 1 is shown the Patrick coverage table:

<i>dec. equiv.</i>	0	1	2	5	6	7	8	9	10	14
p_i										
p_0		1		1						
p_1			1		1					
p_2				1		1				
p_3					1	1				
p_4									1	1
p_5	1	1					1	1		
p_6	1		1				1		1	
p_7			1		1				1	1

Fig. 1. The SOP coverage table

It writes the coverage equation:

$$\begin{aligned} &(p_5 + p_6) \cdot (p_0 + p_5) \cdot (p_1 + p_6 + p_7) \cdot \\ &\cdot (p_0 + p_2) \cdot (p_1 + p_3 + p_7) \cdot (p_2 + p_3) \cdot \\ &\cdot (p_5 + p_6) \cdot p_5 \cdot (p_4 + p_6 + p_7) \cdot (p_4 + p_7) \equiv 1 \end{aligned} \quad (7)$$

Simplifications are made by using the laws of Boolean algebra: the redundancy law, the identity law and the distributive law.

$$\begin{aligned} &p_5 \cdot (p_1 + p_6 + p_7) \cdot (p_1 + p_3 + p_7) \cdot \\ &\cdot (p_4 + p_7) \cdot (p_2 + p_0) \cdot (p_2 + p_3) = 1 \quad \text{or} \\ &p_5 \cdot (p_7 + p_1 + p_3 \cdot p_6) \cdot (p_7 + p_4) \cdot \\ &\cdot (p_2 + p_0 \cdot p_3) = 1 \quad \text{or} \quad (8) \\ &p_5 \cdot (p_7 + p_1 \cdot p_4 + p_3 \cdot p_4 \cdot p_6) \cdot \\ &\cdot (p_2 + p_0 \cdot p_3) = 1 \quad \text{or} \\ &(p_5 \cdot p_7 + p_1 \cdot p_4 \cdot p_5 + p_3 \cdot p_4 \cdot p_5 \cdot p_6) \cdot \\ &\cdot (p_2 + p_0 \cdot p_3) = 1 \end{aligned}$$

A version of the optimal solution corresponds to $p_5 \cdot p_7 \cdot p_2$ triplet, i.e.

$$\begin{aligned} y &= f(x_3, x_2, x_1, x_0) = F + H + C = \\ &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0 \end{aligned} \quad (9)$$

The cost of this function in SOP implementation is:

$$C(y) = C(\overline{x_2} \cdot \overline{x_1}) + C(x_1 \cdot \overline{x_0}) + C(\overline{x_3} \cdot x_2 \cdot x_0) + 3 = 10$$

(it was considered the variables $x_i, \overline{x_i}$, available at input).

It can verify that any other coverage has a higher cost.

For example, the coverage $p_5 \cdot p_7 \cdot p_0 \cdot p_3$ which corresponds to

$$\begin{aligned} y &= F + H + A + D = \\ &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot \overline{x_1} \cdot x_0 + \overline{x_3} \cdot x_2 \cdot x_1 \end{aligned} \quad (10)$$

has the cost $C_1(y) = 14$.

III. THE STATIC HAZARD DETECTION AND ELIMINATION

In this paragraph we apply the consensus method [1] and the method of solving some specific Boolean equations.

A. The consensus method

We apply the same method as in [1], only that it has a strong computing nature. Any logic function can be written as:

$$y = e_0 \cdot \overline{x_i} + e_1 \cdot x_i \quad (11), \text{ where}$$

$$e_0 = y(x_{n-1}, x_{n-2}, \dots, x_{i+1}, 0, x_{i-1}, \dots, x_0),$$

$$e_1 = y(x_{n-1}, x_{n-2}, \dots, x_{i+1}, 1, x_{i-1}, \dots, x_0).$$

Obviously, if $y = a \cdot \overline{x_i} + b \cdot x_i + c \quad (12)$, then

$$e_0 = a + c, \quad e_1 = b + c.$$

If we add the term $e_0 \cdot e_1$ to relation (11), the function presents no hazard towards x_i .

In terms of the consensus method, the term that covers the static 1 hazard is

$$e_0 \cdot e_1 = (a + c) \cdot (b + c) = c + a \cdot b \quad (13), \text{ therefore}$$

for the form (11) will be $e_0 \cdot e_1$, and for the form (12), $a \cdot b$.

Considering the second example, we will have:

a) hazard in relation to the input x_0 :

$$y = F + H + C = \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0 \quad (14),$$

where $e_0 = x_1 + \overline{x_2} \cdot \overline{x_1} = x_1 + \overline{x_2}$,

$$e_1 = \overline{x_3} \cdot x_2 + \overline{x_2} \cdot \overline{x_1} = \overline{x_3} \cdot x_2 + \overline{x_2} \cdot \overline{x_1},$$

$$\begin{aligned} e_0 \cdot e_1 &= (x_1 + \overline{x_2}) \cdot (\overline{x_3} \cdot x_2 + \overline{x_2} \cdot \overline{x_1}) = \\ &= \overline{x_3} \cdot x_2 \cdot x_1 + \overline{x_2} \cdot \overline{x_1} = F + D \end{aligned} \quad (15)$$

By adding $F + D$ term to relation (14), it obtains:

$$\begin{aligned} y &= F + H + C + (F + D) = \\ &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0 + \overline{x_3} \cdot x_2 \cdot x_1 \end{aligned} \quad (16)$$

b) hazard in relation to the input x_1 :

$$e_0 = \overline{x_2} + \overline{x_3} \cdot x_2 \cdot x_0 = \overline{x_2} + \overline{x_3} \cdot x_0$$

$$e_1 = x_0 + \overline{x_3} \cdot x_2 + \overline{x_3} \cdot x_2 \cdot x_0 = x_0 + \overline{x_3} \cdot x_2 \quad (17)$$

$$e_0 \cdot e_1 = (\overline{x_2} + \overline{x_3} \cdot x_0) \cdot (x_0 + \overline{x_3} \cdot x_2) = \overline{x_2} \cdot x_0 = G$$

Therefore, the expression of the function becomes:

$$\begin{aligned} y &= F + H + C + D + G = \\ &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0 + \\ &+ \overline{x_3} \cdot x_2 \cdot x_1 + \overline{x_2} \cdot x_0 \end{aligned} \quad (18)$$

c) hazard in relation to the input x_2 :

$$e_0 = \overline{x_1} + \overline{x_0} + x_1 \cdot \overline{x_0} = \overline{x_1} + \overline{x_0}$$

$$e_1 = \overline{x_3} \cdot x_0 + \overline{x_3} \cdot x_1 + x_1 \cdot \overline{x_0}$$

Therefore,

$$\begin{aligned} e_0 \cdot e_1 &= (\overline{x_1} + \overline{x_0}) \cdot (\overline{x_3} \cdot x_0 + \overline{x_3} \cdot x_1 + x_1 \cdot \overline{x_0}) = \\ &= \overline{x_3} \cdot x_1 \cdot x_0 + \overline{x_3} \cdot x_1 \cdot \overline{x_0} + x_1 \cdot \overline{x_0} = \\ &= x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_1 \cdot x_0 = H + A \end{aligned} \quad (19)$$

The expression of the function becomes:

$$\begin{aligned} y &= F + H + C + D + G + A = \\ &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0 + \\ &+ \overline{x_3} \cdot x_2 \cdot \overline{x_1} + \overline{x_2} \cdot x_0 + \overline{x_3} \cdot x_1 \cdot x_0 = \\ &= \overline{x_3} \cdot x_2 \cdot x_0 + \overline{x_3} \cdot x_2 \cdot \overline{x_1} + x_1 \cdot x_0 + t \end{aligned} \quad (20)$$

d) hazard in relation to the input x_3 :

$$e_0 = x_2 \cdot x_0 + x_2 \cdot \overline{x_1} + x_1 \cdot x_0 + t, \text{ where}$$

$$t = \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_2} \cdot x_0 + \overline{x_3} \cdot x_1 \cdot x_0$$

$$e_1 = t$$

$$e_0 \cdot e_1 = (x_2 \cdot x_0 + x_2 \cdot \overline{x_1} + x_1 \cdot x_0 + t) \cdot t = t \quad (21), \text{ so}$$

that remains the same expression (20), which has no hazards in relation to x_3 .

From the relation (20), it sees that the expression of the function without SOP hazards contains all prime implicants without B and E.

B. The method of solving of some Boolean equations [3]

A logic function can be written as:

$$y = a \cdot \overline{x_i} + b \cdot x_i + c \quad (22), \text{ where}$$

$$a + c = f(x_{n-1}, x_{n-2}, \dots, x_{i+1}, 0, x_{i-1}, \dots, x_0),$$

$$b + c = f(x_{n-1}, x_{n-2}, \dots, x_{i+1}, 1, x_{i-1}, \dots, x_0).$$

According to a theorem from [3], a logic function expressed as SOP, presents a static hazard in the situation deducted by solving the following system of logical equations:

$$\begin{aligned} a &= 1 \\ b &= 1 \\ c &= 0 \end{aligned} \quad (23)$$

We return to the same function, (14):

$$y = F + H + C = \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0.$$

a) hazard in relation to the input x_0 :

$$y = (x_1) \cdot \overline{x_0} + (\overline{x_3} \cdot x_2) \cdot x_0 + \overline{x_2} \cdot \overline{x_1} \quad (24)$$

The function will present SOP hazard, if

$$\begin{aligned} x_1 &= 1 \\ \overline{x_3} \cdot x_2 &= 1 \\ \overline{x_2} \cdot x_1 &= 0 \end{aligned} \quad (25)$$

Therefore, $x_3 = 0$, $x_2 = 1$, $x_1 = 1$, which imposes a hazard at commutation

$$\begin{aligned} x_3 x_2 x_1 x_0 \\ 0110 \leftrightarrow 0111, \end{aligned}$$

which imposes the adding of the prime implicant $D = (6,7)$ to function.

The function becomes

$$\begin{aligned} y &= F + H + C + D = \\ &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0 + \overline{x_3} \cdot x_2 \cdot x_1 \end{aligned} \quad (26)$$

b) hazard in relation to the input x_1 :

$$\begin{aligned} a &= \overline{x_2} = 1 & x_2 &= 0 \\ b &= \overline{x_0} + \overline{x_3} \cdot x_2 = 1 & \text{or} & \quad x_0 = 0 \\ c &= \overline{x_3} \cdot x_2 \cdot x_0 = 0 & x_3 &= \phi \end{aligned} \quad (27)$$

Therefore, we will have hazards in the following cases:

$$\begin{aligned} x_3 x_2 x_1 x_0 \\ 0000 \leftrightarrow 0010 \quad (0,2) \\ 1000 \leftrightarrow 1010 \quad (8,10) \end{aligned}$$

The previous commutations are equivalent to the implicant $G = (0,2,8,10)$.

The function becomes

$$\begin{aligned} y &= F + H + C + D + G = \\ &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0 + \\ &+ \overline{x_3} \cdot x_2 \cdot x_1 + \overline{x_2} \cdot \overline{x_0} \end{aligned} \quad (28)$$

c) hazard in relation to the input x_2 :

$$\begin{aligned} a &= \overline{x_1} + \overline{x_0} = 1 & x_3 &= 0 \\ b &= \overline{x_3} \cdot x_0 + \overline{x_3} \cdot x_1 = 1 & \Rightarrow & \quad \text{and} \\ c &= x_1 \cdot \overline{x_0} = 0 & x_0 + x_1 &= 1 \end{aligned} \quad (29)$$

We will have the solution:

$$\begin{aligned} x_3 &= 0 \\ x_1 &= 0 \\ x_0 &= 1 \end{aligned} \quad (30)$$

The corresponding commutation is:

$$\begin{aligned} x_3 x_2 x_1 x_0 \\ 0001 \leftrightarrow 0101 \quad (1,5). \end{aligned}$$

Therefore, the term $A = (1,5)$ is added to the function.

And therefore: $y = F + H + C + D + G + A$ (30)

d) hazard in relation to the input x_3 :

$$\begin{aligned} y &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_3} \cdot x_2 \cdot x_0 + \\ &+ \overline{x_3} \cdot x_2 \cdot x_1 + \overline{x_2} \cdot \overline{x_0} + \overline{x_3} \cdot x_1 \cdot x_0 \end{aligned} \quad (31)$$

$$\begin{aligned} a &= x_2 \cdot x_0 + x_2 \cdot x_1 + \overline{x_1} \cdot x_0 = 1 \\ b &= 0 \\ c &= \overline{x_2} \cdot \overline{x_1} + x_1 \cdot \overline{x_0} + \overline{x_2} \cdot \overline{x_0} = 0 \end{aligned} \quad (32)$$

Because one of the terms (a,b) is zero, we have no hazards in relation to that variable.

IV. CONCLUSION

The contribution of the authors consists in that by analysis of two methods of detection / elimination of the static hazard – it concludes that the classical method of the '70s, the method of solving some specific Boolean equations [3], presents some advantages compared to consensus method [1], which has a strong heuristic nature.

In the first method it's used the consensus theorem to determine the cover term that is equal with the product of the two residual implicants [1], and in the second method it's resolved a Boolean equation system [3]. The authors observed that in the second method the digital hazard can be earlier detected. If the Boolean equation system is incompatible (doesn't have solutions), the considered logical function doesn't have the static 1 hazard regarding the coupled variable. Using the logical computations, this method permits to determine the needed transitions to eliminate the digital hazard.

From the both methods can be observed that static 1 hazard can be removed by adding the prime implicants step by step.

The same methods with the same conclusions can be applied to the static 0 hazard (POS), using the duality theorem [2].

REFERENCES

- [1] Richard F. Tinker, "Engineering Digital Design", Academic Press, 2000.
- [2] Ch. Roth, "Fundamentals of Logic Design", West Publishing Company, 1999.
- [3] J.P. Perrin, M. Denouette, E. Daclin, "Systems Logiques", Tome 1 Dunord, Paris, 1967.
- [4] <http://webdocs.cs.ualberta.ca/~amaral/courses/329/web/slides/Topic7-TimingHazards/sld003.htm>
- [5] <http://www.utdallas.edu/~cantrell/ee2310/sl1hazard.pdf>
- [6] http://www.doe.carleton.ca/~jknight/97.267/267_04W/Asch1HazShortJ.pdf

Implementation and Covering Analysis for the T-way Combinatorial Testing Strategy

G. Varvara, V. Sandu

Abstract— The paper stands from the general concepts of combinatorial testing in order to point out the benefits of the automatically generated tests based on the t-way combinatorial techniques. Then it implements an optimal strategy that reduce the necessary number of test cases and analysis its effectiveness on practical examples that involves a reduce number of correlated input parameters. Some remarks on the proposed implementations cost and test covering advantages conclude this work.

I. INTRODUCTION

CURRENT software systems are complex entities with demanding quality specifications. In order to ensure all the functional and non-functional requirements, the testing process becomes more expensive and trade-offs often exist to optimize the use of resources. In the literature several systematic approaches to software testing have been proposed. Category partitioning [1] is the base of all systematic approaches as finite values of parameters are identified for testing. Each of these finite parameter-values may be tested at least once, in specified combinations together, or in exhaustive combination. The simplest approach tests all values at least once. The most thorough approach exhaustively tests all parameter-value combinations. While testing only individual values may not be enough, exhaustive testing of all possible combinations is not always feasible. Combination strategies are a reasonable alternative that falls in between these two extremes.

This work makes a short presentation of the combinatorial testing methodology with focus on the T-way testing characteristics and performance. Then it details the algorithm behind the proposed T-way Generator tool, its implementation decisions and operation aspects. In the second part of the paper we present some experiments to evaluate the testing effort using this tool and its performance versus other open source similar tools.

The paper is organized in a short introduction, four sections and a paragraph devoted to some conclusions

based on the practical experience in working with T-way testing facilities and further development. The first section makes a theoretical general presentation of the combinatorial testing benefits and practical results, the most used approaches with a special space devoted to the T-way strategy. Section III presents the T-way combinatorial efficiency comparing with other testing techniques by means of specific metrics. Section IV is devoted to the presentation of the In-Parameter-Order-General (IPOG) - test set construction algorithm. Section V details the analysis and design process for the proposed tool, the experimental setup organization and the basis for the results interpretation.

II. THE COMBINATORIAL TESTING STRATEGIES

The combinatorial testing can help to detect problems early in the testing life cycle using simple “black box” techniques. It is based on test cases construction using all the possible combinations between the parameters of the system under test. For a reduced number of parameter this is a simple straightforward technique. For example, a system with 4 parameters, 2 with 3 values, 1 having 5 values and one 8 values, has to be tested exhaustively for $3^2 \times 5 \times 8 = 360$ combinations that is a reasonable process with not excessive costs. But, for a panel with 30 switches with 2 positions there will be necessary for a 100% covering test a total of 2^{30} combinations of “on” and “off” settings. This is not practically tractable. The key insight underlying the reduction of the combinatorial sets is that not every parameter contributes to every failure. Only a small number of interacting parameters contribute to most detected failures. Based on that observation, testers often use the strategy named “pairwise testing” that involves the execution of at least one test for all the possible pairs of parameter values. This is a highly effective method and was implemented by a lot of available tools that automatically generate arrays with all pairs of parameter value combinations. But, if the interactions involve more than 2 parameters, new methods for low cost test generation have to be found and new tools that implement these methods have to be provided. There are consistent studies and valuable results referring to this problem known as T-way combinatorial testing. Some observations related to the parameter interaction level, named interaction strength, have to be mentioned [2], [3]: all failures could be triggered by no more than 4-way to 6-way interactions, the detection rate increases rapidly with the

Manuscript received June 18th, 2011.

G. Varvara is with the Technical University “Gheorghe Asachi” from Iasi, Prof.dr. Dimitrie Mangeron Street, 27, 700050, RO, (phone 0040232230751, email: gvarvara@ac.tuiasi.ro)

V. Sandu is with the Technical University “Gheorghe Asachi” from Iasi, Prof.dr. Dimitrie Mangeron Street, 27, 700050, RO, (email: violeta.sandu@gmail.com).

interaction strength, the efficient generation of the test suites to cover all the T-way combinations is a difficult mathematical problem, most parameters are continuous variables with possible values in a very large range that has to be discretized only in a few distinct numerical values. In response to these problems theoretical and practical results have been arisen. They can be grouped in:

- Computational approaches [4], [5] – the test sets are established by adding new tests, one at a time, to a core set, until all the combinations for the parameter values are covered. A greedy algorithm is often used to test generation. The efficiency of the algorithm is strongly dependent to heuristic applied for greedy algorithm.

- Algebraic approaches [6], [7], [8] – they generate test sets directly from a mathematical function as a generalized extension of the mathematical methods to design orthogonal arrays. As already known, an orthogonal array of power t requires that every possible combination of t columns to be covered exactly one time. Consequently, an orthogonal array can be considered as an optimal T-way test set, if we assume that every row is a test and every column a parameter. Other algebraic methods are based on recursive generation of large test arrays starting for small arrays.

Both algebraic and computational approaches have advantages and disadvantages. The computational methods can be applied to arbitrary system configurations, but need a great number of combinations to be enumerated that is a time consuming process. The algebraic alternatives produce optimal testing arrays, but impose restrictions to the configurations of the systems under test.

Referring to the nature of the parameters for T-way testing, basically there are two different forms of combinatorial methods: combination of values of configurable parameters or combination of values of input data values that allow the creation of a test suite for the application. In the last case combinatorial coverage of input data values is required for the resulting tests and this is the situation covered by this paper.

III. THE T-WAY TESTING COVERAGE

The test coverage is one of the most important topics related to software assurance. It offers, through formal methods or ad-hoc techniques, coverage metrics such as statement, decision or branch, condition or modified condition coverage [9], to evaluate the risk in using a software product. All these criteria have a major drawback – they depend on the access to the code. For the practical situations where the aspects of the source code are not visible, “black-box” techniques, such as the combinatorial testing, could offer valuable results. The application acts as a functional block that receives some inputs and filters them in order to produce predicted outputs. So, measuring

the state space of the inputs it is possible to produce precise and useful coverage metrics. If we consider, for example, a system with n inputs, every input having p values, then the state space of the inputs is a hypercube of n dimensions and p^n points to be tested. An exhaustive testing process is required to analyze the outputs for all p^n combinations of the the inputs. Combinatorial T-way testing offers a significant reduction of the tests and useful metrics based on the following statements [9]:

- For a set of t variables, a variable-value configuration is a set of t valid values, for each of the variables (a, b are two variables, then $a=0$ and $b=2$ is a variable-value configuration).

- For a given test set defined for n variables, the simple t -coverage is the proportion of t -way combinations of n variables for which all the variable-values configurations are fully covered. For example if we are looking for the coverage for two binary variables a and b , then a fully coverage means to find in the testing sets all the configurations variable-values $(0,0),(0,1),(1,0),(1,1)$.

- For a given set of n variables, the $(t+k)$ -way coverage is the proportion of $(t+k)$ –way combination of n variables for which all variable-values configurations are fully covered. This metric is useful for comparing different combinatorial test sets.

- For a given combination of t variables, the variable-value configuration coverage represents the proportion of variable-value configurations that are covered. If we consider four binary variables a,b,c,d , then we can found $C(4,2)=6$ possible variable combinations and $C(4,2) \times 2^2=24$ possible variable-value configurations. If for a given test set of the above variables two variable configurations are covered ((b,d) and (c,d)) and only 21 variable-value are covered ($ab=11, ac=11, ad=10$ are missing) then the t -way coverage is $2/6=33\%$ and the configuration coverage is $21/24=87.5\%$.

- for a given set of n variables, the (p, t) completeness is the proportion of $C(n,t)$ (n – the number of the variables, t – the interaction level) combinations that have configuration coverage at least p . For example, if the combination (a,b) is covered 50% (the combination $(1,1)$ and $(1,0)$ are missing) and each of the combinations $(a,c),(a,d), (b,c), (b,d),(c,d)$ are covered 75% then the completeness $(.50,2)$ of this set is 100%.

For all the above defined metrics various coverage measurements tools are available on the market. They uses existing test sets, identify the missing combinations in order to supplement the existing tests. On the other hand, the metrics for the configuration-spanning coverage can help to understand the state space coverage. For t -way combinatorial testing the configuration-spanning coverage is 100% for the selected t . An investigation on the coverage for $(t+1)$ or $(t+2)$ of the current set can be used to select between t -way covering arrays generated by different algorithms.

On the other hand, it is possible to measure the difference between the combinatorial t-way testing and the random testing that is one of the most used strategies in software industry. As mentioned in [9], the difference between these methods can be summarized in the following conclusions:

- For a system with all or almost all the variables with only two values the random and the combinatorial strategies are equivalent.
- For pairwise testing, the efficiency of the random testing decreases as the number of values for the variables increases.
- For 4-way testing for a reduced number of parameters (between 8 and 15) the combinatorial testing is significantly more efficient than the random one, but for more than 20 variables the performances are comparable for both methods.
- For a 100% of combinations coverage, the efficiency advantage of combinatorial testing varies directly with the number of values per variable and inversely with the interaction strength t . For this demand the combinatorial testing should be significantly less expensive than random test generation.

IV. THE IPOG FORMULATION

This paragraph details the algorithm behind the proposed T-Way Generator testing tool. It is based on the extension of the In-Parameter-Order (IPO) strategy [10], extensively used for pairwise testing, refer to as In-Parameter-Order-Order-General (IPOG) introduced by Yu Lei and al., in [11]. Our decision is based on some notable advantages to this approach, namely:

- Extends the pairwise testing to a general T-way combinatorial one, where T has an arbitrary value from 3 to the number of the input parameters for the system under test;
- Allows the generation of a minimum number of tests as the strength of coverage increases that reduces significantly the time and computation effort;
- It does not impose restrictions on the system under test configuration, computational methods being better suited for test cases generation; consequently it can be applied to general software applications;
- It is a deterministic strategy that generates repeatedly the same test sets as the system configuration remains unchanged.

