



ICINCO 2010

7th International Conference on
Informatics in Control, Automation and Robotics

Proceedings

Volume 1

Funchal, Madeira - Portugal · 15 - 18 June, 2010

Sponsored by:



Co-sponsored by:



In Cooperation with:



ICINCO 2010

Proceedings of the
7th International Conference on
Informatics in Control, Automation and Robotics

Volume 1

Funchal, Madeira, Portugal

June 15 - 18, 2010

Co-Sponsored by

**INSTICC – Institute for Systems and Technologies of Information, Control
and Communication**

IFAC – International Federation of Automatic Control

In Cooperation with

AAAI – Association for the Advancement of Artificial Intelligence

WfMC – Workflow Management Coalition

APCA – Associação Portuguesa de Controlo Automático

**ACM SIGART – Association for Computing Machinery / Special Interest
Group on Artificial Intelligence**

Copyright © 2010 SciTePress – Science and Technology Publications
All rights reserved

Edited by Joaquim Filipe, Juan Andrade Cetto and Jean-Louis Ferrier

Printed in Portugal

ISBN: 978-989-8425-00-3

Depósito Legal: 311148/10

<http://www.icinco.org>
icinco.secretariat@insticc.org

BRIEF CONTENTS

INVITED SPEAKERS	IV
SPECIAL SESSION CHAIRS	IV
ORGANIZING AND STEERING COMMITTEES	V
PROGRAM COMMITTEE	VI
AUXILIARY REVIEWERS	IX
SPECIAL SESSION PROGRAM COMMITTEE	X
SELECTED PAPERS BOOK	X
FOREWORD	XI
CONTENTS	XIII

INVITED SPEAKERS

José Santos-Victor

Instituto Superior Técnico
Portugal

Alicia Casals

Institute for Bioengineering of Catalonia.IBEC and Universitat Politècnica de Catalunya.UPC
Spain

Bradley Nelson

Robotics and Intelligent Systems at ETH-Zürich
Switzerland

Wissama Khalil

Ecole Centrale de Nantes, IRCCyN
France

Oleg Gusikhin

Ford Research & Adv. Engineering
U.S.A.

John Hollerbach

University of Utah
U.S.A.

SPECIAL SESSION CHAIRS

SPECIAL SESSION ON INTELLIGENT VEHICLE CONTROLS & INTELLIGENT TRANSPORTATION SYSTEMS

Oleg Gusikhin, Ford Research & Adv. Engineering, U.S.A.

ORGANIZING AND STEERING COMMITTEES

CONFERENCE CHAIR

Joaquim Filipe, Polytechnic Institute of Setúbal / INSTICC, Portugal

PROGRAM CO-CHAIRS

Juan Andrade Cetto, Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Spain

Jean-Louis Ferrier, University of Angers, France

PROCEEDINGS PRODUCTION

Patrícia Alves, INSTICC, Portugal

Helder Coelhas, INSTICC, Portugal

Vera Coelho, INSTICC, Portugal

Andreia Costa, INSTICC, Portugal

Patricia Duarte, INSTICC, Portugal

Bruno Encarnação, INSTICC, Portugal

Mauro Graça, INSTICC, Portugal

Raquel Martins, INSTICC, Portugal

Liliana Medina, INSTICC, Portugal

Carla Mota, INSTICC, Portugal

Vitor Pedrosa, INSTICC, Portugal

Filipa Rosa, INSTICC, Portugal

José Varela, INSTICC, Portugal

CD-ROM PRODUCTION

Elton Mendes, INSTICC, Portugal

Pedro Varela, INSTICC, Portugal

GRAPHICS PRODUCTION AND WEBDESIGNER

Daniel Pereira, INSTICC, Portugal

SECRETARIAT

Marina Carvalho, INSTICC, Portugal

WEBMASTER

Sérgio Brissos, INSTICC, Portugal

PROGRAM COMMITTEE

Arvin Agah, The University of Kansas, U.S.A.

Alexandre Poznyak, CINVESTAV-IPN, Mexico

Andrew Adamatzky, University of the West of England, U.K.