For the presented IPOG implementation algorithm the following assumptions were made:

- The system under test has n parameters named $P_1, \dots, P_n$.
- There are interactions among T of these parameters ($3 \leq T \leq n$); there will be generated test sets for T -way

combinations of input parameters.

The generation process is guided by the following successive steps:

- For $(T-1)$ parameters, arbitrarily ordered, there will be generated all the combinations of the input values and collected into a primary test sets vector named t_{base} . The remaining T -way combinations are putting together into $t_{extended}$ sets vector. So $t_{base} \cup t_{extended}$ represents all the possible combinations of the n parameters taken T at a time. The main idea behind the IPOG algorithm is to reduce the number of combinations without a drastic reduction of the test covering.

- The t_{base} vector will be extended both horizontally and vertically by the following sequence of actions:

1) Each set is extended successively with each value of P_T parameter. For the resulting set there will be counted the occurrences number into the $t_{extended}$ vector. Finally, we select the P_T value that appears in the largest number of T -way combinations of the selected $P_1, \dots, P_n$ values into $t_{extended}$ upon a greedy algorithm. Then this set will be deleted from $t_{extended}$ so the next trial will be made into a reduced number of combinations.

2) The already formed test sets vector will be extended by vertical growth. New sets will be added, one at a time, to include all the uncovered combinations. A new testing set will be obtained either by changing an existing test, if possible, or by adding a new one in order to cover all the values of the $t_{extended}$ vector that remains after the first step. Finally $t_{extended}$ will become an empty set.

V. THE T-WAY GENERATOR TOOL DEVELOPMENT AND EXPERIMENTAL RESULTS

A. Analysis and Design process

The proposed T-way Generator tool offers an efficient solution for T-way combinatorial testing set generation and storage through a simple and clear operation. The application receives as inputs the number and the name of the correlated parameters, as depicted in Fig. 2. The output of the tool is the optimal IPOG test set (see Fig. 3). The C# developed solution has two projects that separate the programming logic from the user interface:

- The Core project is a Class Library comprising all the object entities. It includes also the IPOG implementation Tway class as illustrated in the package view from Fig. 1.
- The TwayGenerator that is the main project comprising the user interface and the controls logic.

During the implementation, the following problems had to be solved:

- The generation of the Cartesian product for the parameter values sets that is a time consuming procedure involving a great number of lines to solve this heuristic.

We chose the Language Integrated Query (LINQ) technology under .NET Framework that reduces the code to a single method named Cross as depicted in Fig. 1.

- The storage of the results, as the application manipulates a great number of integers or strings by a lot of costing operations. There were adopted the dynamic and generic list and dictionary structures that exist in .NET Framework.

- In order to allow further reuse of the code the application was divided in a DLL project type and an interface.

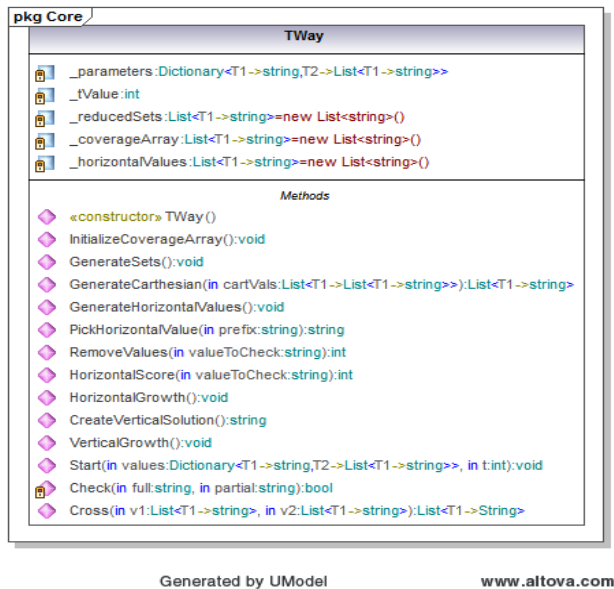


Fig. 1. The structure of the TWay class

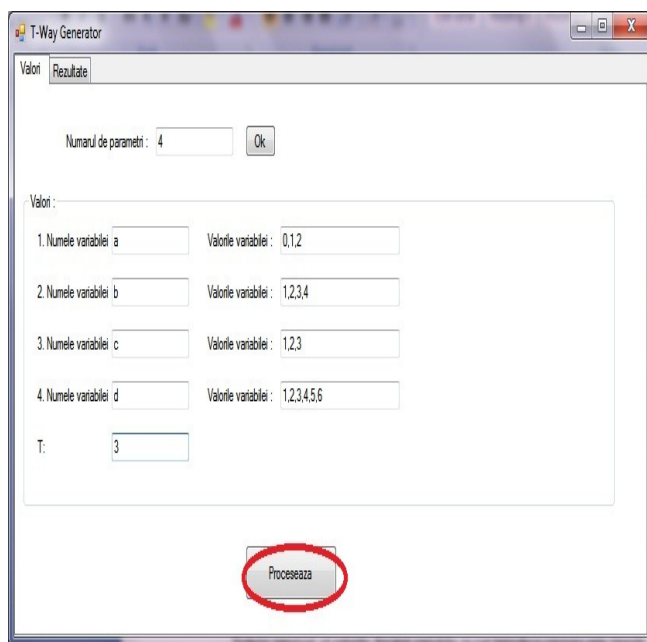


Fig. 2. The T-way Generator interface Values page

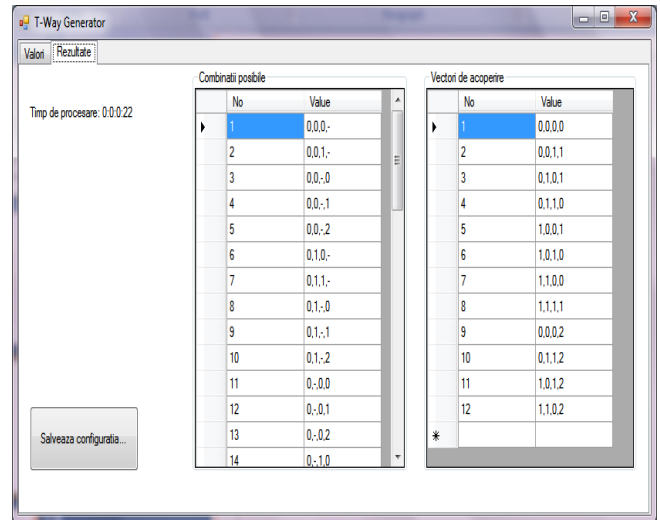


Fig. 3. The T-way Generator interface Results page

Two main tabs can be activated from the main application window, as depicted in Fig. 2: *Values* and *Results*. The *Values* page is used to insert the names and the discrete values for every input parameter and the covering power T for the combinatorial testing procedure. Activating the *Processing* button the *Results* page as depicted in the Fig. 3 will be open.

In order to explain the results from Fig. 3 we will detail the current setup. It can be noticed that different implementation solution and different startup for the same problem lead to slightly different solutions. The system under test has four input parameters P_1 , P_2 , P_3 , with two values 0 and 1 each, and P_4 having three values 0,1,2. The tool has to compute the 3-way test sets using the IPOG strategy. If the above setup would be the subject of a complete combinatorial 3-way testing then for the parameter combinations (P_1, P_2, P_3) , (P_1, P_2, P_4) , (P_2, P_3, P_4) , (P_1, P_3, P_4) will be obtained a total of $8+3 \times 12=44$ possible value combinations. The IPOG strategy will be applied to reduce this set to a minimal one without diminish of the testing coverage. Several actions will be made sequentially:

- 1) The initial test will contain an empty set of four values, one for each parameter.

- 2) An exhaustive combinatorial testing set is computed for an arbitrary combination of the first three parameters; if we assume that we chose $((P_1, P_2, P_3))$, then the $t_{base} = ((0,0,0), (0,0,1), (0,1,0), (0,1,1), (1,0,0), (1,0,1), (1,1,0), (1,1,1))$ having 8 different combinations. In order to cover the values of P_4 , there will be made horizontal and vertical extensions.

- 3) The $t_{extended}$ test set is computed. It results the remaining 36 combinations from the exhaustive set.

4) Each combination of the t_{base} is extended with a value for P_4 . It will be chosen the value that covers the largest number of combinations in $t_{extended}$ based on a greedy algorithm. For instance, the set $(P_1, P_2, P_3) = (0,0,0)$ can be extended with a P_4 value (0,1 or 2). For $P_4=0$, the following combinations can be covered: $(0,0,0,-)$, $(0,0,-,0)$, $(0,-,0,0)$, $(-,0,0,0)$ where “-“ denotes a don’t care value. Similarly, for $P_4=1$ there will be found the next combinations $(0,0,0,-)$, $(0,0,-,1)$, $(0,-,0,1)$, $(-,0,0,1)$ and for $P_4=2$ the set $(0,0,0,-)$, $(0,0,-,2)$, $(0,-,0,2)$, $(-,0,0,2)$. All the three values of P_4 are covered by 4 combinations from $t_{extended}$. So, we chose the first value $P_4=0$ (see Fig. 3 the first covering vector). Then the set $(0,0,0,0)$ will be deleted from $t_{extended}$. The procedure will be continued until all the 8 sets of t_{base} are extended.

5) After the horizontal extension, the following combinations remain uncovered: $((0,0,-,2), (0,1,-,2), (0,-,0,2), (0,-,1,2), (1,0,-,2), (1,1,-,2), (1,-,0,2), (1,-,1,2), (-,0,0,2), (-,0,1,2), (-,1,0,2), (-,1,1,2))$. For each of the above combinations, the value “-“ will be replaced by each possible value of the corresponding parameter. Finally it remains the value that has the biggest score. For instance, in the combination $(0,0,-,2)$ P_3 could be 0 or 1. $P_3=0$ will cover three combinations $(0,0,-,2)$, $(0,-,0,2)$ and $(-,0,0,2)$ and $P_3=1$ will cover another three combinations $(0,0,-,2)$, $(0,-,1,2)$, $(-,0,1,2)$. For the second combination of $t_{extended}$ $(0,1,-,2)$ $P_3=0$ will cover three sets, namely $(0,1,-,2)$, $(0,-,0,2)$, $(-,1,0,2)$ and $P_3=1$ another three sets $(0,1,-,2)$, $(0,-,1,2)$ and $(-,1,1,2)$. In the same manner each set from $t_{extended}$ will be analyzed. Finally, all the values that could replace “-“ for P_3 position cover exactly three values. So, $P_3=0$, the test $(0,0,0,2)$ will be added to t_{base} and all the combinations covered by $P_3=0$ will be deleted from $t_{extended}$. The same procedure will be repeated until $t_{extended}$ becomes an empty set.

B. Experimental results

We used the T-way Generator to make experiments and evaluate the results to achieve two different goals:

1) The evaluation of the dimensional testing sets growth and time increasing effort for different covering power (T) and number of input parameters (n).

2) The evaluation of the tool performances referring to the number of generated tests and the time of this process versus other open source similar applications.

In order to achieve the first goal three different experiments were made on a laptop running the operating system Windows 7x64, with the CPU frequency of 2.0 GHz and the RAM memory of 2GB. As shown in [11], the number of tests generated by IPOG algorithm is in $O(d \times \log n)$, where d is the largest domain size for the parameters, t is the covering power and n the number of parameters. The results as depicted in the Table I, II and

III are consistent with the complexity of the IPOG algorithm computed theoretically.

TABLE I
THE RESULTS FOR 8 PARAMETERS EACH HAVING 2 VALUES
WITH T FROM 2 TO 6 WAYS

T-way	2	3	4	5	6
Number of test sets	11	17	45	63	119
Time (sec.)	0.066	0.296	1.972	4.402	4.633

TABLE II
THE RESULTS FOR 5 TO 7 PARAMETERS EACH HAVING 4
VALUES FOR 4-WAY TESTING

Parameters number	5	6	7
Number of test sets	256	448	700
Time (sec.)	0.749	46	15 min.

TABLE III
THE RESULTS FOR 5 PARAMETERS EACH HAVING 2 to 8 VALUES
FOR 4-WAY TESTING

No. of values	2	3	4	5	6	7	8
No. of test sets	16	94	256	734	1504	2773	4096
Time (sec.)	0.001	0.093	0.780	14	120	540	720

To achieve the second experimental goal there were identified three existing similar applications that are open source or accessible for academic purposes, namely:

- Jenny [12] that implements a computational method; it is written in C and the details of the algorithm are not known.

- TConfig [13] from Ottawa University that implements a recursive construction method and is implemented in Java.

- ACTS [14] from the National Institute of Standards and Technology, United States Department of Commerce that implements in Java a version of IPOG algorithm.

The experiments were made using all the mentioned tools in order to generate test cases for the same module of a house controlling system that has 8 input parameters, 7 with 2 boolean values 0 and 1 and one that has three discrete values for the temperature sensor 24, 25 and 26 Celsius degrees. The results for all the analyzed tools are depicted in Table IV.

TABLE IV
THE PERFORMANCE OF DIFFERENT TOOLS FOR
HOUSE CONTROLLING APPLICATION

T-way/number of test sets	Tway Generator	Jenny	TConfig	ACTS
2	10	9	9	8
3	23	21	22	24
4	62	46	50	56
5	100	93	102	144
6	171	174	204	384

For all the input combinations the proposed T-way Generator has results comparable with other existing tools, with better performance for increased covering power (6-way test cases generation).

VI. CONCLUSIONS AND FURTHER DEVELOPMENT

In our opinion the T-way testing is a promising strategy for multiple reasons. First of all it is a specification based technique and, consequently, does not involve the knowledge of the system under test. The T-way specifications are easy to identify from the input parameters and their values domain. Second, as established by Kun et al. [3], [9], the failures of a large number of industrial applications are generated by the interactions of no more than 6 parameters. Consequently, the T-way testing could be a valuable solution for practical reasons. Furthermore, the process of tests generation could be automated as a distinct component that starts by pushing a simple button. This is extremely important for an industrial context acceptance.

The IPOG strategy could be further developed in some distinct directions:

- The presented algorithm has to enumerate in an explicit manner of all the possible combinations that it is difficult to achieve for increased number of correlated parameters. New technique that does not involve the enumeration of all the combinations would bring notable execution performances.

- The method could be extended to include the relationships between the parameters and their constraints. The parameter relationships are used to avoid the generation of combinations among non correlated inputs. The constraints-parameters relationship could be used to avoid the combinations that are not important for the domain semantics.

Finally, as T-way method generates a large number of test cases, it is not practical to execute and evaluate manually these ones. The presented tool can be integrated with other support facilities in order to automate the entire process of tests generation, execution and evaluation.

REFERENCES

- [1] R. Kuhn, R. Kacker, Y. Lei, and J. Hunter, „Combinatorial Software Testing,” *IEEE Computer*, vol. 42, no. 8, pp. 94 – 96, ISSN: 0018-9162, 2009.
- [2] B. Beizer, „*Software Testing Techniques*”, Van Nostrand Reinhold, New York, 2nd edition, 290 pp, ISBN-13: 978-0442245924, 1990.
- [3] D.R.Kuhn and V.Okun, „Pseudo-exhaustive Testing for Software”, The 30th NASA/IEEE Software Engineering Workshop, pp. 153-158, 2006.
- [4] P. Ammann and J. Offutt, „*Introduction to Software Testing*”, Cambridge University Press, New York, ISBN: 978-0-521-88038-1, 340 pp., 2008.
- [5] D.M.Cohen, S.R. Dalal, L. Fredman, and G.C. Patton, „The AETG System: An Approach to Testing Based on Combinatorial Design”, *IEEE Transactions on Software Testing*, (vol. 23 no. 7) pp. 437-444, 1997.
- [6] E.M. Clarke, K.L. McMillan, S. Campos, and V. Hartonas-Garmhausen, „Symbolic model checking”, In Rajeev Alur and Thomas A. Henzinger, editors, Eighth International Conference on Computer-Aided Verification; in „*Lecture Notes in Computer Science*”, no. 1102, New Brunswick, NJ, USA, pp. 419-422, 1996.
- [7] A. Hartman, L. Raskin, „Problems and algorithms for covering arrays”, *Discrete Mathematics* 284 (1-3), pp. 149-156, 2004.
- [8] A.W.Williams and R.L. Probert, „A practical strategy for testing pairwise coverage of network interfaces”, 7th International Symposium on Software Reliability Engineering (ISSRE), White Plains, New York, pp.246, ISBN: 0-8186-7707-4, 1996.
- [9] D.R. Kuhn, R. N. Kacker, and Y. Lei, „*Practical Combinatorial Testing*”, National Institute of Standards and Technology (NIST) Special Publication 800-142, 2010. [Online]. Available: <http://csrc.nist.gov/groups/SNS/acts/documents/SP800-142-101006.pdf>
- [10] Y. Lei and K.C. Tai, „*In-parameter-order: a test generation strategy for pairwise testing*”, The 3rd IEEE International Conference on High-Assurance Systems Engineering Symposium, pag. 254-261, 1998.
- [11] Y. Lei, R. Kacker, D.R.Kuhn, V. Okun, and J. Lawrence, „IPOG: A General Strategy for T-Way Software Testing”, Engineering of Computer-Based Systems Conference, ECBS '07, pp.549 - 556 , ISBN: 0-7695-2772-8, 2007.
- [12] Jenny combinatorial tool [Online]. Available: <http://www.burtleburtle.net/bob/math/>
- [13] TConfig combinatorial tool [Online]. Available:, <http://www.site.uottawa.ca/~awilliam/>
- [14] ACTS tool [Online]. Available:, <http://csrc.nist.gov>

Rule-based Reasoning for Altering Pattern Navigation in Programming Tutoring System

Boban Vesin, Mirjana Ivanović, Aleksandra Klačnja-Milićević and Zoran Budimac

Abstract - Semantic Web technologies seem to be a promising technological foundation for the next generation of e-learning systems. Although ontologies have a set of basic implicit reasoning mechanisms derived from the description logic which they are typically based on (such as classification, relations, instance checking, etc.), they need rules to make further inferences and to express relations that cannot be represented by ontological reasoning. In our previous work, we presented tutoring system named Protus (PRogramming TUtoring System) that is used for learning Java programming basics. One of the most important features of Protus is the adaptation of the presentation and navigation system of a course based on the level of particular learner knowledge. It uses principles of adaptive hypermedia and content recommendation for course personalization. There can be different sequence of resources that depends on navigational sequence determined for particular learner. In this paper we present set of proposed SWRL rules for altering navigation sequences.

I. INTRODUCTION

Semantic Web technologies seem to be a promising technological foundation for the next generation of e-learning systems. Ontologies allow specifying formally and explicitly the concepts that appear in a concrete domain, their properties and their relationships [1]. Furthermore, they are useful in many environments: and especially in educational environments, as they enable people and/or software agents to share a common understanding of the knowledge structure. Moreover, ontologies permit to reuse knowledge. It is not necessary to develop ontology from scratch if another ontology is available for use in the modeling of the current domain.

Ontologies are typically based on the description logic which has a set of basic implicit reasoning mechanisms (such as classification, relations, instance checking, etc.). Nevertheless, they need additional rules to make further inferences and to express relations that cannot be represented by ontological reasoning [2]. Ontologies require a rule system to derive/use further information that cannot be captured by them, and rule systems require ontologies in order to have a shared definition of the concepts and relations mentioned in the rules. In our current research, we are going to exploit SWRL [3], a Semantic Web Rule

Language combining OWL and RuleML. We use the SWRL as a reasoning and inference mechanism to obtain the essential production rules, as well as analyze the domain knowledge and interaction data. SWRL allows interoperability with major rules systems (commercial or noncommercial) such as JESS [2]. Rules defined in SWRL can describe the business logic. Then rule-based reasoning can be achieved to generate adaptation in e-learning systems.

The major goal of learning systems is to support a given pedagogical strategy [4]. In order to achieve specific pedagogical goals, ontologies can be associated with reasoning mechanisms and rules to enforce a given adaptation strategy in learning system. Often this strategy consists of selecting or computing a specific navigation sequence among the learning resources. Thus, formal semantics are required here to enable such computation. Adaptation presents a key feature for modeling educational environments capable to present learning objects in the adequate order based on the specific learning profile of each individual learner.

In our previous work, we presented tutoring system named Protus (PRogramming TUtoring System) that is used for learning Java programming basic concepts [5], [6]. One of most important features of Protus is the adaptation of the teaching material and navigation in a course based on the level of knowledge of a particular learner. Adaptation module of Protus uses principles of adaptive hypermedia and content recommendation for course personalization. Ontology architecture for tutoring system supported by several ontologies was proposed in our previous works as a way of addressing the problems of maintenance and reuse of system's components.

Original architecture of Protus was extended by the use of ontology, where the representation of each component is made by a specific ontology [7]. This way, it is possible to promote a clear separation of concerns of the tutoring system's components:

- to make explicit the communication among the components,
- to specify the support to build the components and
- to emphasize the gains of the use of Semantic Web in the development of tutoring systems.

That will also allow better interoperability and reusability of Protus components in the future.

This ontology-based approach allows us to implement efficient adaptation in Protus. The main objective of the

B. Vesin and A. Klačnja-Milićević are with the Higher School of Professional Business Studies, Novi Sad, Serbia (e-mail: {bobanvesin,aklasnja}@yahoo.com).

M. Ivanović and Z. Budimac are with the Faculty of Science, Department of Mathematics and Informatics, Novi Sad, Serbia (phone: +381 21 458 888, e-mail: {mira.zjb}@dmi.uns.ac.rs).

paper is to propose new features of Protus that will allow rule-based reasoning. The learner demand is derived from the knowledge contained in the ontology. Various conditions are captured in the body of SWRL rules. As a result of the firing of rules, recommendations in the form of adapted navigation sequence are generated. This can be used to implement the concept of adapted content and adapted navigation.

The rest of this paper is organized as follows: Section 2 describes the related literatures review. We present the Protus architecture and principles of the implemented adaptation in Section 3 and 4, respectively. This is followed by a description of the proposed rules in Section 5. Finally, we draw some conclusions in Section 6.

II. RELATED WORK

Various approaches to adapting navigation and sequencing have been presented in numerous projects. The vast majority of existing tutoring systems can adapt navigation to only one kind of teaching operation. For example, a number of tutoring systems including [8], [9] can only manipulate the order of questions or problems during testing. This approach was usually called task sequencing. A number of authors made a step further and their systems can do sequencing of educational material [10], [11]. The most advanced systems are able to sequence several kinds of teaching operations such as presentation of the teaching material, examples, and assessments [12], [13].

A wide range of educational software that implements ontology-based components has been developed in recent period, but the most of these systems use ontologies only for representation of concepts, knowledge or learners data [1], [14]-[16].

Several studies show how semantic rules can be used in combination with ontologies for providing adaptation process in e-learning systems [17]-[20]. In the structure of previously mentioned systems, the use of ontology focuses mainly on learning objects and their related aspects. Besides, that does not facilitate the definition and communication between the other components of the system's architecture. Architecture of Protus supported by several ontologies was described in [7]. We further enhance that work by not only adapting the content modeling but we will show a way how to link semantics and content with implementing rule-based reasoning for adaptation process in Protus.

III. PROTUS ARCHITECTURE

Protus is a tutoring system designed to support learning processes in different courses and domains but with intention to be used for learning programming languages. General ideas, of ontology-based architecture for Protus have been presented in [7]. Protus architecture is highly modular with five central components: *the application module, the adaptation module, the learner model, session monitor* and

domain module. The general architecture of Protus is specially constructed to allow several types of adaptation (Figure 1).

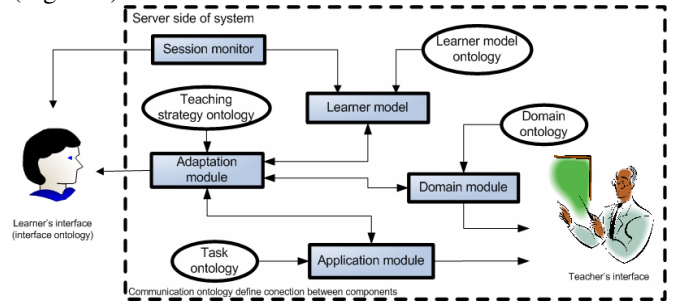


Fig. 1. Overall architecture of Protus

Domain module of Protus is assumed to be hierarchically structured into *Concepts* that are supported by the different *Resources* (Figure 2). There are several types of the *Resources* in Protus: tutorials, task, examples, tests, etc. A *hasPrerequisite* property is given by this hierarchical document structure and it is proposed for navigational purposes. It assigns the next document in sequence for each resource. It allows pointing out concepts that must be known before starting to study a concept, and the concepts for which it is a prerequisite. The concept will not be covered unless the prerequisite condition is satisfied. There can be different sequence of resources that depends on navigational sequence determined for particular learner.

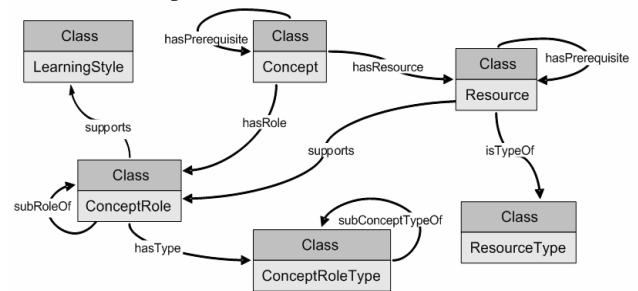


Fig. 2. An excerpt of domain ontology

IV. RESOURCE SEQUENCING

According to [19], adaptation in tutorial systems can be done with regard to three levels:

- *Content Level Adaptation* by means of content hiding, additional explanations, different presentation methods, specific learning concepts filtering, adaptation to different student learning styles, etc.
- *Presentation Level Adaptation* by means of inserting/removing resources (learning objects), altering/sorting resources, adapting resources, etc.
- *Navigation Level Adaptation* by means of direct guidance, link ranking, link disabling/hiding, link annotation, link generation, etc. Often this strategy consists of selecting or computing specific navigation sequences among the resources.

Issues of content and presentation level adaptation implemented in Protus were subjects of our previous work

[5], [6], while focus of this paper will be navigational level adaptation.

Designers and instructors, while developing the on-line structure of the course and course material, have a navigation pattern in mind and assume all on-line learners would follow a consistent path. That path was determined in the design and materialized by some hyperlinks [21]. Learners, however, could follow different paths generating a variety of sequences of learning activities. Often this sequence is not the optimal sequence, and probably not the sequence intended by the designer.

Resource sequencing is a well-established technology in the field of intelligent tutoring systems [22]. The idea of resource sequencing is to generate a personalized course for each learner by dynamically selecting the most optimal teaching actions, presentation, examples, task or problems at any given moment. By optimal teaching action it is considered an operation that in the context of other available operations brings the learner closest to the ultimate learning goal. Most often the goal is to learn and acquire some knowledge up to a specific level in an optimal amount of time. However, it is easy to imagine other learning goals, such as minimizing student error rates in problem solving.

As stated in [23] the instructional contents' adaptation to the learner's individual characteristics implies that the system must:

- recognize the cognitive patterns of each learner and its pedagogical implications
- know the domain knowledge being proposed
- know the organization of domain material, and
- be able to dynamically generate the best suited selection and sequence for each learner in a particular course stage.

In contrast to traditional classroom-based learning, the learning behavior in web-based environments is more determined by the learner's own decisions how to organize learning process [24]. Learners could follow different paths based on their preferences and generate a variety of learning activities. All these variations in series of learning activities are noted down by the Protus system. In order to investigate learning activities in detail, sequential pattern mining algorithm of AprioriAll [25] is adopted to extract behavioral (interaction) patterns from the log file. These patterns will be used to analyze how learners evolve from the beginning of learning of particular unit, until they successfully finish it, or less successfully, give up. Learners with different learning habits and/or knowledge make different sets of frequent sequence. Hence, behavioral patterns were discovered for each learner by AprioriAll algorithm.

When the learner completes the sequence of learning materials, the Protus system evaluates the learner's acquired knowledge (Figure 3). The learners' ratings can be interpreted according to the percentage of correct answers. Two learners are said to be similar to each other if they are evaluated by the system with the same ratings for a similar

navigational sequence. Recommendation process can be carried out according to these learning sequences based on the collaborative filtering approach that is described in our earlier work [6].

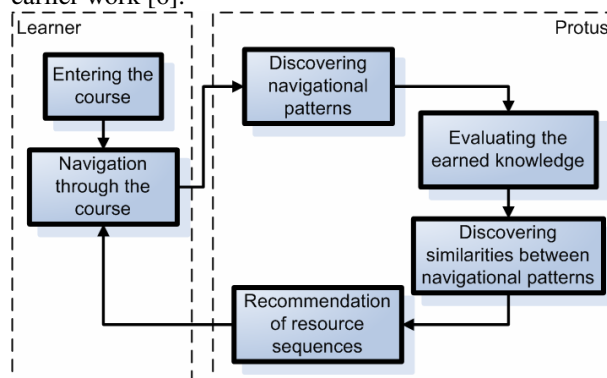


Fig. 3. Learning and recommendation process in Protus

There can be different a sequence of resources that depends on navigational sequence determined for particular learner. In next section we will present the set of proposed SWRL rules for altering navigation sequences. The benefit of representing teaching strategies as SWRL rules is that the strategies' computations would be explicitly represented in the ontology, and could be viewed and edited, as well as reasoned about by other applications.

V. IMPLEMENTED RULES

SWRL rules are one of the most popular forms of knowledge representation, due to its simplicity, comprehensibility and expressive power [11]. There are different types of rules depending on the knowledge they store. They are referred to as: decision rules, association rules, classification rules, prediction rules, causal rules, optimization rules, etc. The types of rules we will use are the mostly decision and classification rules. Decision rules are used for choice of presented learning objects (resources) while classification rules helps in categorization of those resources.

In some cases, the interface elements for sequential navigation (in our case the buttons *Next* and *Previous*) are hidden, giving the learner the possibility to freely jump through the courseware. Furthermore, there will be added links to related or more complex content to help situate the learnt subject and contribute in creating clear overall view on the subject being thought.

Adaptation module of Protus used for building automatic recommendations will be composed of three groups of rules [6]:

- learner-system interaction rules
- off-line rules
- recommendation rules

The details of the whole process of recommendation and examples of implemented rules are presented in the rest of the paper.

Learner-system interaction rules pre-process data to build learner models. *Interaction* presents any kind of action performed by learner on specific learning object (page visiting, submitting test and tasks results, search and browse options, requesting additional examples, etc.). The data about learners' activities (like sequential patterns, pages that learner had visited, test results and grades earned) are collected with this rules. When the learner completes the sequence of learning materials, the system evaluates the learner's knowledge for each lesson [6]. Following rule updates learner model.

```
Learner(?x) ∧ Interaction(?y) ∧
hasInteraction(?x,?y) ∧ Concept(?c) ∧
conceptUsed(?y,?c) ∧ Performance(?p) ∧
hasResult(?y,?p) ∧ hasGrade(?p,?m) ∧
swrlb:greaterThan(?m, 1) → isLearned(?c,
true) ∧ hasPerformance(?x,?p)
```

With the previous rule, Protus is using recorded results of learner's interaction, earned grade and data about used concepts to memorize learner's performance in the session. Variables *x*, *y*, *c*, *m* and *p* present *Learner*, *Interaction*, *Concept*, *Grade* and *Performance*, respectively. Class *Performance* contains results of learner's interaction with specific concept (learning object): time, date, possible grade range, earned grade, etc. Meaning of the rule is: if in any time of the execution of Protus, there exists learner which interacts with specific concept, and during that interaction he/she took the test and earned specific grade, than system should memorize that learner's performance and mark that concept as learned. Rule is executed only if the learner earned a positive grade. The learners' grades have been interpreted according to the percentage of correct answers [6], as follows:

- 5 (excellent) - (80–100%)
- 4 (good) - (70–79%)
- 3 (average) - (60–69%)
- 2 (passing) - (50–59%)
- 1 (marginal) - (0–49%)

This five-point grading scale is based on our secondary school grading system. Consequently, learners have a better sense of having mastered the material using this system of evaluation. The system can be easily transformed and adapted to other standards of grading.

In the same time, system must update database of learners' interactions that is later used for grading of sequential patterns. Following rule is used for that matter:

```
Learner(?x) ∧ Concept(?c) ∧ Resource(?r) ∧
hasResource(?c,?r) ∧ isLearned(?c, true) ∧
hasPerformance(?x,?p) ∧ BehaviourPattern(?b)
∧ NavigationSequence(?n) ∧ isTypeOf(?b,?n) →
swrl:add(?n,?r,?p)
```

Previous rule adds visited resource to the navigation sequence in the current session. Built in function

swrl:add(?x,?y,?z) adds resource *y* and details about performance *z* to navigation sequence *x*. Variables *x*, *r*, *c*, *b*, *p* and *n* presents *Learner*, *Resource*, *Concept*, *Behavior Pattern*, *Performance* and *Navigation Sequence*, respectively. Meaning of the rule is: if in any time of the execution of Protus, exists learner who successfully learned specific concept which is supported with specific resource, than system should add that resource and details about performance to the successful navigational pattern.

Off-line rules use learner models on-the-fly to recognize learners' goals and content profiles. Learners are grouped into clusters. These clusters are used to identify coherent choices in frequent sequences of learning activities [6]. Then, a recommendation list can be created according to the ratings of these frequent sequences provided by the Protus system. First, Protus distinguishes cluster that learner belongs to with one of the appropriate rules. Example of rule that is triggered if learner belongs to cluster *c11* is (analog rules are used for other clusters):

```
Learner(?x) ∧ Performance(?p) ∧
hasPerformance(?x,?p) ∧ Condition(?c) ∧
generates(?p,?c) ∧ BehaviourPattern(?b) ∧
include(?p,?b) ∧ isTypeOf(?b,?n) ∧
NavigationSequence(?n) ∧ consistsOf(?n,a1) ∧
consistsOf(?n,b2) ∧ consistsOf(?n,c1) ∧
consistsOf(?n,d1) ∧ consistsOf(?n,e4) ∧
swrlb:greaterThan(grade, required) →
belong(?x,?c11)
```

This rule determines that learner belongs to the appropriate pattern, based on navigation sequence *n* that contains learning resources {a1, b2, c1, d1, e4}. This sequence is a list that contains: taken tutorial, visited examples, tasks and tests taken, etc. In this case, determined pattern is *c11*.

Variables *x*, *p*, *c*, *b* and *n* present *Learner*, *Performance*, *Condition*, *Behavior Pattern*, and *Navigation Sequence*, respectively. *Condition* class collects data about learner's performance and his/her learning style and generates appropriate type of performed personalization that will be implemented. Generated personalization, in fact, presents specific navigational pattern recommended to learner. Meaning of the rule is: if in any time of the execution of Protus, exists learner which interacts with system under specific condition and during that interaction he/she successfully completed navigational sequence (predefined for specific behavior pattern), than that sequence can be treated as appropriate for that learner and he/she should be put in adequate cluster. Pattern discovering is only executed if learner successfully completes navigation sequence, that is to say, if learner has earned sufficient grade.

Recommendation rules produce a list or recommended learning objects. From the filtered list of learning content and based on the discovered sequences, the list of recommended actions and recourses is sent to alter learner–

system interaction within a new session. The recommendation module is design to create a recommendation list according to the ratings of these frequent sequences, provided by the Protus system [6]. For example, if the learner is determined to belong to cluster c11, it means that he/she is attended sequence of resources: {a1, b2, c1, d1, e4}. Based on that initial sequence, Protus rated highest extended set of resources: {a1, b2, c1, d1, e4, f2} and, therefore, resource f2 is recommended to him/her with next rule:

```
Learner(?x) ∧ Performance(?p) ∧
hasPerformance(?x,?p) ∧ Condition(?c) ∧
generates(?p,?c) ∧ BehaviourPattern(?b) ∧
include(?p,?b) ∧ isTypeOf(?b,?n) ∧
NavigationSequence(?n) ∧ belong(?x,?c11) →
isRecommended(f2,true)
```

Meaning of the previous rule is: if in any time of the execution of Protus, exists learner whom specific navigation sequence of resources has been recommended (with specific condition) than system should recommend to him/her next specific resource that belongs to that navigational sequence. Recommendation status of the resource f2 is set to true, therefore link to that resource is annotated or highlighted. Following of the recommended path does not influence the rules itself, but influence execution of rules in next sessions because ratings of frequent navigation sequences are calculated after every session. Ratings of frequent sequences are not calculated only by followed sequences itself but earned grades throughout session are also included in calculation. Therefore, every system-imposed path still counts towards placing the learner in a particular cluster.

VI. CONCLUSION

In the previous sections, both OWL and SWRL were used to demonstrate that the generic benefits promised by the Semantic Web can be useful to formally represent instructional models, and to let computers to automatically deduce new knowledge from existing facts and allow resource sequencing and adaptation of navigation patterns.

While the ontology represents the different adaptivity dimensions in terms of learner, domain, task, environment and system model, in this paper we tried to demonstrate that the logic underlying the adaptation can also be explicitly captured on the basis of a rule-based model. Our aim was to increase interoperability, reusability and extensibility by the use of SWRL rules.

In Protus, behavioral patterns are discovered for each learner by AprioriAll algorithm. Next, a recommendation list is created according to the ratings of the frequent sequences, provided by the Protus system. The provided recommendation is expected to have a higher accuracy in matching learners' requirements to learning material and thus a higher level of acceptance by the learners.

Mining the frequent sequences has the potential to

improve the quality of an intelligent tutoring system, as well as keep the recommendation up-to-date. We have presented a set of adaptation rules that completes personalized recommendation of the learning content according to the ratings of these frequent sequences, provided by the Protus system.

The automatic recommendation is based on the intended sequence of navigation in the course material, or based on navigation patterns of other successful learners. This technique is used for recommending shortcuts or jumps to some resources to help learners better navigate the course materials.

As a result of the execution of rules, recommendations in the form of content presentation are generated, which can be used to implement the concept of adapted navigation. Our goal was to offer a simple concept for adaptation and personalization of both the content and the navigation to the user needs and goals. For the future work, we consider presenting how the system adapts to various learners in the actual circumstances.

REFERENCES

- [1] J. M. Gascueña, A. Fernández-Caballero, and P. González, "Domain ontology for personalized e-learning in educational systems," *The Sixth IEEE International Conference on Advanced Learning Technologies*, 2006, pp. 456-458.
- [2] F. Carmagnola, F. Cena, C. Gena, and I. Torre, "A semantic framework for adaptive web-based systems," *In Proceedings of SWAP 2005, the 2nd Italian Semantic Web Workshop, Trento, Italy*, 2005, p.14
- [3] SWRL: A semantic Web rule language combining OWL and RuleML. Retrieved 19.12.08 from <<http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/>>
- [4] S. Dehors, and C. Faron-Zucker, "QBLs: A semantic Web based learning system," *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*, 2006, pp. 795-802.
- [5] A. Klačnja-Milićević, B. Vesin, M. Ivanović, and Z. Budimac, "Integration of recommendations and adaptive hypermedia into Java tutoring system," *Computer Science and Information Systems – ComSIS*, Volume 8, Number 1, 2011, pp. 211-224.
- [6] A. Klačnja-Milićević, B. Vesin, M. Ivanović, and Z. Budimac, Z. "E-learning personalization based on hybrid recommendation strategy and learning style identification," *Computers & Education* 56, 2011, pp. 885-899.
- [7] B. Vesin, A. Klačnja-Milićević, M. Ivanović, and Z. Budimac, "Ontology-based architecture for providing recommendation in Java tutoring system," *The 6th International Conference on Information Technology ICIT 2011 Jordan*, 2011, to be published
- [8] P. Brusilovsky, and J. Vassileva, "Course sequencing techniques for large-scale web-based education," *International Journal of Continuing Engineering Education and Lifelong Learning*, 13(1-2), 2003, pp. 75-94.
- [9] A. Rios, E. Millarn, M. Trella, J. L. Perrez, and R. Conejo, "Internet based evaluation system," *In S. P. Lajoie and M. Vivet (Eds.), Artificial intelligence in education: open learning environments. Amsterdam: IOS Press*. 1999, pp. 387-394
- [10] P. Brusilovsky, "ILEARN: an intelligent system for teaching and learning about UNIX," *In Proceedings of SUUG international open systems conference. Moscow: ICSTI*, 1994, pp. 35-41.
- [11] C. Romero, S. Ventura, C. Hervás and P. Gonzales, „Rule mining with GBGP to improve web-based adaptive educational systems“, *Data mining in e-learning, wit press*, 2006, pp. 173-190.
- [12] M. Chen, M. H. Chang, C. Y. Liu, and W. C. Chiu, "Personalized learning path generation based on real-coded genetic algorithm for

- web-based instruction,” In *Proceedings of the 16th international conference of information management (ICIM'05)*, Taiwan, 2005, p. 56.
- [13] E. Mor, J. Minguillon, and J.M. Carbo, „Analysis of user navigational behaviour for e-learning personalisation,” *Data mining in e-learning, wit press*, 2006, pp. 227-246.
- [14] C. Hee Lee, J. Hyun Seu, and M.W. Evens, “Building an ontology for CIRCSIM-tutor,” *13th Midwest AI and Cognitive Science Society Conference*, Chicago, 2002, pp. 161–168.
- [15] R. Mizoguchi, Y. Hayashi, and J. Bourdeau, “Inside theory-aware and standards-compliant authoring system,” *SWEL Workshop of Ontologies and Semantic Web Services for IES*, AIED, 2007, pp. 1-18.
- [16] P. Dolog, and W. Nejdl, “Semantic Web technologies for the adaptive web,” *The Adaptive Web, LNCS 4321, Springer-Verlag Berlin Heidelberg*, 2007, pp. 697–719.
- [17] M.A. Sicilia, M.D. Lytras, S. Sances-Alonso, Garcia-Barriocanal E. and Zapata-Ros M., “Modeling instructional-design theories with ontologies: Using methods to check, generate and search learning designs,” *Computers in Human Behavior*, 2010, doi:10.1016/j.chb.2010.07.040
- [18] Y.L. Chi, “Ontology-based curriculum content sequencing system with semantic rules,” *Expert Systems with Applications* 36, 2009, pp. 7838–7847.
- [19] E. Popescu, C. Badica and P. Trigano, “Rules for learner modeling and adaptation povisioning in an educational hypermedia system,” *Ninth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, 2007, pp. 492-499.
- [20] N. Henze, and W. Nejdl, “A logical characterization of adaptive educational hypermedia”, *Hypermedia - Special issue: Adaptive hypermedia in the age of the adaptive web Volume 10 Issue 1*, 2004, pp. 77 – 113.
- [21] A. Klačnja-Milićević, B. Vesin, M. Ivanović, and Z. Budimac, “Integration of recommendations into Java tutoring system,” *The 4th International Conference on Information Technology ICIT 2009 Jordan*, 2009, Paper no 154.
- [22] M.J. Huang, H.S. Huang, and M. Y. Chen, “Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach,” *Expert Systems with Applications* 33, 2007, pp. 551–564.
- [23] M. A. M. Souto, et al., “Web-adaptive training system based on cognitive student style,” *IFIP World Computer Congress, 17th Edition, Montréal, Canada*, 2002, pp. 281 – 288.
- [24] P. A. Northrup, “Framework for designing interactivity into web-based instruction,” *Educational Technology*, 41(2), 2001, pp. 31–39.
- [25] W. Tong, and H. Pi-Lian, “Web log mining by an improved Aprioriall algorithm,” In: *Proceeding of world academy of science, engineering, and technology*, 2005, pp. 97–100.

Statistical analysis and modeling of the PM₁₀ concentration values in Dolj County (Romania) between 2008 and 2010

Gabriel Vladut, Monica Mateescu, Liana Simona Sbirna, Sebastian Sbirna and Anisoara Oubraham

Abstract— Air quality is analyzed in Dolj (as well as in the other Romanian counties) by means of automatic air quality monitoring stations. The current study presents statistical processing of all data regarding PM₁₀ concentration values – gathered from one of the five such stations situated in Dolj County – during a period of three years (however, it should be noted that nefelometric measurements – *i.e.*, automatically recorded data – must be validated by gravimetric means, so that only validated data are reported to the European Commission). The paper shows that a log-normal distribution appears, by taking into account these nefelometric data and that some specific parameters may be consequently calculated, in order for the log-normal statistical distribution function to fit to the experimental

curve. The concentration values of PM₁₀ exceeded for twenty-six times the critical value 120 µg/m³, which is the maximum daily admitted eight hours average.

I. INTRODUCTION

AS ambient aerosol concentrations have been implicated in human health effects, in visibility reduction in urban and regional areas, in acid deposition and in perturbing the Earth's radiation balance, a main concern of the air quality managers is to achieve compliance to the established air quality standards. As they are exceeded in numerous sites worldwide, it is essential to reduce the air pollutants.

Ambient particulate matter is considered as a serious air pollution problem in urban agglomerations, because of its detrimental impact on public health and living standards. A large number of epidemiological studies have been conducted around the world that ascertain associations between ambient particle concentrations and excesses in daily mortality and morbidity [1]-[4].

Atmospheric aerosols are also responsible for light scattering, cloud formation, acid deposition, reduction in visibility and they are a key factor in

Manuscript received May 31, 2011.

G. Vladut is with the S. C. IPA CIFATT Craiova company, 12 Stefan cel Mare Street, Craiova. He is an IEEE Member (e-mail: office@ipacv.ro).

M. Mateescu is with the Faculty of Chemistry, University of Craiova, which is situated on 1071 Calea Bucuresti Street, Craiova (e-mail: monica.daniela.mateescu@gmail.com).

L. S. Sbirna is with the Faculty of Chemistry, University of Craiova, 1071 Calea Bucuresti Street, Craiova (e-mail: simona.sbirna@gmail.com).

S. Sbirna is with Craiova Regional Environment Protection Agency, 1 Petru Rares Street, Craiova (e-mail: s.sbirna@gmail.com, phone: 0721.329779).

A. Oubraham is with National Institute for Cryogenics and Isotopic Technologies, 4 Uzinei Street, Rm. Vâlcea (e-mail: anisoara@icsi.ro).

balancing Earth's radiation and consequently in perturbing global climate [5].

European Union has established air quality standards for particulate matter with an aerodynamic diameter lower than $10\text{ }\mu\text{m}$ (PM_{10}), in order to improve the quality of ambient air. The former Directive 1999/30, as well as the later Directive 2008/50, state that since January 1st 2005, the daily average (DA) value of PM_{10} concentration should not exceed $50\text{ }\mu\text{g}/\text{m}^3$ on more than 35 days per year and the annual average (AA) value of PM_{10} concentration should not exceed $40\text{ }\mu\text{g}/\text{m}^3$. The concentration of air pollutant is influenced by the air pollutants levels, meteorological conditions and topography

When the parent probability distribution of air pollutants is properly chosen, this specific distribution

can be used to estimate the mean concentration. A piece of information regarding the frequency distribution is necessary for developing air pollutant control strategies.

There are many types of distributions used to fit the parent distribution of an air pollutant concentration. The log-normal distribution is the most popular distribution used for fitting air pollutant concentration [6].

According to the theory of successive random dilutions [7] and assuming a single source with quite constant emissions, the pollutants undergo successive mixing and diluting before reaching the receptor, resulting in a log-normal frequency distribution.