Eugenio Aguirre, University of Granada, Spain

Hesham Alfares, King Fahd University of Petroleum and Minerals, Saudi Arabia

Adel Al-Jumaily, University of Technology, Sydney, Australia

Francesco Amigoni, Politecnico di Milano, Italy

Peter Arato, Budapest University of Technology and Economics, Hungary

Alejandro Hernandez Arieta, University of Zurich, Switzerland

Marco Antonio Arteaga, Universidad Nacional Autonoma de Mexico, Mexico

Vijanth Sagayan Asirvadam, Universiti Teknologi PETRONAS, Malaysia

T. Asokan, Indian Institute of Technology Madras, India

Ruth Bars, Budapest University of Technology and Economics, Hungary

Adil Baykasoglu, University of Gaziantep, Turkey

Karsten Berns, University of Kaiserslautern, Germany

Mauro Birattari, IRIDIA-CoDE, Université Libre de Bruxelles, Belgium

Christian Blum, Universitat Politècnica de Catalunya, Spain

Patrick Boucher, Supélec, France

Bernard Brogliato, INRIA, France

Kevin Burn, University of Sunderland, U.K.

Clifford Burrows, Innovative Manufacturing Research Centre, U.K.

Dídac Busquets, Universitat de Girona, Spain

Javier Fernandez de Canete, University of Malaga, Spain

Giuseppe Carbone, LARM - Laboratorio di Robotica e Meccatronica, Italy

J. L. Martins de Carvalho, Instituto de Sistemas e Robótica - Porto, Portugal

Alessandro Casavola, University of Calabria, Italy

Riccardo Cassinis, University of Brescia, Italy

Ratchatin Chanchareon, Chulalongkorn University, Thailand

Antonio Chella, Università di Palermo, Italy

Wen-Hua Chen, Loughborough University, U.K.

Graziano Chesi, University of Hong Kong, China

Sung-Bae Cho, Yonsei University, Korea, Republic of

Ryszard S. Choras, University of Technology & Life Sciences, Poland

Carlos Coello Coello, CINVESTAV-IPN, Mexico

António Dourado Correia, University of Coimbra, Portugal

José Boaventura Cunha, University of Trás-os-montes and Alto Douro, Portugal

Mingcong Deng, Okayama University, Japan

Guilherme DeSouza, University of Missouri, U.S.A.

Denis Dochain, Université Catholique de Louvain, Belgium

Tony Dodd, The University of Sheffield, U.K.

Venky Dubey, Bournemouth University, U.K.

Frederick Ducatelle, Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), Switzerland

Ashish Dutta, Indian Institute of Technology Kanpur, India

Petr Ekel, Pontifical Catholic University of Minas Gerais, Brazil

Atilla Elci, Middle East Technical University, Turkey

Ali Eydgahi, University of Maryland Eastern Shore, U.S.A.

PROGRAM COMMITTEE (CONT.)

Jean-marc Faure, Ecole Normale Supérieure de Cachan, France

Paolo Fiorini, Università degli Studi di Verona, Italy

Georg Frey, Saarland University, Germany

Wai-Keung Fung, University of Manitoba, Canada

Dragan Gamberger, Rudjer Boskovic Institute, Croatia

Andrea Garulli, Università di Siena, Italy

Ryszard Gessing, Silesian University of Technology, Poland

Lazea Gheorghe, Technical University of Cluj-Napoca, Romania

Paulo Gil, Universidade Nova de Lisboa, Portugal

Alessandro Giua, University of Cagliari, Italy

Luis Gomes, Universidade Nova de Lisboa, Portugal

Dongbing Gu, University of Essex, U.K.

Kevin Guelton, University of Reims Champagne-Ardenne, France

Maki K. Habib, The American University in Cairo, Egypt

Wolfgang Halang, Fernuniversität, Germany

Onur Hamsici, Qualcomm, U.S.A.

John Harris, University of Florida, U.S.A.

Inman Harvey, University of Sussex, U.K.

Dominik Henrich, University of Bayreuth, Germany

Suranga Hettiarachchi, Indiana University Southeast, U.S.A.

Victor Hinostroza, University of Ciudad Juárez, Mexico

Wladyslaw Homenda, Warsaw University of Technology, Poland

Guoqiang Hu, Kansas State University, U.S.A.

Joris Hulstijn, Vrije Universiteit, Amsterdam, The Netherlands

Fumiya Iida, Robot Locomotion Group, U.S.A.