Earlier studies have concluded to the conclusion that this is the most appropriate parent distribution to represent PM_{10} concentration [8].

In this study, it is verified that the PM_{10} concentration distribution can be adequately simulated by log-normal distribution.

II. AIR QUALITY MONITORING IN DOLJ COUNTY

AMONG other air pollutants, in Dolj County (south-western Romania) PM_{10} is monitored at modern automatic air quality monitoring stations, and data automatically registered by them are then validated by experts from Craiova Regional Environment Protection Agency. Nefelometric measurements – *i.e.*, automatically recorded data – must be validated by gravimetric means, so only validated data are reported to the European Commission).

The results obtained during the period 2008-2010 are presented below (a similar study for 2011 is in progress, showing no big differences).

III. STATISTICAL ANALYSIS AND MODELING

DATA from experiments, product testing, simulation, surveys, and statistical process and quality control must be appropriately analyzed before results can be determined and conclusions drawn.

Statistical analysis and modeling involves the appropriate application of statistical analysis techniques, each requiring certain assumptions be met, to perform hypothesis tests, interpret the data, and reach valid conclusions.

To be credible, results from experimentation or testing must be obtained following established statistical procedures, including experimental design and the appropriate use of statistical analysis and modeling techniques.

These results can then be reproduced, within sampling error, by repeating the experiment.

Statistical analysis and modeling requires careful selection of analytical techniques, verification of assumptions, and verification of data.

Descriptive statistics, graphs, and relational plots of the data should first be examined to evaluate the legitimacy of the data, identify possible outliers and assumption violations, and form preliminary ideas on variable relationships for modeling.

The many different statistical analysis and modeling techniques have different goals and are appropriate for different types of data.

Statistical analysis and modeling has important benefits, such as: application of appropriate statistical analysis techniques; development of appropriate conclusions and key learning from the data; ensuring results address experimental objectives; maximizing information gained from the data; maximizing chances of the interpretation being successful.

IV. LOG-NORMAL DISTRIBUTION

IN the probability theory, a log-normal distribution is a probability distribution of a random variable whose logarithm is normally distributed.

A variable might be modeled as log-normal if it can be thought of as the multiplicative product of many independent random variables each of which is positive.

V. PARAMETERS AND PROBABILITY DENSITY FUNCTION OF THE LOG-NORMAL DISTRIBUTION

THE The parameters of the log-normal distribution, namely the geometric mean μ_g and the standard geometric deviation σ_g , are

defined by (1) and (2), respectively. The probability density function of the log-normal distribution, the cumulative distribution function,

which is defined as the probability of the variable x to be smaller than a critical value x_0 and the complementary distribution function, which is defined as the probability of the variable x exceeding the critical value x_0 , are given by (3), (4) and (5), respectively. The variable m that is involved in (4) is calculated by (6). In (1)-(6), variable x represents the concentration of a pollutant.

$$\mu_g = (x_1 \cdot x_2 \cdot \dots \cdot x_N)^{\frac{1}{N}} \quad (1)$$

$$\sigma_g = \exp \left\{ \frac{1}{N} \left[\sum_{i=1}^N (\ln x_i - \ln \mu_g)^2 \right] \right\}^{\frac{1}{2}} \quad (2)$$

$$p(x_i) = \frac{1}{\sqrt{2\pi} \times x_i \times \ln \sigma_g} \exp \left[- \frac{(\ln x_i - \ln \mu_g)^2}{2 \times \ln^2 \sigma_g} \right] \quad (3)$$

$$F(x) = \Pr[x_i < x_0] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^m e^{-\frac{t^2}{2}} dt \quad (4)$$

$$F'(x) = \Pr[x_i > x_0] = 1 - F(x) \quad (5)$$

$$m = \frac{(\ln x_0 - \ln \mu_g)}{\ln \sigma_g} \quad (6)$$

VI. PARENT AND OBSERVED DISTRIBUTION

THE suitability of a distribution can be examined via statistical indexes, to perform the computational analysis.

The mean bias error (MBE), the mean absolute error (MAE), the root mean square error (RMSE), its systematic (RMSE_s) and

unsystematic (RMSE_u) components and the index of agreement (d) are those that are calculated in this study (7)-(12).

MBE defines whether a model over- or underpredicts the observations.

MAE and RMSE are measures of the total deviation of the predicted values from the observed values and illustrate the presence of significant mispredictions, whereas d indicates the degree to which the predictions of a model are error free.

The RMSE and MBE provide information about the short- and long-term performance of a model, respectively.

A drawback of RMSE is that few large errors in the sum may produce a significant increase in its value, whereas a drawback of MBE means that an overestimation in one observation may be canceled by an underestimation in the other.

In (7)-(12), P and O represent the predicted and the observed values respectively, whereas the overbar indicates mean values.

$$MBE = \bar{P} - \bar{O} \quad (7)$$

$$MAE = \frac{1}{N} \sum_{i=1}^N |P_i - O_i| \quad (8)$$

$$RMSE = \left[\frac{1}{N} \sum_{i=1}^N (P_i - O_i)^2 \right]^{\frac{1}{2}} \quad (9)$$

$$RMSE_s = \left[\frac{1}{N} \sum_{i=1}^N (P_i^* - O_i)^2 \right]^{\frac{1}{2}} \quad (10)$$

$$RMSE_u = \left[\frac{1}{N} \sum_{i=1}^N (P_i^* - P_i)^2 \right]^{\frac{1}{2}} \quad (11)$$

$$d = 1 - \frac{\sum_{i=1}^N (P_i - O_i)^2}{\sum_{i=1}^N (|P_i - \bar{O}| + |O_i - \bar{O}|)^2} \quad (12)$$

$$t = \left[\frac{(N-1)MBE^2}{RMSE^2 - MBE^2} \right]^{\frac{1}{2}} \quad (13)$$

In (10) and (11), P_i^* is calculated as $a + b \cdot O_i$, where a and b are the intercept and the slope of the least squares line between the predicted and the observed values, whereas N is the number of concentration classes in which the data are divided.

The reduction R in current emissions to meet air quality standards is often calculated by “the rollback equation” (14), where $E\{C\}$ is the current annual mean value of the pollutant's concentration, $E\{C\}_s$ is the annual mean value corresponding to air quality standards and C_b is the background concentration [5].

The mean concentration for log-normal distribution can be calculated with (15).

$$R = \frac{(E\{C\} - E\{C\}_s)}{(E\{C\} - C_b)} \quad (14)$$

$$\ln E\{C\} = \ln \mu_g + \frac{1}{2} \ln^2 \sigma_g \quad (15)$$

When the parent probability distribution of air pollutants is properly chosen, this specific distribution can be used to estimate the mean concentration, the number of times when the air quality standards is exceeded and the emission sources reduction, to meet the air quality standards.

Therefore, a piece of information regarding the frequency distribution is necessary for developing air pollutant control strategies.

In order to estimate the emission sources

reduction of a pollutant by applying the rollback equation, one needs to know the type of the distribution that fits better to the pollutant concentration.

However, some errors would result in calculations, as rollback equation can be only applied when assuming that the spatial distribution of emission sources, the meteorological conditions and the background concentration of the pollutant remain unchanged and also that the pollutant is not reactive.

VII. RESULTS AND DISCUSSION

FIG. 1 presents the PM10 concentration values recorded at the selected station between 2008 and 2010; these data are used to perform a statistical modeling, in order to confirm the fact that the log-normal distribution is appropriate for describing the percentage of appearance for each range of concentration.

Table I presents the statistics of the observed values of the PM10 concentration in Dolj County during the period 2008–2010.

It shows that the number of times when the value of PM10 concentration exceeded the limit value of 50 µg/m³ imposed by the European Union is 14.5% from total available data.

Exceeding the value 120 µg/m³, imposed as maximum admitted eight hour average, occurred in just 0.1% cases (taking into account only the times when an exceeding automatically recorded value is gravimetrically confirmed by experts from Craiova Regional Environment Protection Agency and therefore validated by them).

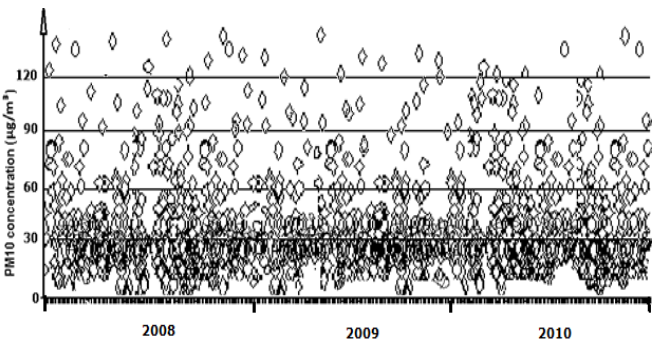


Fig.1. Temporal variability of PM10 concentration values recorded at the selected station recorded between 2008 and 2010

TABLE I
STATISTICS OF THE MAIN PARAMETERS REGARDING
PM₁₀ LEVEL
IN DOLJ COUNTY BETWEEN 2008 AND 2010

Data availability (days)	1083
Data availability (values)	25992
Number of times when the value of PM10 concentration exceeded the limit value of 50 µg/m ³	3774
Number of times when the value of PM10 concentration exceeded the limit value of 120 µg/m ³	26

Minimum value ($\mu\text{g}/\text{m}^3$)	7	
Maximum value ($\mu\text{g}/\text{m}^3$)	150	
Most frequently occurring value ($\mu\text{g}/\text{m}^3$)	23	
Mean value ($\mu\text{g}/\text{m}^3$)	32.68	
AA value of PM10 concentration, μg ($\mu\text{g}/\text{m}^3$)	2008	34.55
	2009	32.20
	2010	31.29
standard	2008	1.42
geometric deviation, σ_g	2009	1.44
	2010	1.53

The PM10 concentration average exhibits a seasonal variability. It tends to be higher in winter and lower in summer, a fact that could be explained by the seasonal pattern of the PM10 emissions due to the central heating operation in winter and also to the seasonal variation of the atmospheric boundary layer height, which is generally lower during winter and becomes higher during summer.

VIII. ACCORDANCE BETWEEN PARENT AND OBSERVED DISTRIBUTION

As in earlier studies [4], [9], the log-normal distribution best approximates the distribution of the PM10 concentration values.

Table II presents this accordance in figures.

TABLE II
ACCORDANCE BETWEEN PARENT AND OBSERVED DISTRIBUTION

No.		Real perce nt	Calcd. perce nt	Diff.*
1	0-10	5	5.5	+0.5
2	10-20	20	20	0
3	20-30	25	23	-2
4	30-40	23	21	-2
5	40-50	12.5	15	+2.5
6	50-60	7.5	6.5	-1
7	60-70	2.8	4.2	+1.4
8	70-80	2	2.3	+0.3
9	80-90	1.2	1.5	+0.3
10	90-100	0.42	0.4	-0.02
11	100-110	0.31	0.33	+0.02
12	110-120	0.17	0.2	+0.03
13	120-130	0.05	0.04	-0.01
14	130-140	0.03	0.02	-0.01
15	140-150	0.02	0.01	-0.01
avg.	0-150	32.68	33.48	0

* Difference between calculated and real percentage

Fig. 2 shows the log-normal distribution to fit very well to the distribution of the PM10 concentration values

The goodness-of-fit is verified by calculating the statistical indicators that were previously presented. The mean of the observed values is identical to the mean of the predicted values ($O = P$).

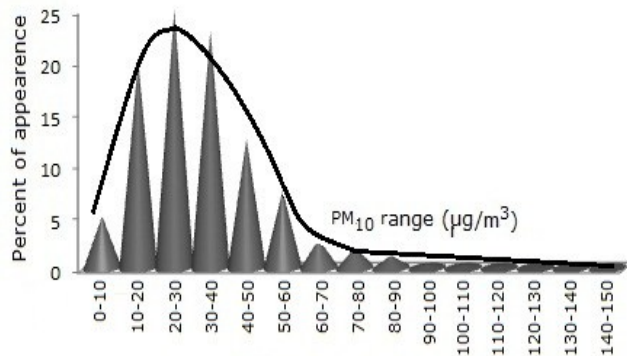


Fig.2. The log-normal distribution fits very good the real data

Consequently, as MBE is practically zero (its value is about 10^{-6}), the frequencies that are predicted by the log-normal distribution are neither overestimated nor underestimated, when compared to the observed ones, as seen in Table II.

MAE and RMSE are found to be $5 \cdot 10^{-3}$ and $8 \cdot 10^{-3}$, and their very low values reveal that the log-normal distribution is capable to adequately simulate the experimental data.

The systematic and unsystematic components of RMSE are equal to $9 \cdot 10^{-4}$ and $8 \cdot 10^{-3}$, respectively.

The fact that the systematic component of RMSE approaches zero and the unsystematic component approaches RMSE indicates that the log-normal distribution fits well to the observed frequency distribution.

This is also verified by the index of agreement that equals to 0.99, a value that is very close to the unit.

Moreover, the t test indicates that the observed and the predicted values are not statistically different at a 99% confidence level.

The values of μ_g and σ_g are displayed in Table 1.

As it can be seen, the value of σ_g , that represents the fluctuation of the distribution, remains almost steady during the examined period, this fact indicating that stochastic effects like meteorological conditions are similar [10].

Fig. 2 also reveals that the frequency distribution of the PM10 concentration is unimodal.

IX. CONCLUSION

In this study, the log-normal distribution is applied in order to describe the observed distribution of PM10 levels in Dolj County (south-western Romania).

The major conclusion is that the PM10 levels (slightly higher compared to the air quality standards imposed by the European Union) are very good fitted by a log-normal distribution, which also is valuable as a prediction.

The study provides useful information towards the air pollution control strategies.

X. REFERENCES

- [1] V. S. Patankar, "Numerical heat transfer and fluid flow," New York: McGraw Hill Book Company, 1980.
- [2] D. M. Moreira, M. T. Vilhena, D. Buske and T. Tirabassi, "The GILTT solution of the advection – diffusion equation for an inhomogeneous and nonstationary PBL," *Atmospheric Environment*, vol. 40, pp. 3186-3194, 2006.
- [3] S. Baroutian, A. Mohebbi, and A. Soltani

- Goharrizi, "Measuring and modeling particulate dispersion: A case study of Kerman Cement Plant," *J. Hazardous Materials*, vol. 136, pp. 468-474, 2006.
- [4] M. Crawford, "*Air pollution control theory*," New York: McGraw - Hill Book Company, 1976.
- [5] E. Canepa, and C. F. Ratto, "Algorithms to simulate the transport of pollutant elements: A model validation exercise and sensitivity analysis", *Environmental Modeling and Software*, vol.18, pp. 365-372, 2003.
- [6] M. Delgado, J. L. Verdegay and M. Vila, "A general model for fuzzy linear programming," *Fuzzy Sets Syst.*, vol. 29, pp. 21-29, 1989.
- [7] D. Dubios and H. Prade, "Qualitative possibility theory and its applications to constraint satisfaction and decision under uncertainty," *Int. J. Intell. Syst.*, vol. 17, no. 1, pp. 45-53, 2002.
- [8] G. H. Huang and N. B. Chang, "The perspectives of environmental informatics and systems analysis," *J. Environ. Inform.*, vol. 1, pp. 1-6, 2003.
- [9] L. Liu, G. H. Huang, Y. Liu and G. A. Fuller, "A fuzzy-stochastic robust programming model for regional air quality management under uncertainty," *Eng. Optim.*, vol. 35, no. 2, pp. 177-199, 2003.
- [10] V. Guttal and C. Jayaprakash, "Spatial variance and spatial skewness: leading indicators of regime shifts in spatial ecological systems," *Theor. Ecol.*, vol. 11, pp. 450-460, 2009.

Context Aware Recommendation of Location-based Data

Karol Waga, Andrei Tabarcea, Pasi Fränti

Speech & Image Processing Unit, School of Computing, University of Eastern Finland, Joensuu

Abstract—There is much information available, but the problem is how to find which is relevant. We present a context aware recommendation system, which recommends relevant location-based data. We study its performance within MOPSI service that includes fixed form maintained database and free form user collection.

Keywords—Recommendation, relevance, context-aware, location, content, network, time, location-based applications

I. INTRODUCTION

Recommendation systems are important research and are in scope of interest of both universities and companies [1]. Recommendation systems produce personalized search results by performing analysis of user actions [3]. Such systems can be used, for example, for recommending similar products in online stores, music or videos which may be of interest of particular user, and advertisements targeted to specific audience. Recommendation system takes into account additional information about user, which is called context. Examples of contexts we identified are user's location (distance to the service), identity (age, gender, hobbies and language), social network, history of activities, time, technical resources (network accessibility, bandwidth), and the purpose of use (work, leisure time).

Location is very important attribute of our data. Mobile technology is increasing its popularity and availability and it allows collecting location data [9]. Furthermore, mobile phones are one of the main devices for information access [19]. Because of technical limitation, such as bandwidth and screen size, recommendation system can be used to reduce amount of information presented to user. Recommendation system can consider user's location for recommending the nearest service (ATM, restaurants, pubs, tourist sights and social events). However, the nearest service may not be the most interesting for user and other factors should also be considered.

Our goal is to design a context-aware recommendation system based on the four aspects of relevance (content, time, location and social network) discussed in [7]. For recommendation we use the MOPSI services¹ geo-tagged database, which contains user-generated photo collection and service database. Our solution is designed and implemented as a prototype solution within MOPSI (see Fig. 1), as a case study. The MOPSI project implements various location-based services and applications such as mobile search engines, data collection, user tracking and route

recording. It has applications integrated both on web and in mobile phones.

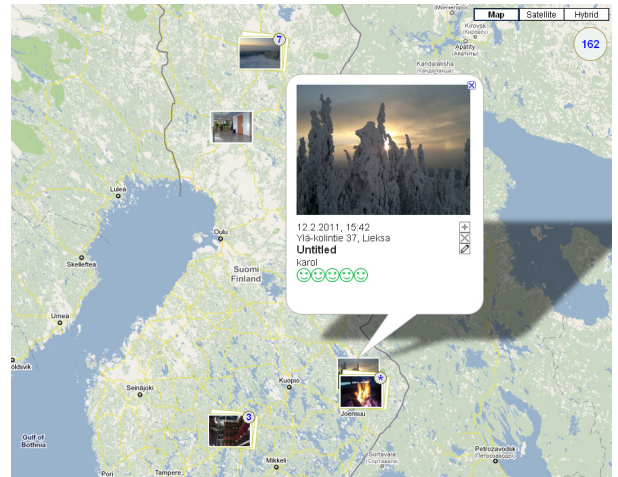


Figure 1: User generated data collection in MOPSI.

II. RELATED WORK

Content-based multimedia information retrieval has been described in [16]. Exploiting synergy between the various media, including text and context information is described as one of the major challenges for the future. Our recommendation system has the goal of combining textual, context and multimedia information in order to get relevant results.

Recommendation systems produce personalized search results relying on a variety of contextual information. Main approaches are *collaborative filtering*, *content-based filtering* and *knowledge-based recommendations*. We aim at as little user interaction as possible during the recommendation process, and because our data is mostly user-generated, collaborative filtering is the most suitable approach in our case. According to [6], collaborative filtering is not sufficient in modern recommendation system because of limited contextualization; therefore we use hybrid approach, similarly as in [4]. The advantage of using hybrid recommendation when applying for recommending web pages was emphasized also in [10].

An intelligent multimedia browsing system was described in [2] by combining the use of patterns with low-level features. Their main contribution is using the browsing behavior of users combined with features of objects to provide recommendations for content-based multimedia retrieval. We also monitor behavior of users (recent searches, searches in the particular area) for better

¹ <http://cs.joensuu.fi/mopsi>

recommendations. In specific, our photo recommendation problem is similar in a sense that we also recommend from a multimedia database (photo, descriptions, locations). Although a multimedia database is used in both cases, the main difference is that we use location and our recommendation is mostly location-oriented. Furthermore, they use only the previous behavior of the current session (there is no explicit login), whilst we store previous behavior history for every user.

Most of the existing recommendation systems are tested using restaurants and tourist targets. Our system aims to be more general and can be summarized by the phrase “What is interesting in the area?”. In [17], a map-based and context-based personalized recommendation system was proposed using Bayesian Networks, which is applied to restaurant recommendation. This is similar to our service, but we do not use Bayesian networks because of lack of flexibility and the user does not explicitly input his profile and preferences.

The method in [22] has two part recommendation system which contains the location-based data (traffic, road-condition, map, service information) and value-added data provided by users (ratings, comments, blogs and tags). They focus on how to gather value-added data provided by users and how to extract it from Web 2.0 websites (ratings sites, comments, blogs). We extract our value-added data from explicit user ratings and from user previous behavior.

Browsing data and image semantics is used in [14] for recommending purchasable items. One of our future goals is to use image semantics as an extra criterion in our photo recommendation. In [20], a recommendation system of shops based on users’ past location history was described. The frequency of shop visits by users are the input of an item-based collaborative filtering algorithm and the predictions are narrowed down based on user movement and geographical conditions in the city. Our MOPSI system also tracks down user movement (explicitly, on user request) and our future goal is to use this tracking data in determining the area of interest of the user and assigning extra relevance score to the items in the database that are frequently visited or photographed by other users. Reference [11] shows a good example on how to effectively use social network in order to provide better recommendation by defining and implementing a trust-based model. The key concept is trust between users and two trust metrics are defined: *personal trust* and *person-group trust*. We also use the social network as a relevance criterion, although the current solution is still in a very early stage.

Location-based recommendation systems using mobile devices have been described in [12], [15], [13], [21], who tested it either for restaurant or tourist recommendation using various hybrid techniques and contextual information.

Reference [5] has similar motive to ours by proposing a system to suggest tourist destinations based on visual matching and minimal user input. The input is a photo of desired scenery or a keyword describing a place of interest and the output is a place that shares the visual

characteristics. The system uses a large-scale geo-tagged web photo collection (Flickr). The queries are compared against representative tags or representative images in order to discover interesting tourist destinations. Their main contributions are: an efficient clustering of the geo-tagged image database, proposal of using representative images for having faster retrieval, and designing a flexible interface to allow user to enter either keyword (such as architecture, beach, flower.) or image to describe its interest. Although our recommendation is more general, there main similarity is that we also use a geo-tagged image database as input. In addition, our system can also take keywords as input, although this uses a search algorithm. However, this is not the focus of this paper and has been documented in [8].

III. SYSTEM DESCRIPTION

In this section, we provide description of what our system actually recommends and how it uses the four main aspects of relevance as the context.

A. What is recommended

Our aim in MOPSI is to recommend interesting places in user’s surrounding. In the service, we have two databases that are used for the recommendation (see Fig.2). First database contains trusted services verified by administrators. These services represent variety of categories from restaurants, bars, and cafeterias, through grocery stores, pharmacies, and ATM machines, to car repairs, and museums. Service data include location, contact information, and relevant keywords.

Second database contains photos users have taken using mobile phones and uploaded real-time with several related information, such as location, time, and description. Both photos and services (referred as *items* from now onwards) are rated by users. Moreover, photos can picture any place, which is found interesting by the user.

B. Recommendation methods

In our recommendation system, we give personalized recommendations by combining various paradigms of recommendation systems. We combine collaborative filtering with information about user profile and context.

Having these two sources for recommendation, the challenge is how to select the most relevant items to users. First we define context for each recommendation request. In our previous work we identified four aspects of relevance: location, content, time, and network [7]. Location is physical place of the user represented by geographical coordinates (latitude and longitude). Content in MOPSI is determined currently based on the description of the photos and keywords attached with the services. Time is considered only for photos and measures age of photo and the season (of the year) when photo was taken. Network is utilized via ratings given by other users to items and it constitutes an integral part of the system based on collaborative filtering. Considering these context in mind, we create profile for each user of MOPSI.

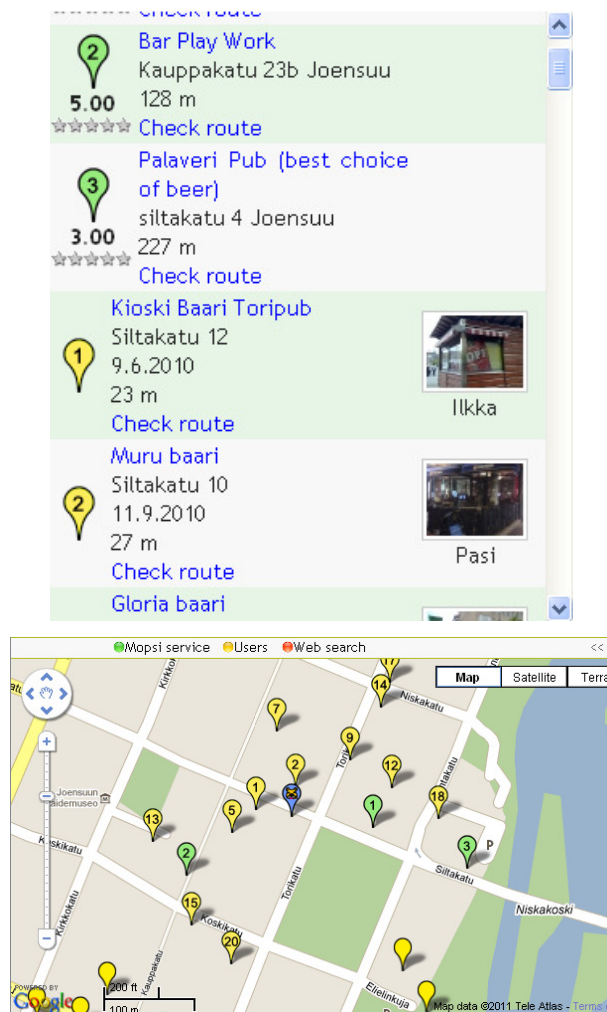


Figure 2: Data from both databases visualized as a list (top) and on map (bottom) with user position marked with blue bubble

User profile contains user behavioral data, such as location and previous usage of data, i.e. how user interacted with the system. Currently this is measured by the keywords user has performed earlier searches and visited locations.

C. System description

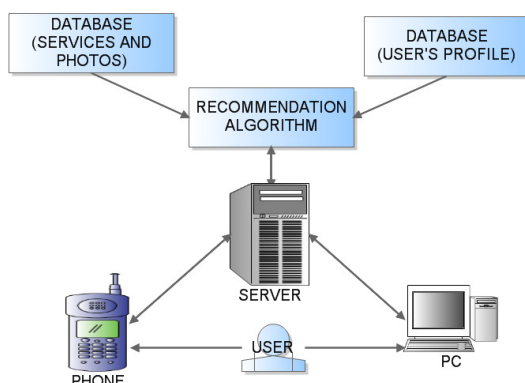


Figure 3: System diagram

Users can access the recommendation system through MOPSI website and mobile application. Mobile access to

the service is important, since this is the most natural environment where the system can be most beneficial in real life. It is key functionality of the system that user can ask for recommendation in any location when he can easily visit the suggested places immediately. When recommendation request is sent to server, after computing, list of recommended items is returned to user as illustrated in Fig. 3.

D. System implementation

In this section, we describe in details how we implement the system. Summary of the algorithm is followed by description of the scoring function used by the algorithm.

1) Recommendation algorithm

Recommendation algorithm consists of three major steps. Its input is the username and location of the user. The steps of the recommendation algorithm are outlined in Fig. 4. First step is to select potential items that are to be considered. We use location as the criterion for this pre-selection. Items that are far from user are considered irrelevant and are skipped already at this stage. The selected items are then scored in the second step. The scoring function will be described in details in Section IV.

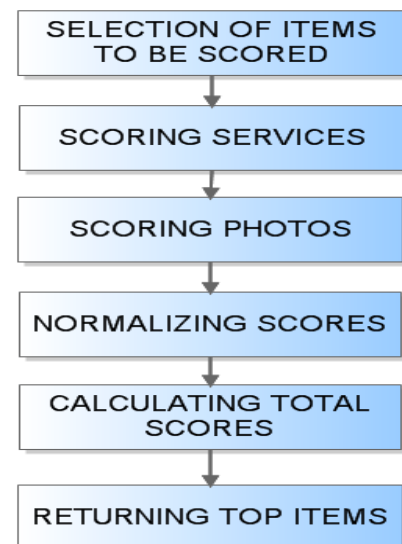


Figure 4: Scoring function schema

Third step is to prepare final list of the recommended items ordered according to the scores received in the second step. The system outputs the final recommendation list, which consists of 20 items of the highest scores.

2) User interface

User interface is provided both for the MOPSI mobile application and the website. In web, the recommendation function is embedded on the MOPSI main page where user can request recommendation by pressing a single button (see Fig. 5). The results are visualized on screen in two ways. On the left, there is scroll list of the recommended items including title, street address and distance. Photo results include the description user has entered (if available), street address, date, distance, photo thumbnail and author. All the

results are marked on the map visible on right side of the screen. Services are marked with green bubbles and photos with yellow bubbles. User location is marked by blue bubble with “Mopsi dog” icon.

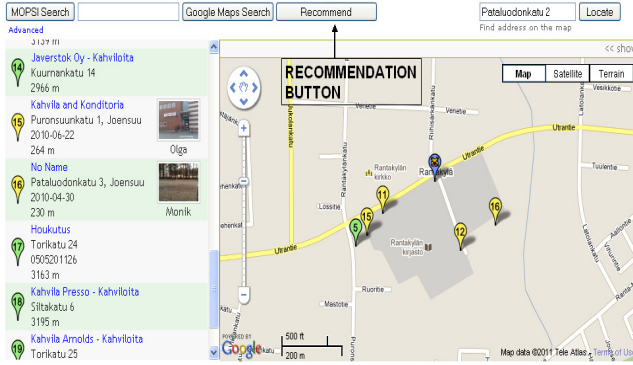


Figure 5: Web interface for the recommendation system.

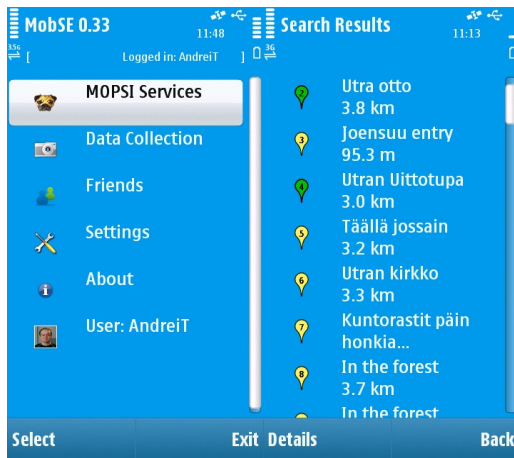


Figure 6: Mobile interface for the recommendation system. MOPSI Services screen in main menu on the left. Recommendation results list on the right.

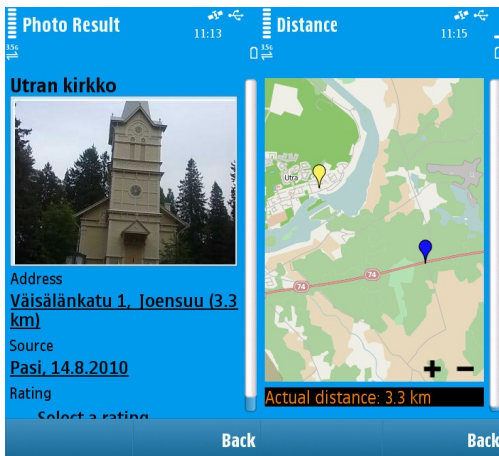


Figure 7: Recommendation result in mobile application. Details of the item are shown on the left. Item location on map is shown on the right.

In mobile application, the recommendation function is embedded in MOPSI Services screen (see Fig. 6) where user can request recommendation by pressing a single button. Application will show list of results including name,

distance and type (service or photo) of each item (see Fig. 6). It is possible to see details of every result by selecting it on the list. Details of service results include title, street address, distance and list of keywords. Photo results include description user has entered (if available), photo thumbnail, street address, distance, author and date (see Fig. 7). By clicking on the address field, user can see map with item and its location marked.

IV. SCORING FUNCTION

Services are scored using contextual information about search history, location, and explicit rating. Photos are scored based on the same three factors and also on time. These factors are discussed next. We explicitly use two of the four aspects of relevance (location and time), whilst the other two factors (search history and rating) combine content, social network and time.

A. Search history

We define two search histories that are based on previous user behavior: *general* and *user-specific*. For services we take into consideration both the service name and the associated keywords, and for photos, we use the description which is assigned by the user.

The *general* history records keywords used for searches by all MOPSI users. It is used in three ways. Firstly, to check if any of the keywords associated to the service in question has been searched in nearby locations (S_N). Secondly, to check if any of the keywords has been searched recently (S_S). Thirdly, to check if the keyword has high frequency within all search requests (S_F).

User specific history records keywords that a given user has been used before for searches (S_U). Keywords of services and photos that are found in the history list are given 3 points each. For example, let us consider a service with keywords *café* and *restaurant*, and a user who has searched for *restaurant*, *bar*, *café* and *sauna* in the past. In total, this service gets 6 points for user specific history since two matched keywords were found.

Total score of search history consists of the following components:

$$S_H = S_N + S_S + S_F + S_U \quad (1)$$

where S_N , S_S , S_F , and S_U are the raw counts for keyword matches in nearby locations, within recent time, frequency of keywords in search history, and searches done by current user, respectively.

B. Location

We calculate the distances between each recommendation item and the user's location and define it as location score. By use of distance, we introduce location relevance aspect to the system.

C. Rating

Users can rate photo and services through the web or mobile interface. Services in MOPSI database have been

rated by users in scale of 0 to 5 and the rating for photos is cumulative, using a thumbs up/thumbs down system (e.g. a photo liked by 5 users and disliked by 2 has a score of 3). The average score, in the case of services and the total score, in the case of photos, represents the rating score.

D. Time

Time is also a very important aspect of relevance. Photo relevance decreases with time, as the places or views capture by user may suffer changes over the years. Also, the season when the photo is taken is very important for the relevance, as winter activities, for example, cannot be recommended during summer.

More recent photos in user collection are considered more relevant than old ones and the newer the photo is, the higher score it receives. Additional difference is that the score is also influenced by time of the year when photo was taken.

We define following time thresholds (points given in brackets): 1 week (10), 1 month (7), and 1 year (4). Secondly, photos are classified into one of the four seasons of the year (winter, spring, summer, autumn). If the recommendation request is performed during the same season as the photo was taken, it is given additional 10 points. Thus, total score based on time is for each photo calculated as follows:

$$S_T = S_A + S_Y \quad (2)$$

For example, photos that were taken 4 days ago, and in the same day are scored $S_A=10$. The photo was also taken in the same time of year thus $S_Y=10$. By use of time for scoring photos, we introduce time relevance aspect to our system.

E. Total score

All the above scores are normalized to the scale [0..1] using the following formula:

$$N = \frac{S - \text{MIN}(S)}{\text{MAX}(S) - \text{MIN}(S)} \quad (3)$$

where S is the raw score, N is the normalized score, , and $\text{MIN}(S)$ and $\text{MAX}(S)$ are the minimum and maximum scores for each of the criterion respectively.

Final score of each service is then calculated using the following formula:

$$S_{\text{SERVICE}} = w_H N_H + w_L N_L + w_R N_R + 1 \quad (4)$$

where N_H stands for the normalized score for search history, N_L for location, and N_R for rating; w_H , w_L and w_R are weights for the corresponding scores. A constant of one point is added in order to promote services for recommendation, because they are assumed to originate from a trusted source and therefore more relevant than older photos from user collection. The location score is multiplied by two emphasize nearby locations.

Final score of each photo item is calculated in the same way as services, having an additional time score:

$$S_{\text{PHOTO}} = w_H N_H + w_L N_L + w_R N_R + w_T N_T \quad (5)$$

where N_T stands for time score and w_T is the weight for

the time score.

Currently all weights are set to 1 except for the location weight which is set to 2 in order to emphasize importance of location.

V. PRELIMINARY EXPERIMENTS

As stated by [18] and [2] evaluation of user satisfaction is important part of the system evaluation. Moreover, evaluating user subjective opinion towards the system helps to improve various aspects of the system [18]. We conducted several experiments where we collected subjective opinions on recommended items. In the tests we have chosen several locations in Joensuu and analyzed usefulness of the recommendation there. Tests have been limited to Joensuu area, because user data collection is richest and most diversified in the town. There were locations with different characteristics selected. We checked whether the recommendation is relevant for different type of users such as tourists and locals. We also evaluated services and photos that were scored but not recommended in order to find out what relevant results may have been overlooked by the implemented scoring function.

Recommendations in downtown always give many cafeterias, pizzerias and bars taken from service database. In addition, the system provides additional recommendations such as sport places and shops taken from photo collection. Experiments have shown that all factors have impact on recommendation results. For example, in Papinkatu 3 (area with many block of flats and services nearby) there are many eating places recommended, but they are chosen based on rating and search history. Same example show well that our recommendation system chooses relevant photos from the collection, such as shop, kiosk and mailbox, whilst general photos of streets, houses and people are skipped, although located nearby.

In other districts of Joensuu service collection is smaller, and more of the recommendation results are photos. In some locations, such as Latolankatu 12, which is mostly a residential area, many photos in the very near proximity are selected although their content completely lacks relevance to most users. These photos are chosen due to lack of enough services and better photos available at the area. It also revealed that the location was given too much weight in this particular case, and it should be redefined more adaptively in future.

In other test cases, recommendations outside downtown area are mostly useful. For instance in Utrantie 75 (residential district with single houses and recreational areas in forest and on riverside) many recreational areas are found and recommended. Best bars from center of Joensuu are recommended in addition to the local bars and cafeterias nearby. This example also revealed that direct distance calculation is not sufficient as some results across the river are included despite there is no bridge nearby in this area to access these recommended services. Using navigational distance based on road network could help in this case. This

example demonstrated the relevance of seasonal time score as photos taken in spring one year ago were selected instead of winter time photos taken 3 months ago.

VI. DISCUSSION AND CONCLUSIONS

In conclusion, our system gives useful recommendation and selects relevant items. There some exceptions where behavior of the system is not satisfactory. However, the system fails to give useful recommendation in specific, untypical cases, for example when user generated photo collection is very dense and limited to test photos with useless content or when there is no information about particular area in our data collection.

Although we already have useful results, there is still room for further improvements. First, user profile can be extended to include other factors besides the search history. In Facebook, people reveal personal data such as age and gender, which can be used for additional factors. For example, age can influence the choice of restaurant. Young people usually prefer fast food and older ones more expensive restaurants. Moreover, Facebook user's profile contains information about hobbies and favorite types of food. For example, people who like pizza should be given recommendation of nearby pizzerias instead of Chinese restaurants. Therefore, connection of user's profile in MOPSI with user profile in Facebook would give potentially much more information to build user profile.

Scoring function can also be improved further. Selection of items to be scored is the first area of improvement. Current solution with static distance threshold and maximum limit of items to be scored cannot adapt between areas with extremely low or extremely high density of items available in MOPSI data collection. In case of low density, there is small number of items scored (sometimes no items at all). In case of high density, some relevant items may not be chosen for scoring at all as happened with the Latolankatu example.

Furthermore, with growing number of items available in MOPSI we may need to increase speed of the recommendation algorithm. When MOPSI account will be integrated with Facebook, user's profile need to be systematically updated based on Facebook profiles. Weights for the scores should be adjusted in future, because experiments suggested that the importance of location was emphasized too much in some cases.

Calculation of time score should also be revised, however, it has to be done so that it preserves current behavior of the system in cases when time score improves recommendation results. Promoting services over user collection photos by using constant one is not always beneficial. It is worth to consider time score for services so that bars get more points in the evening, while lunch places in lunch time.

REFERENCES

- [1] Adomavicius, G. & Tuzhilin, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transaction on Knowledge and Data Engineering* 17(6), 734-749, June 2005.
- [2] Albanese, M., Chianes, A., d'Acerno, A., Moscato, V., Picariello, A. A multimedia recommender integrating object features and user behavior. *Multimedia Tools and Applications*, 50(3), 563-585. December 2010.
- [3] Birukov, E. Blanzieri, E. & Giorgini, P. 2005. Implicit: An agent-based recommendation system for web search. *International Conference on Autonomous Agents and Multi-Agent Systems*. Utrecht, The Netherlands, 2005.
- [4] Burke, R. Hybrid Recommender Systems: Survey and Experiments. *User Modelling and User-Adapted Interaction*, 12(4), November 2002.
- [5] Cao, L., Luo, J., Gallagher, A. A Worldwide Tourist Recommendation System Based on Geotagged Web Photos. *IEEE International Conference on Acoustics Speech and Signal Processing*. Dallas, Texas, USA, 2010
- [6] Ducheneaut, N., Partridge, K., Huang, Q. Collaborative Filtering Is Not Enough? Experiments with a Mixed-Model Recommender for Leisure Activities. *International Conference on User Modeling, Adaptation, and Personalization*. Trento, Italy, 2009.
- [7] Fränti, P., Chen, J., Tabarcea, A. Four Aspects of Relevance in Sharing Location-based Media: Content, Time, Location and Network. *International Conference on Web Information Systems and Technologies*. Noordwijkerhout, The Netherlands, May 2011.
- [8] Fränti, P., Tabarcea, A., Kuittinen, J., Hautamäki, V. Location-based search engine for multimedia phones. *IEEE International Conference on Multimedia & Expo*. Singapore, July 2010.
- [9] Ge, Y., Xiong, H., Tuzhilin, A., Xiao, K., Gruteser, M. & Pazzani, M. J. 2010. An Energy-Efficient Mobile Recommender System. *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Washington, DC, July 2010.
- [10] Göksedef, M., Sule G.-Ö. Combination of Web page recommender systems. *Expert Systems with Applications*, 37(4), April 2010.
- [11] Hinze, A., Quan, Q. Trust- and Location-Based Recommendations for Tourism. *Confederated International Conferences*. Vilamoura, Portugal, 2009.
- [12] Horozov, T., Narasimhan, N., Vasudevan, V. Using location for personalized POI recommendation in mobile environments. *International Symposium on Applications on Internet*. 2006.
- [13] Kang, E.-Y., Kim, H., Cho, J. Personalization Method for Tourist Point of Interest (POI) Recommendation. *Knowledge-Based Intelligent Information and Engineering Systems*. 2006.
- [14] Kwon, O. B. "I know what you need to buy": context-aware multimedia-based recommendation system. *Expert Systems with Applications*, 25(3), 387-400. October 2003.
- [15] Lee, B.-H., Kim, H.-N., Jung, J.-G., Jo, G. Location-based Service with Context Data for a Restaurant Recommendation. 2006.
- [16] Lew, M. S., Sebe, N., Djeraba, Ch., Jain, R. Content-Based Multimedia Information Retrieval: State of the Art and Challenges. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2(1). February 2006.
- [17] Park, M.-H., Hong J.-H., Cho, S.-B. Location-based Recommender System Using Bayesian User's Preference Model in Mobile Devices. *Ubiquitous Intelligence and Computing*, 1130-1139. 2007.
- [18] Pu, P., Chen, L. A User-Centric Evaluation Framework of Recommender Systems. *ACM Conference on Recommender Systems*. Barcelona, Spain, September 2010.
- [19] Ricci, F., Mobile recommender systems, *Information Technology & Tourism*, Vol 12, No. 3, p. 205-231. 2011
- [20] Takeuchi, Y., Sugimoto, M. CityVoyager: An Outdoor Recommendation System Based on User Location History. *Ubiquitous Intelligence and Computing*, 4159, 625-636. 2006.
- [21] Tung, H.-W., Soo, V.-W. A Personalized Restaurant Recommender Agent Service for Mobile E-Service. *IEEE International Conference on e-Technology, e-Commerce and e-Service*. Taiwan, 2004.
- [22] Yang, F., Wang Z.-M. A Mobile Location-based Information Recommendation System Based on GPS and WEB2.0 Services. *WSEAS Transactions on Computers*, 8(4), April 2009.

ELLEIEC Project: Network of Networks in Electrical and Information Engineering - an European Area of Education for New Skills for New Jobs

Hamed Yahoui, Dorin Popescu, Jean-Marc Thiriet, Olivier Bonnaud, Tony Ward, Cristian Perra

Abstract— The paper is dealing with present Lifelong Learning in Europe through the current state of ELLEIEC European project (Enhancing Lifelong Learning for the Electrical and Information Engineering Community) [1]. ELLEIEC project is the follow-up of EIE-Surveyor TN which aims at acting both as an observatory and a proposition force for the evolution in the interdisciplinary fields of Electrical and Information Engineering (EIE), of engineering curricula through the implementation of the Bologna process [2]-[3]. The first objective of the ELLEIEC project is to develop a virtual European Centre for Entrepreneurship Education in Electrical and Information Engineering with an internal e-learning assessment offer which will be a reference point for any applicant in the Lifelong Learning framework. The second objective is to study through an experiment and a survey of the mutual recognition process to gather information on good practice in the design of international cooperation at both master and bachelor levels. At the same time, a study will be undertaken across different European countries on how to incorporate the accreditation of prior Lifelong Learning into the PhD process which will allow engineers within a company to move back into education to continue their studies and prepare and validate a doctoral thesis. This benchmark will be used to analyse the recognition systems required to ensure the appropriate credit transfer for prior informal learning.

I. INTRODUCTION

IN a close future, HEI (High Education Institution) in Europe will have to challenge the globalization of Education Area if one thinks of attractiveness of the choice of learning and mobility within Europe. This will be achieved by a new university model keeping its innovation competitiveness at high level and by proposing new tools to

fit the job market needs in order to decrease the high unemployment level. A main step will be to adapt existing classical education offer to a new model enhancing private and public partnerships with strong university/industry cooperation. Europe has the chance to be constituted of a various set of educational systems with cultural diversity. To follow these objectives after the Bologna process, Europe should harmonize its Education Area considering each added value. ELLEIEC has been working actively to contribute to this goal in the Electrical and Information Engineering fields thanks to the expertise provided by all partners and the new proposed concept of ICN and VCE. The work aim for the ICN is to propose a strategy in building a harmonized education offer for the European Electrical and Information Engineering sector. It is mainly based on the advertising of a common referenced courses database enhancing Lifelong Learning and ECTS recognition issue based on tools of the Bologna process structure and EQF framework.

II. PROJECT OBJECTIVES

For an educational institution, giving recognition and appropriate ECTS credits for an intensive training or a web based course, which uses its own internal pedagogical resources, is a common occurrence. However, with an Erasmus mobility agreement, because external learning resources are involved the validation and ECTS credit assignment process can result in a much longer pedagogical and administrative process. It should also be noted that in this case, it is the sending institution that determines the number of ECTS credits awarded and records the result for its own degree. The Erasmus process better works when after a long shared experience for formal learning a strong partnership is developed between two institutions. With Lifelong Learning the recognition problem is more concerned by accrediting non-formal learning and informal knowledge gained from external sources. With the Lifelong Learning programme, Europe has shown the importance of providing a framework for the identification and validation of these ‘invisible’ learning outcomes. Arising from this initiative, a large number of European projects have been launched in the two following topics:

- Inventory of validation of non-formal and informal learning has been produced with Cedefop, with a detailed survey of developments in European countries,
- European Qualification Framework for Lifelong Learning

Manuscript received September 1, 2011. This work was supported by the European Commission in the Life long Learning Programme through the ELLEIEC project (Number: 142814-LLP-1-2008-FR-ERASMUS-ENW, Instrument type: ERASMUS NETWORK).

Hamed Yahoui is with the Université de Lyon I, France (e-mail: hamed.yahoui@univ-lyon1.fr).

Dorin Popescu is with University of Craiova, 107 Decebal Blvd, Craiova, Romania. (corresponding author, phone: +40-251-436999; fax: +40-251-436999; e-mail: dorin@robotics.ucv.ro).

Jean-Marc Thiriet is with GIPSA-Lab, Université Joseph Fourier, Grenoble, France (e-mail: jean-marc.thiriet@ujf-grenoble.fr).

Olivier Bonnaud is with Université de Rennes I, France (e-mail: olivier.bonnaud@univ-rennes1.fr).

Tony Ward is with University of York, United Kingdom (e-mail: aew6@york.ac.uk).

Cristian Perra is with University of Cagliari, Italy (e-mail: cperra@diee.unica.it).

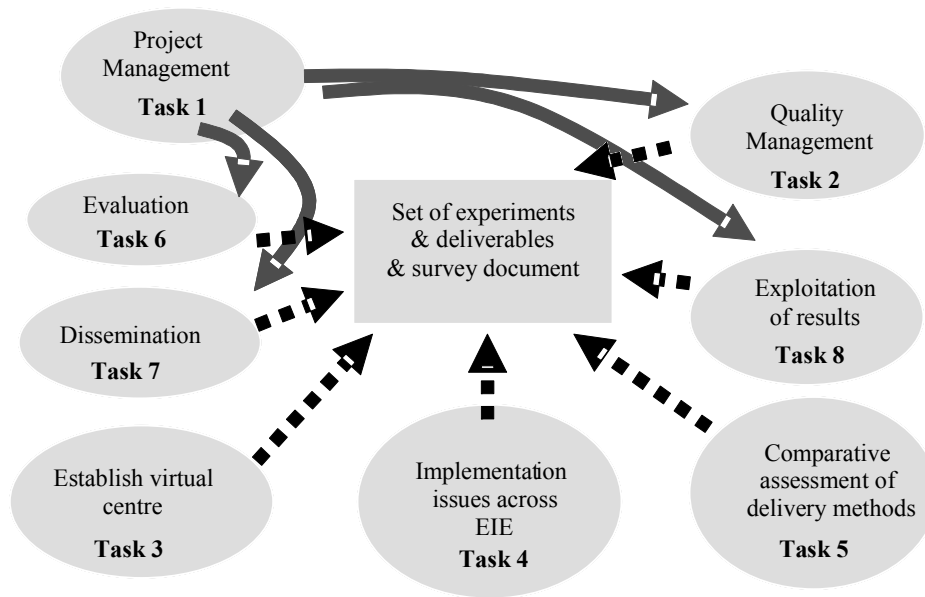


Fig. 1. ELLEIEC project structure.

(EQF).

Unfortunately many institutions still undertake the time consuming process of evaluating all external learning without being fully aware of the new European process available.

Developing a system which will give a learner the opportunity to take advantage of external learning provided with full ECTS recognition will bring to Europe a valuable diversity and specificity in its qualification framework. The original idea was to conduct some experimental work to provide appropriate benchmarks and to undertake a set of surveys on relevant topics across various European countries. All the activities undertaken in this project will follow these objectives:

- to facilitate the faster exchange of information on educational systems by means of a central Gateway;
- to standardise the recognition of diplomas, other qualifications and study periods;
- to stimulate student and staff exchange and cooperation between start-up initiatives;
- to develop joint mobility networks and teaching materials;
- to participate in mutual quality assessment of e-learning tools.

ELLEIEC [1] will principally deal with the following types of informal learning which are increasingly becoming available in HIE, companies, start-up and work-shop seminars & conferences:

- Web based courses: with the evolution of the internet, teachers increasingly are designing valuable web courses,
- Vocational intensive course: mainly dedicated to employer learner and are generally of less than one week duration.

In general these kinds of informal learning, which have a

workload range of close to 1 or 2 credits are not considered for inclusion in a classical formal course in HIE. To take account of this fact, the main output of task 3 will be the design of a Virtual Centre for Enterprise – Centre of Entrepreneurship. The Virtual Centre will provide learners with the opportunity to develop their competence in small, bite-sized pieces as and when they need them or as part of a planned overall learning strategy.

III. PROJECT APPROACH

ELLEIEC TN comprises 60 partners who are involved in a range of different tasks. Based on the experience gained from our last TN (EIE-Surveyor) and the implementation of the Bologna process, we now wish to study and propose some experiments (Fig. 1) to define the more suitable structure framework to improve European incoming mobility and international partnership taking account of the Lifelong Learning issue and recognition problem.

There are three main development tasks (3,4,5) and a set of managing tasks (1,2,6,7,8). This ensures that we have the correct organisational structure to achieve our goals. All partners are involved in at least one of the 3 main tasks.

Task number:

- 1 Project Management
- 2 Quality Management
- 3 Establish virtual centre
- 4 Implementation issues across EIE
- 5 Comparative assessment of delivery methods
- 6 Evaluation
- 7 Dissemination
- 8 Exploitation of results

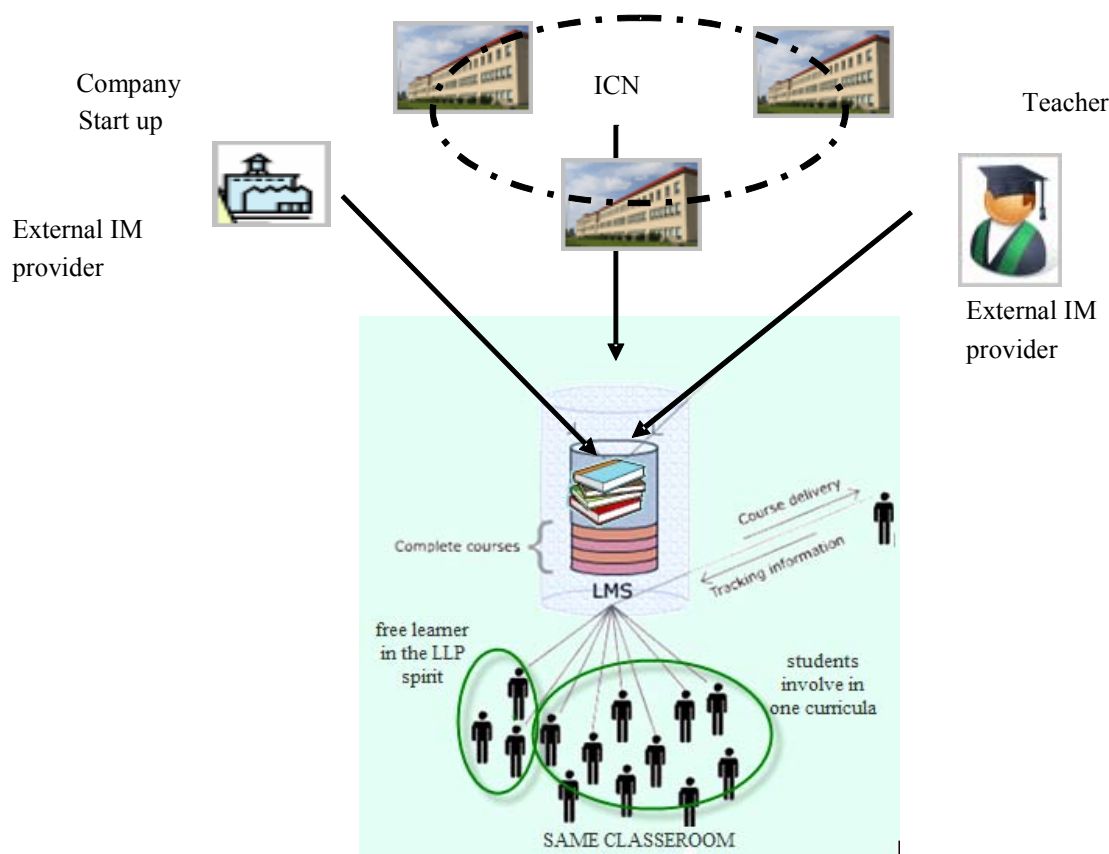


Fig 2. ICN composition: built with the potential of different ELLEIEC partners.

The consortium has been carefully chosen to provide the range of skills and resources required to bring the Virtual Centre for Enterprise into reality. The University of York brings access to an existing data bank of enterprise learning resources, access to a large student enterprise society, links to its partner UK universities of Leeds and Sheffield, access to enterprise assessment instruments through its Education for High Growth Industries project network, and access to its skills forge competence management tool. To ensure the success of the project, we have, in our consortium, a set of relevant partners providing us valuable platform for our experiments:

- An International Bachelor,
- An Erasmus Mundus Master,
- Heads of Doctoral Schools,
- Start-ups and incubator agencies (associated partners),
- Members of accreditation board,
- Experts in assessment of e-learning tools,
- Associated partners from industry and institutions outside of Europe,
- Providers of e-learning centre.

Taking account of this spread of partners we will share the expertise of the various partners to launch a first set of ICN (International Curricula Network) experiments during the 2009/2010 academic year in a real situation (Fig. 2).

An ICN is a set of partners who agree all together to

provide shared curricula or program (partially or totally) in a specialty. Ideally these shared curricula should be recognized at the accreditation level, but the process can be difficult in some cases. In order to initiate the ICN, we consider two pragmatic cases:

1. Cross recognition of "curricula" (learning outcomes, competences, Tuning approach), which is the fact when at least two institutions agree on common modules for an ERASMUS exchange (exchange of students and/or exchange of teachers);
2. Shared recognition of modules, which is the case when a set of modules is developed and implemented within the frame of a program (here, distant modules can be used).

Concerning this second point, we can consider the practical use of a module in various ways:

- Distant course without tutor,
- Distant course with distant tutor,
- Distant course with local tutor,
- Local classical course,
- Intensive training for industrial employer.

The ICN concept has been proposed in the ELLEIEC project. The work done during the project was the result of several discussions to organise the ICN structure and we intend to demonstrate its feasibility with a demo version providing us a useful feedback to ensure its effectiveness. The last role of the ICN is to act as a "pedagogical

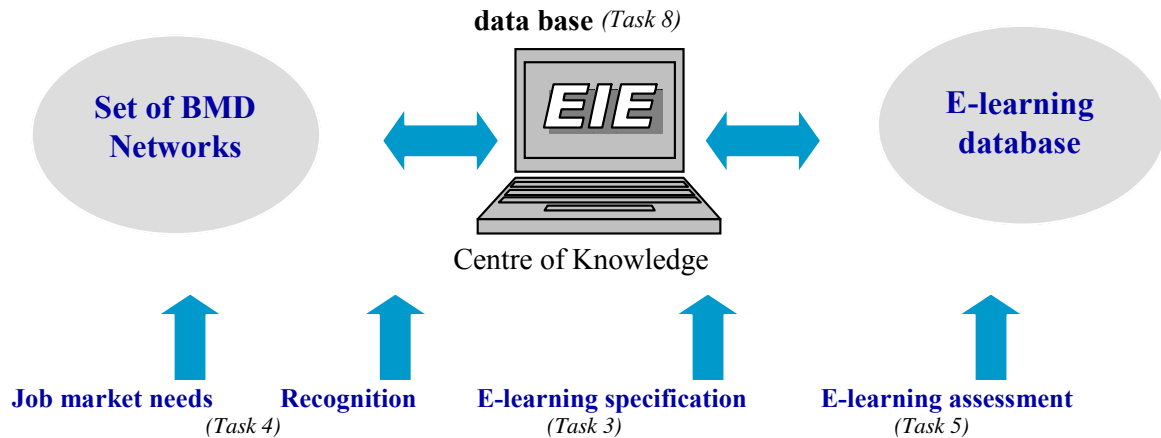


Fig 3. Interaction between tasks and experiment.

committee" in order to identify good approaches in the setting of existing or new LLL curricula.

Experiments:

Experiments will be used to simulate in a real situation a process concerning each of our task objectives. For some experiments, ELLEIEC partners and some BMD degrees (BMD, Bachelor Master Doctorate), together with students, associated partners and pedagogical resources will be involved. They will take place during an academic year or during some dedicated events. A large part of the activities of the three main tasks (3,4,5) will be built around these experiments (Fig. 3).

Experiments will help us to identify the main problems that need to be addressed before the proposed process can be implemented. A set of questionnaires, which will take advantage of all particularities and experiences of European countries involved in ELLEIEC TN, will be used in these experiments. As an added value, some experiments will focus their activities around the EIE (Electrical and Information Engineering) discipline with a special attention to innovative degrees corresponding to the pedagogical needs for developing new skills for new jobs.

IV. CURRENT PROJECT OUTCOMES

One of the two main outcomes of our first term of the project has been to design the entrepreneurship centre and to propose the ICN concept which was not listed as a specific outcome at the beginning of the project. In this first phase, the ICN concept in its first draft has been elaborated and some experiments have been conducted during the last academic year with collaborative work between tasks among various European countries.

After making a survey on the different platforms of web based or intensive course available on the net, it was noted that none of them were offering ECTS recognition in bite-sized pieces. This fact is easy to understand as we know that each institution provides either their pedagogical courses or uses a bilateral agreement to recognise external

courses for big size course modules. In the framework of ELLEIEC project we want to give access to some International Modules (IM: web based or intensive training classroom course) which are used already either in some existing bilateral agreement or in some informal situation and study their use, ensuring ECTS recognition, within an International Curricula Network (ICN) via the ELLEIEC project experiment and survey.

Definition of the ICN: The ICN is a set of Curricula and Course Providers related to the same field and cooperating in a multi-lateral agreement with full ECTS recognition. The curricula entity in the ICN will certify the level and number of credits attributed to the IM course offered by the provider body for its own degree. The provider entity is the owner of the IM used in the ICN and it is their prerogative to set the conditions for joining this course. The IM could be available as a simple link in ELLEIEC website or implemented in the ELLEIEC IM database (depending of the owner).

The organization of the work around the experiment is divided in three main scientific tasks:

- WP3 is in charge to design the entrepreneurship centre as an IM provider;
- WP4 propose some curricula to analyze recognition issue at BMD level (Bachelor Master and Doctoral level);
- WP5 proposes an IM course framework guideline which will be used to specify the agreement between an ICN and an IM provider.

WP3: The overall objective of this task is to develop a sustainable centre that will provide learners with enterprise education across Europe for the foreseeable future in bite-sized [4]. To this end development and growth have been designed specifically to be steady, progressive and careful for the benefit of the long-term success. To ensure this long-term success establishment of the Virtual Centre VCE [5] (Virtual Centre of Entrepreneurship) will be in three stages, the first stage will be a small pilot trial designed to test the learner registration, security and data protection parts of the system; the second stage will be a larger scale



Fig 4. Top Section of VCE Welcome Page.

delivery designed to test how the system and its administrative systems respond to volume usage; the final stage will be the full scale delivery of the system. Stages 1 and 2 are addressed in this short-term section while stage 3 is described in the long-term targets section. The initial learning resources will be provided in English although actions to translate them into a range of languages will be started early in the project.

The first design (Fig. 4) of the VCE has been done and is available at url: <http://vce.york.ac.uk>.

The WP4 undertake an action to measure the need of competencies recognition through a set of some curriculum benchmark and a survey on the different frameworks used to take account of the Lifelong Learning accumulation system.

The study will focus on the Energy, Wireless Network, Imaging & Computer Vision & Robotics disciplines as there are parts of some running bachelor and master degrees within ELLEIEC partners. In a secondary way, the emerging of new challenges like market globalization, technology progress and growing unemployment will lead to a primordial adaptation of our classical curricula to participate to the competitiveness, innovation and attractiveness of the European area by enhancing the industrial culture and Lifelong Learning process inside our

institutions.

A consortium of universities is working in a new ICN around Renewable Energy focusing in a survey in good practice in Tuning methodology in teaching content. It gives the possibility for Studies and comparative Research, or for creating Networks (Fig. 5).

The project aims at enhancing the understanding of the field of renewable energy among the citizens (consumer site and local energy production), and enhancing the awareness of energy producers to move towards renewable energy (production site).

The project will create a web-based database containing useful information in the field of renewable, sustainable and alternative energy for each country. It will include:

- An analysis of the skills needed in the renewable and sustainable energy sector. What are the existing and upcoming technical job profiles in this economy? We will define correctly competences, skills and knowledge required for employers in green jobs, in the spirit of 'new skill for new job'.
- A survey of all education programmes and learning systems yielding to jobs in this field. The information about the education programmes will include the learning outcomes (competences, skills, knowledge) and the links to corresponding job profiles. We will cover all sectors and



Fig 5. An example of ICN member in the field of green energy (master-energie-renouvelable.univ-lyon1.fr).

forms of learning. The education programmes can be provided by traditional education institutes but also by companies, competence centres and sectoral organizations.

- A website gateway to existing jobs vacancies in the field of sustainable energy. It is estimated that more than 2.7 million jobs will be created by 2020 if Europe adopts a full renewable energy agenda.
- Examples of good practices in the field: demonstration projects. Innovations and success stories from the academic world and spin-offs will be made known to the public.

More ICN are presented in the ICN web portal at www.intendia.it.

A special sub-task in WP4 will describe the new missions and evaluation criteria that will be used for doctoral studies after harmonization following the Bologna process. A special attention is devoted to the studies in the field of Electrical and Information Engineering on the basis of experience in the direction of such a graduate school in France.

To prepare a doctorate the student must reach a specific environment that allows developing new researches. This environment is summarized in Fig. 6 that shows the main points that must be fulfilled to prepare a doctorate in suitable conditions in a classical way.

It is well accepted since a very long time that to prepare a doctorate (or PhD) a student must produce a scientific work that is mainly performed, at least in the field of Electrical and Information Engineering (EIE), in a research laboratory or research unit. In this environment, the student may manage some technical experiences, create new tools or instruments, analyze the existing results in the international environment, propose modelling and simulation involving recent tools, etc.

WP5: A key part of Workpackage 5 will be the quantitative assessment of the effectiveness of the bite-sized non classical learning available compared to similar learning by conventional teaching methods.

Comparing traditional Learning and E-Learning

There is an argument that traditional learning is the best way of maintaining a learning process. Ross and Schulz [6]

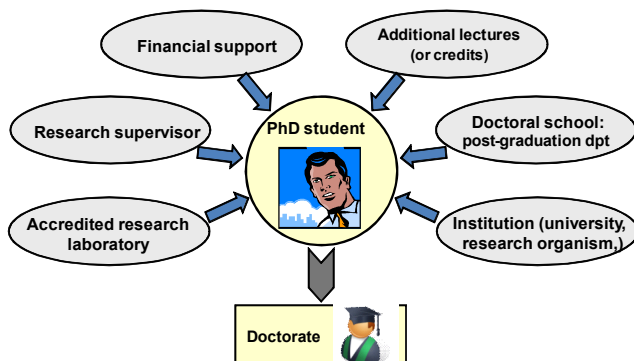


Fig 6. Main general conditions for preparing a doctorate. Not only the research environment is required. Credits of additional education or training may be required. (After [8]).

suggest that computer aided instruction may not be the most appropriate method of learning for all students. Other models are always considered to be inferior or less efficient. However, there is no finding to support this argument, and research shows that technology-supported models are at least as good as traditional learning [7].

The increase is not uniform across all learners in the class owing to the differing entrepreneurial aspirations of students, with some having no aspirations at all. The assessment of these measures is via a questionnaire given to the learners before and after the educational experience. These questionnaires will be built into the Virtual Centre such that all learners will be required to complete a questionnaire upon registration with the Centre and then post questionnaires periodically through their studies. The analysis of the pre and post questionnaires will enable the change in the entrepreneurial measures to be determined. The results will be of use to the project as it will, over a period of time, enable us to build a picture of the relative effectiveness of the available learning resources and their continuous improvement.

Several cases of study have been implemented with students around Europe:

Spain: they are using mobile technology to deploy a more interactive delivery method where students employ the Tablet PCs to both follow the instructor presentations and to solve in-class active learning exercises. Additionally, the ease of soliciting responses from students enables instructors to integrate formative assessment into classroom practices and providing immediate feedback.

UK: This test aims to investigate the reasons for this decline and to examine whether or not there is a real issue in the methods by which mathematics is delivered in engineering subjects. It aims to find out whether new emerging e-learning tools are really a support or an obstacle when teaching mathematics. To this effect, a survey is carried out among final year Bachelor degree Digital Signal Processing students in order to obtain their opinion on the subject matter.

An overview of all running experiments from many European countries is given in the following Table I.

A work package 'Exploitation of results', task 8 has provided the web software assistance to launch the first phase of experiments and to product the specification of the whole Web portal gateway with respect to the three main tasks needed until the end of the project.

TABLE I
LIST OF EXPERIMENTS

Type of Experiments	Country	Number of involved students
A Practice in Using E-Portfolio in a Higher Education Course Taught at Distance	Izmir University, Turkey	22 students: 7 undergrad., 15 grad.
Enhancing learning by using Tablet PCs in a networked classroom	Universidad Politécnica de Valencia, Spain	20-20 / 30
EURO-QLIO – European Virtual Campus	Université Nancy 1, France	51
CISCO Courses	Univerity of Rousse, Bulgaria	15
Medical Information Systems - Handbook for Laboratory Exercises and Self testing	Sofia University, Bulgaria	180/year
Blended Learning in Guided Propagation and Antennas	Institut Superior of Lisboa, Portugal	70
Clickers	Riga University, Latvia	
Elearning versus classical one	Portugal	10
PPT course	Northumbria University, Newcastle, UK	
Elearning versus classical one	Ulm University, Germany	20

V. CONCLUSIONS AND PERSPECTIVES

ELLEIEC project has reached a first step by designing a demonstrator for the ICN concept (International Curricula Network) and the VCE (Virtual Centre of Entrepreneurship).

An analysis of the results achieved during the mid term of the project will help to set correctly the next planned one. Based on our first results, the action of ELLEIEC consortium will work on the interaction between the three main development tasks to provide a Business model framework for the ICN concept and the VCE, gathering experience from recognition assessment and new informal bite sized learning experiments and surveys. The real challenge of the ICN and the VCE will be to encourage all involved associated and ELLEIEC members to use the courses online for their own degree with ECTS recognition to ensure its sustainability. Additional sub-task will focus on:

- Lifelong Learning application to PhD level,
- BMD advertising position offer,
- Database of paper outcomes,
- Analysis on new skills for new jobs in EIE.

The next challenger will be to disseminate the proposed model among university institutions which will be set by a new proposal in European Community framework.

REFERENCES

- [1] ELLEIEC: Enhancing Lifelong Learning for the Electrical and Information Engineering Community, <http://www.elleiec.eu/>
- [2] Results of the EIE-Surveyor project, coordination: Jean-Marc THIRIET (ISBN 2-9516740-2-3). (ISBN - 2-9516740-3-1) (ISBN - 2-9516740-4-X).
- [3] H. Yahoui, O. Roesch, J.M. Thiriet, M. J. Martins and O. Bonnaud, "Evolution of master degree in Europe: An overview of the European programme available in electrical and information engineering", EIEEE'05 Conference, Lappeenranta, Finland, June 2005.
- [4] C. Perra, H. Yahoui and T. Ward, "An Entrepreneurship Centre in the lifelong learning spirit", Proc. of the IEEE-ITHET 2010, April 2010, Turkey.
- [5] url: <http://vce.york.ac.uk>
- [6] J. Ross and R. Schulz, "Can computer-aided instruction accommodate all learners equally?", British Journal of Educational Technology, Vol. 30, No. 1, 1999 pp. 5-24.
- [7] D. Rashty, "Traditional Learning vs. eLearning", Mount St. Mary's College. [Online]. (URL http://www.msmc.la.edu/include/learning_resources/). (Accessed August 2010).
- [8] O. Bonnaud and M. Hoffmann, "New challenges of doctoral studies in Europe in the field of electrical and information engineering", Bologna Handbook. C4-4-4, No. 12, 2009, pp. 2-23.

Vision based stabilization of a quadrotor using nested saturation control approach

Z. Zamudio¹, R. Lozano^{2,3}, J. Torres¹, J. L. Rullan-Lara³

¹Automatic Control Department, CINVESTAV-IPN, México D.F., México

²UMI-LAFMIA, CINVESTAV-CNRS, México D.F., México

³Université de Technologie de Compiègne, UTC Heudiasyc, Compiègne, France

Email: zzamudio@ctrl.cinvestav.mx, rlozano@hds.utc.fr, jtorres@ctrl.cinvestav.mx, jose-luis.rullan-lara@hds.utc.fr

Abstract—The problem of estimating position and orientation (pose) of an object in real time constitutes an important issue for vision-based control of robots. In this paper, we present a vision based controller design and its implementation on a quadrotor. The dynamic model is obtained using the well known Euler-Lagrange approach. Experiment results show good performance of the proposed controller using "real-time" optical flow and image processing.

Keywords: Quadrotor, control, real-time systems, vision systems.

I. INTRODUCTION

In the last few years we have witnessed an increasing interest on unmanned aerial vehicles (UAVs) due to their numerous applications either in the military field or in the civilian domain. An important feature is that autonomous navigation systems may substitute human operators in multiple dangerous environments, like surveillance of active volcanos, search and rescue operations of persons in floodings, etc.. The development of reliable UAVs capable of operating in hazardous circumstances is very important since it will protect human life.

A quadrotor is a four-rotor helicopter which has become very common in control and aeronautical laboratories in view of its mechanical simplicity and ease of maintenance. It is an under-actuated dynamical vehicle having four input forces and six degrees of freedom. Unlike regular helicopters which have variable pitch angle rotors, a quadrotor helicopter has four fixed-pitch angle rotors. The basic motions of a quadrotor are generated by varying the rotor speed of each of the four rotors, thereby changing the lift forces. A good controller should properly arrange the speed of each rotor so that the UAV flies in hover or tracks a desired trajectory.

In order to create an autonomous UAV, precise knowledge of the helicopter position and orientation is needed. This information can be obtained from Inertial Navigation Systems (INS), Global Positioning Systems (GPS) or other sensors like ultrasonic sensors, laser radar sensors, etc.. Typically, multiple sensors are used to overcome limitations of an individual sensor, thereby increasing the reliability and decreasing the errors.

The performance of robotic systems evolving in structured environments in the presence of objects whose position and orientation are well known, is a problem currently being

studied. However, performing tasks in dynamic environments presents numerous difficulties that are not yet solved. Vision systems are able to provide extremely useful information in these changing environments, they provide information about the objects in the work scene. Moreover, the use of vision in the robotics field is prompted by the wide availability of unexpensive cameras. Nowadays vision systems provide very rich information with low noise. However, the extraction of relevant information in real time from a single image or set of images is a hard enough task which remains a major obstacle for using vision in mobile robots [1].

We are interested in stabilizing mini rotorcrafts using a vision based control approach. The nonlinear control that is implemented is based on separated saturations and operates at a sampling rate which is compatible with the vision system and is used for position and velocity estimation. The visual servoing control scheme should be robust enough with respect to unavoidable uncertainties existing on the camera and the environment. The main problem in natural landmark navigation is to detect and match characteristic features from sensory inputs. The selection of features is important since it will determine the complexity in features description, detection and matching. Researchers have used different kinds of patterns to be recognized by systems based on computer vision [2]. In the visual servoing approach [3] the first step consists in defining a particular set of characteristic features in the image. The $2\frac{1}{2}$ visual servoing method is used in [4] to make the cartesian trajectory of the UAV controllable. Other recent works present visual approaches for UAVs positioning. In [5] a target composed of 4 circles was used to localize the UAV with respect to the ground with an error based on a spherical centroid data. In [6] a vision system using a ground camera to estimate the pose of the quadrotor was presented. This system uses 5 colored blobs attached to the bottom of the quadrotor, the position of the helicopter is estimated by averaging the five blobs position. The use of a target for autonomous landing has been applied in [7], [8] and [9]. Some progress and several open problems in vision based estimation for UAVs, are described in [9]. The use of video sequences to estimate the motion of UAV flight has been described in [10]. Tracking strategies using targets are studied in [11].

Several approaches to nonlinear control of a mini rotorcraft

can be found in the literature, e.g., in [12] a nonlinear control based on nested saturations is presented. In this approach, the dynamics is decoupled into lateral and longitudinal dynamical systems. Nested saturations control was used to stabilize each system. In [13], the authors propose a robust linear PD controller considering parametric interval uncertainty. Here, the authors present a robust stability analysis and compute the robustness margin of the system with respect to the parameters uncertainty.

II. DYNAMICAL MODEL

The aim of this section is to develop a dynamic model for the quad-rotor using the Euler-Lagrange approach. This model is obtained by representing the aircraft as a rigid body evolving in a three-dimensional space due to the main thrust and three torques [14]. The main thrust u is shown in figure 1. The dynamics of the four electric motors are fast and will be neglected.

The generalized coordinates of the rotorcraft are:

$$q = (x, y, z, \psi, \theta, \phi) \in \mathbb{R}^6, \quad (1)$$

where $\xi = (x, y, z) \in \mathbb{R}^3$ denotes the position of the center of mass of the helicopter relative to a fixed inertial frame (I), and $\eta = (\psi, \theta, \phi) \in \mathbb{R}^3$ are the Euler angles, ψ is the yaw angle around the z -axis, θ is the pitch angle around the modified y -axis, and ϕ is the roll angle around the modified x -axis [15], [16] (see figure 1).

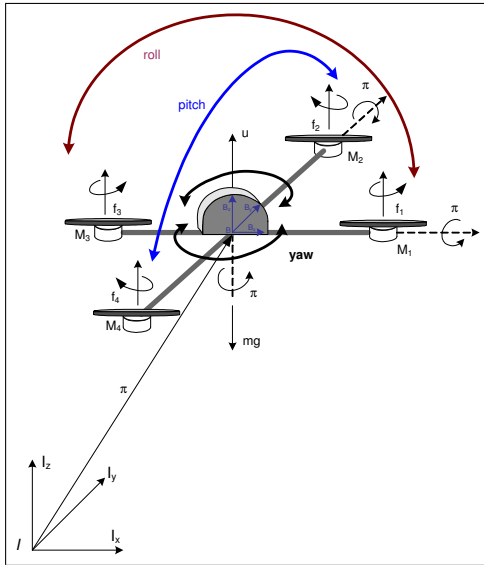


Fig. 1. Schema of the quadri-rotor

Define the Lagrangian

$$L(q, \dot{q}) = T_{trans} + T_{rot} - U \quad (2)$$

where $T_{trans} = \frac{m}{2} \dot{\xi}^T \dot{\xi}$ is the translational kinetic energy, $T_{rot} = \frac{1}{2} \omega^T \mathbf{I} \omega$ is the rotational kinetic energy, $U = mgz$ is the potential energy of the aircraft, z is the rotorcraft altitud, m denotes the mass of the quadrotor, ω is the angular velocity,

$\mathbf{I}$ is the inertia matrix, and g is the acceleration due to gravity. The angular velocity ω in the body fixed frame is related to the generalized velocities $\dot{\eta}$ (in the region where the Euler angles are valid) via the standard kinematic relationship

$$\dot{\eta} = W_v^{-1} \omega \quad (3)$$

where

$$W_v = \begin{bmatrix} -\sin \theta & 0 & 1 \\ \cos \theta \sin \psi & \cos \psi & 0 \\ \cos \theta \cos \psi & -\sin \psi & 0 \end{bmatrix} \quad (4)$$

Define

$$\mathbf{J} = \mathbf{J}(\eta) = W_v^T \mathbf{I} W_v \quad (5)$$

so that,

$$T_{rot} = \frac{1}{2} \dot{\eta}^T \mathbf{J} \dot{\eta} \quad (6)$$

The matrix $\mathbf{J}$ acts as the inertia matrix for the full rotational kinetic energy of the rotorcraft expressed directly in terms of the generalized coordinates η .

The model for the full rotorcraft dynamics is obtained from the Euler-Lagrange equations with external generalized force.

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}} - \frac{\partial \mathcal{L}}{\partial q} = F \quad (7)$$

where $F = (F_\xi, \tau) \cdot \tau$ is the generalized moments and F_ξ is the translational force applied to the rotorcraft due to the control inputs. We ignore the small body forces because they are generally of a much smaller magnitude than the main control inputs u and τ , then we write

$$\hat{F} = \begin{bmatrix} 0 \\ 0 \\ u \end{bmatrix} \quad (8)$$

where u is the main thrust directed out the top of the aircraft expressed as

$$u = \sum_{i=1}^4 f_i \quad (9)$$

and, for $i = 1, \dots, 4$, f_i is the force produced by motor M_i , as shown in Figure 1. Typically, $f_i = k_i \omega_i^2$, where k_i is a constant and ω_i is the angular speed of the i th motor, then

$$F_\xi = R \hat{F} \quad (10)$$

where R is the transformation matrix representing the orientation of the rotorcraft, we use c_θ for $\cos \theta$ and s_θ for $\sin \theta$.

$$R = \begin{bmatrix} c_\theta c_\psi & s_\psi s_\theta & -s_\theta \\ c_\psi s_\theta s_\phi - s_\psi c_\phi & s_\psi s_\theta s_\phi + c_\psi c_\phi & c_\theta s_\phi \\ c_\psi s_\theta c_\phi + s_\psi s_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi & c_\theta c_\phi \end{bmatrix} \quad (11)$$

The generalized torques are thus

$$\tau = \begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} \triangleq \begin{bmatrix} \sum_{i=1}^4 \tau_{M_i} \\ (f_2 - f_4)l \\ (f_3 - f_1)l \end{bmatrix} \quad (12)$$

where l is the distance from the motors to the center of gravity and τ_{M_i} is the couple produced by motor M_i .

Since the Lagrangian contains no cross-terms in the kinetic energy combining ξ and η , the Euler-Lagrange equation can be partitioned into the dynamics for the ξ coordinates and the η dynamics. One obtains

$$m\ddot{\xi} + \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = F_\xi \quad (13)$$

$$J\ddot{\eta} + \dot{J}\dot{\eta} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T J \dot{\eta}) = \tau \quad (14)$$

Defining the Coriolis/Centripetal vector

$$\bar{V}(\eta, \dot{\eta}) = \dot{J}\dot{\eta} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T J \dot{\eta}) \quad (15)$$

we may write

$$J\ddot{\eta} + \bar{V}(\eta, \dot{\eta}) = \tau \quad (16)$$

but we can rewrite $\bar{V}(\eta, \dot{\eta})$ as

$$\begin{aligned} \bar{V}(\eta, \dot{\eta}) &= \left(\dot{J} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T J) \right) \dot{\eta} \\ &= C(\eta, \dot{\eta}) \dot{\eta} \end{aligned} \quad (17)$$

where $C(\eta, \dot{\eta})$ is referred to as the Coriolis terms and contains the gyroscopic and centrifugal terms associated with the η dependence of J .

Finally we obtain

$$m\ddot{\xi} = u \begin{bmatrix} -s_\theta \\ c_\theta s_\phi \\ c_\theta c_\phi \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (18)$$

$$J\ddot{\eta} = -C(\eta, \dot{\eta})\dot{\eta} + \tau \quad (19)$$

In order to simplify let us propose a change of the input variables.

$$\tau = C(\eta, \dot{\eta})\dot{\eta} + J\tilde{\tau} \quad (20)$$

where

$$\tilde{\tau} = \begin{bmatrix} \tilde{\tau}_\psi \\ \tilde{\tau}_\theta \\ \tilde{\tau}_\phi \end{bmatrix} \quad (21)$$

are the new inputs. Then

$$\ddot{\eta} = \tilde{\tau} \quad (22)$$

Finally we obtain

$$m\ddot{x} = -u \sin \theta \quad (23)$$

$$m\ddot{y} = -u \cos \theta \sin \phi \quad (24)$$

$$m\ddot{z} = -u \cos \theta \cos \phi - mg \quad (25)$$

$$\ddot{\psi} = \tilde{\tau}_\psi \quad (26)$$

$$\ddot{\theta} = \tilde{\tau}_\theta \quad (27)$$

$$\ddot{\phi} = \tilde{\tau}_\phi \quad (28)$$

The control inputs u , $\tilde{\tau}_\psi$, $\tilde{\tau}_\theta$ and $\tilde{\tau}_\phi$ are the total thrust or collective input (directed out from the bottom of the aircraft) and the new angular moments (yawing moment, pitching moment and rolling moment).

III. CONTROL STRATEGY

In this section we use a control strategy for stabilizing the quadrotor at hover, developed in [12], [13]. In these papers, the authors proved the stability of the closed loop system. First, the main thrust u is used to stabilize the altitude of the rotorcraft and then, the yaw angle is stabilized. Next, the roll angle ϕ and the y -displacement are stabilized using a controller designed for the PVTOL model [17], [18]. Finally, the pitch angle θ and the x -displacement are stabilized around the origin.

A. Control of Altitude and Yaw

The vertical displacement z in (25) is controlled by forcing the altitude to satisfy the dynamics of a linear system. Thus, we set

$$u = \frac{r_1 + mg}{\cos \theta \cos \phi} \quad (29)$$

with

$$r_1 = -a_{z_1} \dot{z} - a_{z_2} (z - z_d) \quad (30)$$

where a_{z_1} , a_{z_2} are positive constants and z_d is a positive constant representing the desired altitude. The control yaw angle is chosen as

$$\tilde{\tau}_\psi = -a_{\tau_1} \dot{\psi} - a_{\tau_2} (\psi - \psi_d) \quad (31)$$

Introducing (29)-(31) in to (23)-(26) and assuming $\cos \theta \cos \phi \neq 0$, that is $\theta, \phi \in (-\pi/2, \pi/2)$, one obtains

$$m\ddot{x} = \frac{(r_1 + mg) \tan \theta}{\cos \phi} \quad (32)$$

$$m\ddot{y} = (r_1 + mg) \tan \phi \quad (33)$$

$$m\ddot{z} = \frac{1}{m} (-a_{z_1} \dot{z} - a_{z_2} (z - z_d)) \quad (34)$$

$$\ddot{\psi} = -a_{\tau_1} \dot{\psi} - a_{\tau_2} (\psi - \psi_d) \quad (35)$$

The control gains a_{z_1} , a_{z_2} , a_{τ_1} , a_{τ_2} , are positive constants chosen to ensure a stable well-damped response of the quadrotor. From (34) and (35) it follows that if ψ_d and z_d are constants, $\dot{\psi}$ and $\dot{z} \rightarrow 0$ which means (see 35) that $\psi \rightarrow \psi_d$, similarly $z \rightarrow z_d$.

B. Control of Lateral Displacement and Roll

We now determine the input $\tilde{\tau}_\phi$ to stabilize the (y, ϕ) subsystem. The control algorithm is obtained step by step. For ϕ close to zero, the (y, ϕ) subsystem is represented by four integrators in cascade. The (y, ϕ) subsystem is given by (28) and (33). We assume $\psi_d = 0$ in (31) and (35). Therefore, from $\psi \rightarrow \psi_d$ it follows that $\dot{\psi} \rightarrow 0$. Note that from (30) and (34) that $r_1 \rightarrow 0$. For a time T large enough, r_1 and $\dot{\psi}$ will be arbitrarily small.

Given that, the quadrotor control inputs are saturated, we have chosen the control strategy proposed in (25). Next we choose the amplitudes of the saturation functions in the nested saturation control law in such a way that after a finite time, the roll angle lies in the interval $-1 \text{ rad} \leq \phi \leq 1 \text{ rad}$. Therefore,

the difference $\tan \phi - \phi$ is also small. Thus, after a time T large enough r_1 is small and the (y, ϕ) subsystem reduces to

$$\ddot{y} = g\phi \quad (36)$$

$$\ddot{\phi} = \tilde{\tau}_\phi \quad (37)$$

which represents four integrators in cascade.

C. Control of forward displacement and pitch

For small ϕ and r_1 , (32) reduces to

$$\ddot{x} = -g \tan \theta \quad (38)$$

For the (x, θ) subsystem

$$\ddot{x} = -g \tan \theta \quad (39)$$

$$\ddot{\theta} = \tilde{\tau}_\theta \quad (40)$$

it is assumed that the control strategy will ensure a small bound on $|\theta|$ in such a way that $\tan \theta \approx \theta$. Therefore, (39) reduces to $\ddot{x} = -g\theta$. Using a procedure similar to the one proposed for the roll control, we obtain

$$\tilde{\tau}_\theta = -\sigma_{\theta_1} (\dot{\theta} + a) \quad (41)$$

$$a = \sigma_{\theta_2} (\theta + \dot{\theta} + b) \quad (42)$$

$$b = \sigma_{\theta_3} \left(2\theta + \dot{\theta} - \frac{\dot{x}}{g} + c \right) \quad (43)$$

$$c = \sigma_{\theta_4} \left(\dot{\theta} + 3\theta - 3\frac{\dot{x}}{g} - \frac{x}{g} \right) \quad (44)$$

IV. VISION SYSTEM

A. Optical Flow

Optical Flow is an approximation of the local image motion based on local derivatives in a given sequence of images. It is assumed that all temporal intensity changes are due to motion only [19].

Assume $I(x, y, t)$ is the center pixel in a $n \times n$ neighbourhood and moves by $\delta x, \delta y$ in time δt to $I(x + \delta x, y + \delta y, t + \delta t)$. Since $I(x, y, t)$ and $I(x + \delta x, y + \delta y, t + \delta t)$ are the images of the same point. We then have:

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t) \quad (45)$$

We can perform a 1st order Taylor series expansion about $I(x, y, t)$ in the equation above to obtain:

$$\begin{aligned} I(x + \delta x, y + \delta y, t + \delta t) &= \\ &= I(x, y, t) + \frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \delta t + H.O.T. \end{aligned} \quad (46)$$

$$I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) + \nabla I \cdot (\delta x, \delta y) + \delta t I_t + R^n \quad (47)$$

where H.O.T. are the Higher Order Terms, which we assume small and can safely be ignored.

Using the above two equations we obtain:

$$(I_x, I_y) \cdot (v_x, v_y) + I_t = 0 \quad (48)$$

where $\nabla I = (I_x, I_y)$ is the spatial intensity gradient and $\vec{v} = (v_x, v_y)$ is the image velocity or optical flow at pixel (x, y) at time t , where $v_x = \frac{dx}{dt}, v_y = \frac{dy}{dt}$. $\nabla I \cdot \vec{v} = -I_t$ is called the 2D Motion Constraint Equation. From this equation is not possible to determine $\vec{v} = (v_x, v_y)$ since it is an ill conditioned problem. This is a consequence of the aperture problem: there is usually insufficient local image intensity structure to measure full image velocity, but sufficient structure to measure the component normal to the local intensity structure. The problem of computing full image velocity then becomes the one of finding an additional constraint that yields a second different equation in the same unknowns.

1) *Lucas-Kanade Method:* One of the most popular algorithms currently known for computing optical flow is the algorithm developed by Lucas and Kanade [20], [21]. The Lucas-Kanade detector is a gradient base displacement detector and is the basis of many image pairing algorithms. It does not give an exact solution but a relatively good approximation.

Lucas-Kanade supposed a locally constant flow in a small spatial neighborhood Ω and proposed a solution for (48) using weighted least squares.

The solution is obtained by minimizing the following term in the neighborhood Ω

$$\sum_{x \in \Omega} W^2(x) [\nabla I(x_1), \dots, \nabla I(x_N)]^T \quad (49)$$

where $W(x)$ is a weighted function. Using the following notation

$$A = [\nabla I(x_1), \dots, \nabla I(x_N)]^T \quad (50)$$

$$W = \text{diag} [W(x_1), \dots, W(x_N)] \quad (51)$$

$$b = -[I_t(x_1), \dots, I_t(x_N)]^T \quad (52)$$

(49) can be rewritten in compact form as

$$A^T W^2 A v = A^T W^2 b \quad (53)$$

If $A^T W^2 A$ is not singular, a solution can be obtained by using the Moore-Penrose pseudoinverse which is a solution in the least squares sense. Matrix $A^T W^2 A$ is in fact singular when the gradient is constant in one or more directions, over the neighborhood Ω .

Matrix $A^T W^2 A$ is 2×2 as follows

$$A^T W^2 A = \begin{bmatrix} \sum W^2(x, y) I_x^2(x, y) & \sum W^2(x, y) I_x(x, y) I_y(x, y) \\ \sum W^2(x, y) I_y(x, y) I_x(x, y) & \sum W^2(x, y) I_y^2(x, y) \end{bmatrix} \quad (54)$$

where all sums are taken over the pixels (x, y) in the spatial neighborhood Ω .

B. Position measurement using a single camera

1) *Pinhole Camera Model:* The pinhole camera model defines the geometric relationship between a 3D point and its 2D corresponding projection on to the image plane. When using a pinhole camera model, this geometric mapping from 3D to 2D is called a perspective projection [22].

We consider the central projection of points in space onto a plane. Let the center of projection be the origin of a Euclidean coordinate system, and consider the plane $z = f$, which is

called the *image plane*. Under the pinhole camera model, a point in space with coordinates $X = (x, y, z)^T$ is mapped to the point on the image plane where a line joining the point x to the center of projection meets the image plane. This is shown in figure (2). By similar triangles, one readily finds that the point $(x, y, z)^T$ is mapped to the point $(f x/z, f y/z, f)^T$ on the image plane. Ignoring the final image coordinate, we see that

$$(X, Y, Z)^T \mapsto (fX/Z, fY/Z)^T \quad (55)$$

describes the central projection mapping from world to image coordinates.

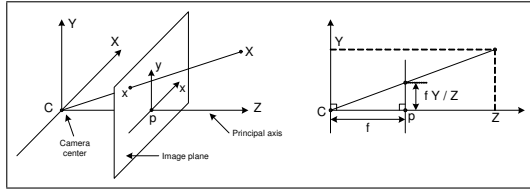


Fig. 2. Pinhole camera geometry

The center of projection is called the camera center. It is also known as the *optical center*. The line from the camera center perpendicular to the image plane is called the *principal axis* or *principal ray* of the camera, and the point where the principal axis meets the image plane is called the *principal point*. The plane through the camera center parallel to the image plane is called the *principal plane* of the camera.

If the world and image points are represented by homogeneous vectors, then central projection is very simply expressed as a linear mapping between their homogeneous coordinates, for more details, see [22].

C. OpenCV Implementation

In this work we have used Logitech webCam (Quickcam Pro 9000, 2 Megapixels with autofocus), which provides the images in real time. In order to process the images we used OpenCV library [23], [24], and the programming environment is Visual Studio C++ 2008.

The optical flow was computed by using the function developed in the Lucas-Kanade algorithm. This function was obtained using a Gaussian pyramid in an iterative way trying to minimize the error of optical flow. Optical Flow is computed only on a set of points in the scene [25].

Some of the main functions used for optical flow are: *allocateOnDemand*, *cvConvert Image*, *cvGoodFeaturesToTrack*, *cvCalcOpticalFlowPyrLK*.

For the camera position a mark is distinguished, which provides the distance z , and with this parameter we can get both x and y .

V. EXPERIMENTAL PLATFORM

The mini rotorcraft is powered by four brushless motors with fixed pitch propellers. Altitude is controlled by the collective thrust of the motors. The pitch motion is achieved through the angular velocity difference between motors M_1

and M_2 as is shown in Figure 1. The vehicle roll position is controlled by the angular velocity difference between motors M_3 and M_4 . The total weight of the vehicle is about 900 gr, with a flight endurance of 15 minutes approximately.

Theoretical results obtained were incorporated into an autopilot control system using an architecture based on a Rabbit microcontroller RCM3000. These microcontrollers are capable of handling floating point operations and multitasking processing virtually due to the enhancement compiler Dynamic C. We use an Inertial Measurement Unit (IMU) using accelerometers, gyros and a compass to obtain the yaw, pitch and roll angles and angular rates. The IMU information is sent to the microcontroller which also reads reference control inputs from an operator through a serial wireless modem. This modem sends the parameters obtained with the camera, i.e., optical flow (linear velocities and directions), and position obtained from a landmark or reference, from the computer to the microcontroller. The microcontroller subsequently combines this information to compute the control input and sends the control corrections to the motors through an I2C serial port. Hence, the brushless motor speed controllers or booster are handled by the I2C port of the microcontroller.

VI. EXPERIMENTAL RESULTS

A. Vision System

In figure (3) we can see the result of image processing, where it is used a one in a circle as a landmark or reference to compute the quadrotor position in the axes x , y and z (represented by: dx , dy , dz in the black box). The closed loop algorithm for the altitude used the z measurement obtained by the camera.

Optical flow is indicated by the green arrows, which provide the velocity in x and y .

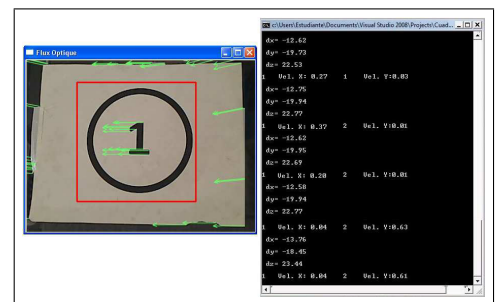


Fig. 3. Position and Optical Flow Results

B. Control

The control objective is to make the quadrotor hover at an altitude of 40 cm, the gains of control equations were adjusted in practice to obtain a fast aircraft response but avoiding mechanical oscillations as much as possible. The parameters were also chosen in such a way that the aircraft attitude remains very close to a desired point. The figure (4) shows the quadrotor flying at approximately 40 cm in real time, figure (5) shows the position signal to the aircraft.

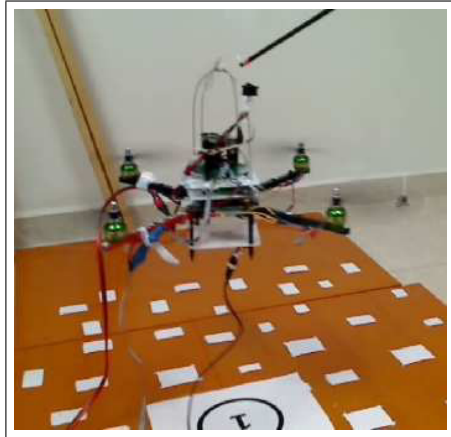


Fig. 4. Quadrotor

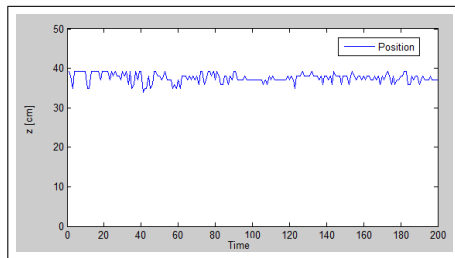


Fig. 5. Position Signal

Figure (6) shows the signal with corresponding angular velocity and acceleration, i.e. around $(\psi, \theta, \phi) = (0, 0, 0)$ (orientation).

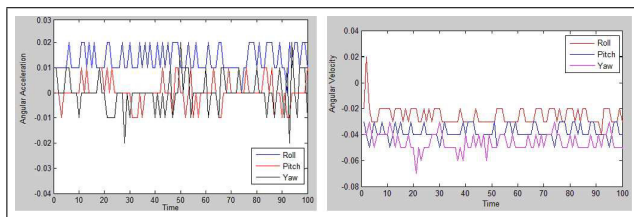


Fig. 6. Angular Velocity and Acceleration

VII. CONCLUSIONS AND FUTURE WORK

In this work we have presented the stabilization of a quadrotor using vision. We applied a nested saturation non-linear control strategy to achieve the aircraft stabilization and positioning. We presented a vision system using a single camera which takes a real time image and after image processing we obtain the velocities in both x and y axis and the quadrotor altitude z . The real-time experiments have shown an acceptable performance of the helicopter when applying the proposed control law. Future work includes improving the vision algorithm in order to reduce the time delay in processing the image.

REFERENCES

- [1] Carlos M. Soria, Ricardo Carelli, "Control de un Robot Móvil utilizando el flujo óptico obtenido través de un sistema omnidireccional catadióptrico", *Memorias Jornadas de Investigación EIA*, 2009. ISSN:2027-0313.
- [2] J. Borenstein, H.R. Everett, L. Feng, and D. Wehe, "Mobile Robot Positioning-Sensors and Techniques", *Invited paper for the Journal of Robotic Systems, Special Issue on Mobile Robots*, Vol. 14 No. 4, pp. 231 – 249.
- [3] B. Espiau, F. Chaumette, and P. Rives, "A New Approach to visual Servoing in Robotics", *IEEE Trans on Robotics and Automation*, Vol. 8, No. 3, June 1992.
- [4] N. Metni, T. Hamel, "A UAV for bridge inspection: Visual servoing control law with orientation limits", *Automation in Construction*, Vol. 17, pp. 3–10, Elsevier, 2007.
- [5] N. Guenard, T. Hamel and R. Mahony, "A practical Visual Servo Control for a Unmanned Aerial Vehicle", *Proc. of the IEEE International Conference on Robotics and Automation*, ICRA2002.
- [6] E. Altug, J.P. Ostrowski, R. Mahony, "Control of a Quadrotor Helicopter Using Visual Feedback", *Proc. of the 2002 IEEE International Conference on Robotics and Automation*, Washington, DC. May 2002.
- [7] H.-W. Schulz, M. Buschmann, L. Kruger, S. Winkler and P. Vörsmann, "Towards Vision-Based Autonomous Landing for small UAVs-First Experimental Results of the vision system", *Journal of Aerospace Computing, Information, and Communication*, Vol. 4, May 2007.
- [8] Z. Yu, K. Nonami, J. Shin and D. Celestino, "3D Vision Based Landing Control of a Small Scale Autonomous Helicopter", *International Journal of Advanced Robotic Systems*, Vol. 4, No. 1, pp. 51-56, 2007.
- [9] A. Kurdila, M. Nechyba, R. Praznica, et al. "Vision-Based Control of Micro-Air-Vehicles: Progress and Problems In Estimation", *43rd IEEE Conference on Decision and Control*, Atlantis Paradise Island, Bahamas, 2004.
- [10] Z. He, R.V. Iyer, and P.R. Chandler, "Vision-Based UAV Flight Control and Obstacle Avoidance", *American Control Conference*, June 2006.
- [11] I.F. Mondragón, et al. "Visual Model Feature Tracking For UAV Control", *Intelligent Signal Processing, 2007. WISP 2007. IEEE International Symposium on*, 1-6 pp. Oct. 2007.
- [12] P. Castillo, R. Lozano and A. Dzul, "Stabilization of a mini-rotorcraft having four rotors", in *IEEE/RSJ-IROS. (2004)*, pp. 2693-2698, Sendai, Japan.
- [13] D. Lara, A. Sanchez, R. Lozano and P. Castillo, "Real Time Embedded Control System for VTOL Aircrafts: Application to Stabilize a quadrotor helicopter", in *IEEE/CCA. (2006)*.
- [14] P. Castillo, A. Dzul, R. Lozano, "Real-Time Stabilization and Tracking of a Four Rotor Mini-Rotorcraft", *IEEE Transactions on Control Systems Technology*, Vol. 12, No. 4, July 2004.
- [15] T.S. Alderete, "Simulator aero model implementation", NASA Ames Research Center, Moffett Fiel, California. (Available at <http://www.aviationsystemsdivision.arc.nasa.gov/publications/html/rsim/Toms.pdf>).
- [16] B. Etkin and L. duff Reid, "Dynamics of Flight", John Wiley and Sons, Inc., New York, 1959. ISBN 0-471-03418-5.
- [17] P. Castillo, R. Lozano and A. Dzul, "Modeling and Control of Mini-Flying Machines", Springer-Verlag in *Advances in Industrial Control*, 2005.
- [18] J. Hauser, S. Sastry and G. Meyer, "Nonlinear control desing for slightly nonminimum phase systems: Application to V/STOL aircraft", *Automatica*, vol. 28, no. 4, pp. 665-679, 1992.
- [19] J.L. Barron, N.A. Thacker, "Tutorial: Computing 2D and 3D Optical Flow", Medical School, University of Manchester, 2005.
- [20] Bruce D. Lucas and Takeo Kanade. "An Iterative Image Registration Technique with an Application to Stereo Vision". *Proceedings of Imaging Understanding Workshop*.
- [21] Jean-Yves Bouguet, "Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the algorithm", Intel Corporation Micro-processor Research Labs (2000).
- [22] Richard Hartley, Andrew Zisserman, "Multiple View Geometry in computer vision", Second Edition, Cambridge University Press. 2003.
- [23] Gary Rost Bradski and Adrian Kaeller, O' Reilly Media, "Learning OpenCV: Computer Vision with the OpenCV Library", September 2008.
- [24] opencv.willowgarage.com
- [25] Jianbo Shi and Tomasi C. "Good features to track". *IEEE Computer Society Conference on* 21-23 June 1994 Page(s):593 – 600.

INDEX OF AUTHORS

Hazem	Abbas	9	Zoran	Budimac	306, 644
Hesham	Abdel-Ghaffar	9	Elena	Bunciu	138
Mihail	Abrudean	400	Adrian	Burlacu	144
Zied	Achour	57	Bogdan	Burlacu	238
Kondo H.	Adjallah	57	Simona	Caraiman	27
Felix	Adochiei	526	Madalina	Carbureanu	450
Ioan	Agavriloaei	15, 21	Daniela	Carp	93
Luis	Aguilar	382	Constantin-Florin	Caruntu	149
Bogdan Ioan	Aignatoaiei	324, 634	Petru	Cascaval	288
Francisco Antonio	Alencar Menezes	377	Gabriel Antonio Perez	Castañeda	156
Adrian	Alexandrescu	15, 21	Emil	Ceanga	93
Andrei	Amariutei	27	Vlad	Cehan	336
Gladiola	Andruseac	526	Daniela Cristina	Cernega	493, 576
Takahiko	Aoyagi	430	Marta	Chinnici	178
Bogdan	Apostol	33	Elena	Cimpoeșu	162
Alexandru	Archip	40	Razvan	Ciobotariu	526
Ovidiu	Aritoni	45	Constantin	Circiumaru	365
Timotei	Asaftei	51	Bogdan	Ciubotaru	162
Victor	Asavei	462	Frédéric	Clairret	537
Maen	Atli	57	Tiberiu Teodor	Cocias	273
Jean-Francois	Aubry	156	Valeriu	Codreanu	168
Cezar	Babici	63	Dorian	Cojocar	212, 359
Costin	Badica	69, 531, 412	Mihaela	Colhon	172
Vadim	Baitoiu	75	Tiberiu	Colosi	400
Naim	Bajcinca	81	Vasile	Comnac	341
Andreea	Balau	87	Hariton	Costin	526
Jean François	Balmat	505	Sorin	Cotofana	224
Marian	Barbu	93	Mitica	Craus	15, 21, 40
Tudor	Barbu	99	Salvatore	Cuomo	178
Viorel	Barbu	103	Nguyen van	Cuong	184, 424
Lucian-Florentin	Barbulescu	105	Gilney	Damm	184, 424
Alexandru	Barleanu	75, 582	Bruno	Depraetere	467
Giovanni	Barroso	377	Mirela	Dobra	200
Andrzej	Bartoszewicz	294	Petru	Dobra	200
Nicu George	Bizdoaca	444	Radu	Dobrescu	436
Eugen	Bobasu	550	Constantin-Catalin	Dosoftei	190
Olivier	Bonnaud	664	Elia	Dragomir	450
Francisco Rodríguez	Borja	628	Claudia-Adina	Dragos	194
Corneliu	Botan	110	Radu	Duma	200
Alexandru Ionut	Botas	115	Bogdan	Dumitrascu	206
Nicolae	Botezatu	120	Sorin	Dumitru	212, 359
Radu	Bozomitu	336	Abhishek	Dutta	218, 467
Florin Catalin	Braescu	126	Marius	Enachescu	224
Nicolae	Brinzei	156	Mona	Faraji-Niri	230
Cristina	Budaciu	132	Lavinia	Ferariu	126, 238

Adrian	Filipescu	206, 499	Sorin	Manoiu-Olaru	347,353, 598
Adriana	Filipescu Jr.	206, 499	Florin	Manta	212, 359
Dorian	Florescu	244	Vasile	Manta	33, 120, 256, 277, 570
Andrei	Franciuc	126	Dan Adrian	Marior	365
Pasi	Frănti	658	Lucian	Mastacan	190
Matteo	Gagliolo	250, 467	Mihaela-Hanako	Matcovschi	51, 244, 371
Marius	Gavrilescu	256	Monica	Mateescu	650
Morteza	Ghatee	262	Jován	Mérida	382
Narcis	Ghita	267	Pasquale de	Michele	178
Marc	Goniche	537	Eugenia	Minca	499
Juan	González-Gómez	628	Viorel	Minzu	93, 206, 499
Stijn	Goossens	250	Madalina	Mircioiu	475
Sorin Mihai	Grigorescu	273, 341	Letiția	Mirea	388
Eduard	Gröeller	256, 277	Radu	Miron	394
Sherif	Hammad	9	Dejan	Mitrovic	306
Ahmed	Hassan	9	Adrian	Moise	475
Paul	Herghelegiu	277	Florica	Moldoveanu	462
Fernando	Herrero-Carrón	628	Florin	Moldoveanu	273, 300, 487
Radu	Hobincu	168	Anca	Morar	462
Mihai	Hulea	394	Masoud	Mosayebi	262
Mircea	Hulea	282	Vlad	Muresan	400
Cristina	Huzum	288	Pal-Stefan	Murvay	406
Przemyslaw	Ignaciuk	294	Alex	Muscar	412
Sorin	Ilie	531	Andreea Maria	Neaca	418
Lucian Mihai	Itu	300	Mitica Iustinian	Neaca	418
Mirjana	Ivanović	306, 544, 644	Viorel	Negru	45
Mohammad Reza	Jahed-Motlagh	230	Maximilian	Nicolae	436
Yasushi	Kami	430	Mircea	Nitulescu	347, 353, 598, 628
Petros	Karamanakos	312	Eitaku	Nobuyama	430
Bahram	Karimi	262	Ann	Nowé	250, 467
Robin de	Keyser	218, 467	Mihai	Obada	144
Aleksandra	Klašnja-Milićević	644	Sorin	Olaru	620
Marius	Kloetzer	267	Virginia Ecaterina	Oltean	436
Martin	Knecht	318	Andreea Raluca	Olteanu	444
Y.	Kouhi	81	Alexandru	Onea	63
Walter	Kropatsch	570	Doina	Onu	149
Patrick	Kühtreiber	318	Mihaela	Oprea	450
Corneliu	Lazăr	87, 132, 149	Florin	Ostafi	110
Andrei	Leca	324	Anisoara	Oubraham	650
Florin	Leon	324	Doru	Panescu	456
Tiberiu	Letia	394	Georgios	Papafotiou	312
Rogelio	Lozano	671	Carlos	Pascal	456
Lucian	Luncean	330	Octavian	Pastravanu	244, 267, 371
Robert Gabriel	Lupu	336	Cristian	Perra	664
Nacer K.	M'Sirdi	505	Emil	Petre	138, 586
Gigel	Macesanu	273, 341	Lucian	Petrescu	462
Cristian	Mahulea	554	Lucian	Petrica	168
Laurentiu	Manafu	481	Emil	Petriu	194
Stefanos	Manias	312	Gregory	Pinte	218, 250, 467

Dan	Popescu	436, 592	Viorel Adrian	Stanciu	138
Dorin	Popescu	664	Roxana	Stanica	586
Dumitru	Popescu	162, 475	Alexandra-Iulia	Stinean	194
Elvira	Popescu	481	Florin	Stinga	592
Adrian	Postelnicu	300	Julian	Stoev	218, 467
Radu-Emil	Precup	194	Viorel	Stoian	347, 598
Stefan	Preitl	194	Constantin	Suciu	300
Dan	Puiu	487	Zhen	Sun	603
Alexandru Ionut	Pustianu	493	Ioan	Susnea	609
Mircea-Bogdan	Radac	194	Jan	Swevers	467
Adrian	Radaschin	499	Wim	Symens	250, 467
Miloš	Radovanović	544	Andrei	Tabarcea	658
Lala H.	Rajaoarisoa	505	Oleg	Tarabuta	614
George Constantin	Rascanu	511	Catalin-Stefan	Teodorescu	620
Vladimir	Rasvan	516	Jean-Marc	Thiriet	664
Cristina	Resceanu	522	Vladimir	Tiganas	144
Abdel	Rodríguez	250	Mihai	Timis	634
Cristian	Rotariu	526	Jorge	Torres	671
Jose-Luis	Rullan-Lara	671	Christoph	Traxler	318
Houman	Sadjadian	230	Andreea	Udrea	436
Liviu	Sandu	531	Florina	Ungureanu	336
Violeta	Sandu	638	Ionut	Urziceanu	628
Enrique	Santiago	537	Kevin van	Vaerenbergh	250
Alexandre	Sava	57	Alexandru	Valachi	582, 634
Miloš	Savić	544	Pablo	Varona	628
Liana Simona	Sbirna	650	Gabriela	Varvara	638
Sebastian	Sbirna	650	Grigore	Vasiliu	609
Raluca	Sbora	531	Boban	Vesin	644
Mihnea	Scafes	69	Milan	Vidakovic	306
Dorin	Sendrescu	550	George	Vieriu	120
Adriana	Serbencu	493	Gabriel	Vladut	650
Ghazanfar	Shahgholian	262	George	Voicu	224
Yasser	Shokry	9	Mihail	Voicu	371
Ioan	Silea	406	Karol	Waga	658
Manuel de	Silva	554	Tony	Ward	664
Vasile	Sima	564	Emmanuel	Witrant	537
Ionel Cristinel	Simion	444	Bart	Wyns	218, 467
Ioan Valentin	Sita	200	Hamed	Yahoui	664
Cristian	Smochina	570	Zhenyu	Yang	603
Razvan	Solea	115, 511, 576	Zizilia	Zamudio	671
Andrei	Stan	75, 582	Yu	Zhong	218, 467

LIST OF REVIEWERS

Bogdan-Ioan Aignatoaiei	Eva-Henrietta Dulf
Dorel Aiordachioaie	Stefan Dumbrava
Cristian Amarandei	Eugen Dumitrascu
Neculai Andrei	Tiberius Dumitriu
Gladiola Andruseac	Abhishek Dutta
Alexandru Archip	Lavinia Ferariu
Dragos Arotaritei	Markus Fiedler
Nicole M. Artner	Adrian Filipescu
Costin Badica	Mihaela Florescu
Eugen Balaban	Laurentiu Frangu
Marcelin-Iulian Balasca-Mihoci	Pasi Franti
Andreea E. Balau	Teohari Ganciu
Peter Baranyi	Eugen Ganea
Andrzej Bartoszewicz	Marius Gavrilescu
Nicu Bizdoaca	Arpad Gellert
Eugen Bobasu	Marian Gheorghe
Costin-Anton Boiangiu	Carmen Grigoras
Corneliu Botan	Eduard Groeller
Nicolae Botezatu	Ralf Habel
Remus Brad	Martin Haidacher
Catalin Braescu	Remus Gabriel Hanea
Florin Catalin Braescu	Paul Herghelegiu
Cristina Budaciu	Rui Huang
Zoran Budimac	Eugen Iancu
Adrian Burlacu	Anca Ion
Simona Caraiman	Clara Ionescu
Sergiu Caraman	Cosmin Ionete
Constantin Florin Caruntu	Mircea Ivanescu
Petru Cascaval	Mirjana Ivanovic
Emil Ceanga	Dragan Jankovic
Marius Cristian Cerlinca	Magnus Jonsson
Daniela Cernega	Ioannis Kazanidis
Konda Chevva	Robin De Keyser
Dorian Cojocaru	Vladimir Kharitonov
Cosmin Copot	Marius Kloetzer
Hariton Costin	Peter Korondi
Sorin Cotofana	Jacek Kucharski
Balázs Csébfalvi	Kristina Kunert
Mitica Craus	Vladimir Kurbalija
Vladimir Cretu	Corneliu Lazar
Valentin Cristea	Gheorghe Lazea
Nicoleta Cucu-Laurenciu	Florin Leon
Janetta Culita	Ioan Alfred Letia
Silvia Curteanu	Tamas Levente
Daniela Danciu	Robert Gabriel Lupu
Catalin Dimon	Camelia Maican
Constantin Catalin Dosoftei	Dan Mancas
Claudia-Adina Dragos	Lucian Mandache
Viorel Dugan	Vasile-Ion Manta

Stephan Mantler
 Cosmin Marcu
 Constantin Marin
 Lucian Mastacan
 Mihaela-Hanako Matcovschi
 Cristian Mihaescu
 Letitia Mirea
 Gabriel Mistelbauer
 Viorel Minzu
 Alin Moldoveanu
 Florica Moldoveanu
 Sabine Mondie
 Alex Muscar
 Gabriel Neagu
 Sergiu Nedevschi
 Mihai Negru
 Viorel Negru
 Adrian Nicolescu
 Mircea Nitulescu
 Aleksandra Nowacka-Leverton
 Iosif Olah
 Sorin Olaru
 Mihaela Oprea
 Alexandru Onea
 Florin Ostafi
 Cristea Pal
 Romeo Paduraru
 Doru Panescu
 Octavian Pastravanu
 Alina Patelli
 Lacra Pavel
 Emil Petre
 Ariadna Pletea
 Dan Popescu
 Dorin Popescu
 Dumitru Popescu
 Elvira Popescu
 Mihai Postolache
 Rodica Potolea
 Radu-Emil Precup
 Stefan Preitl
 Andrei Pricop
 Dorina Purcaru

Werner Purgathofer
 Gheorghe Puscasu
 Mircea-Bogdan Radac
 Milos Radovanovic
 Vladimir Rasvan
 Hemant Kumar Rath
 Cristian Rotariu
 Adriana Scarlat
 Dan Selisteanu
 Dorin Sendrescu
 Elena Serban
 Sridhar Seshagiri
 Katrin Sjoberg
 Vasile Sima
 Ioan Silea
 Radu Silion
 Constantin Siriteanu
 Francisk Sisak
 Katrin Sjöberg
 Cristian Smochina
 Razvan Solea
 Andrei Stan
 Radu Stefan
 Viorel Stoian
 Adrian-Mihail Stoica
 Cristina Stoica
 Julian Stöttinger
 Andy Tanase
 Razvan Tudor Tanasie
 Radu Tarca
 Catalin-Stefan Teodorescu
 Tiberiu Teodorescu
 Cornel Turcu
 Cristina Turcu
 Stefan Udristoiu
 Florina Ungureanu
 Gabriela Varvara
 Radu-Daniel Vatavu
 Lucian Vintan
 Mihail Voicu
 George R. Voicu
 Draguna Vrabie
 Mihai Horia Zaharia

LOCAL PARTNERS AND SPONSORS



National Authority for Scientific Research



Continental Automotive Iași



IBM Romania



**Embarcadero Technologies, Inc.
Iasi, Romania**



Centric IT Solutions Romania



The RED Point Iasi, Romania