Atsushi Imiya, IMIT Chiba University, Japan

Giovanni Indiveri, University of Salento, Italy

Mirjana Ivanovic, Faculty of Science, University of Novi Sad, Serbia

Sarangapani Jagannathan, Missouri University of Science and Technology, U.S.A.

Masoud Jamei, Simcyp Ltd, U.K.

Ping Jiang, The University of Bradford, U.K.

Graham Kendall, University of Nottingham, U.K.

DaeEun Kim, Yonsei University, Korea, Republic of

Won-jong Kim, Texas A&M University, U.S.A.

Israel Koren, University of Massachusetts, U.S.A.

Gerhard K. Kraetzschmar, Bonn-Rhein-Sieg University of Applied Sciences, Germany

Mianowski Krzysztof, Politechnika Warszawska, Poland

H. K. Lam, King's College London, U.K.

Alexander Lanzon, University of Manchester, U.K.

Kathryn J. De Laurentis, University of South Florida, U.S.A.

Kauko Leiviskä, University of Oulu, Finland

Zongli Lin, University of Virginia, U.S.A.

Guoping Liu, University of Glamorgan, U.K.

Jing-Sin Liu, Institute of Information Science, Academia Sinica, Taiwan

José Tenreiro Machado, Institute of Engineering of Porto, Portugal

Anthony Maciejewski, Colorado State University, U.S.A.

Frederic Maire, Queensland University of Technology, Australia

Om Malik, University of Calgary, Canada

Hervé Marchand, INRIA, France

Philippe Martinet, Lasmea, France

Rene V. Mayorga, University of Regina, Canada

PROGRAM COMMITTEE (CONT.)

Seán McLoone, National University of Ireland (NUI) Maynooth, Ireland

Prashant Mehta, University of Illinois at Urbana-Champaign, U.S.A.

Carlo Menon, Simon Fraser University, Canada

António Paulo Moreira, INESC Porto / FEUP, Portugal

Vladimir Mostyn, VSB - Technical University of Ostrava, Czech Republic

Rafael Muñoz-salinas, University of Cordoba, Spain

Kenneth Muske, Villanova University, U.S.A.

Andreas Nearchou, University of Patras, Greece

Luciana Nedel, Universidade Federal do Rio Grande do Sul (UFRGS), Brazil

Sergiu Nedeveschi, Technical University of Cluj-Napoca, Romania

Monica N. Nicolescu, University of Nevada, Reno, U.S.A.

Klas Nilsson, Lund University, Sweden

Urbano Nunes, University of Coimbra, Portugal

Manuel Ortigueira, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, Portugal

Selahattin Özcelik, Texas A&M University-Kingsville, U.S.A.

Igor Paromtchik, INRIA, France

Mario Pavone, University of Catania, Italy

Claudio de Persis, Sapienza University of Rome, Italy

D. T. Pham, Cardiff University, U.K.

Michael Piovoso, The Pennsylvania State University, U.S.A.

Raul Marin Prades, Jaume I University, Spain

José Ragot, Centre de Recherche en Automatique de Nancy, France

Jerzy Respondek, Silesian University of Technology, Poland

A. Fernando Ribeiro, Universidade do Minho, Portugal

Robert Richardson, University of Leeds, U.K.

Mihailo Ristic, Imperial College London, U.K.

Juha Röning, University of Oulu, Finland

Agostinho Rosa, IST, Portugal

Daniilo De Rossi, University of Pisa, Italy

Fariba Sadri, Imperial College London, U.K.

Mehmet Sahinkaya, University of Bath, U.K.

Priti Srinivas Sajja, Sardar Patel University, India

Abdel-badeeh Salem, Ain Shams University, Egypt

Marcello Sanguineti, University of Genova, Italy

Jurek Sasiadek, Carleton University, Canada

Sergio M. Savaresi, Politecnico di Milano, Italy

Carla Seatzu, University of Cagliari, Italy

Michael Short, Teesside University, U.K.

Silvio Simani, University of Ferrara, Italy

Dan Simon, Cleveland State University, U.S.A.

Olivier Simonin, INRIA - LORIA, France

Joaquin Sitte, Queensland University of Technology, Australia

Adam Slowik, Koszalin University of Technology, Poland

Andrzej Sluzek, Nanyang Technological University/Nicolaus Copernicus University, Singapore

Safeullah Soomro, Yanbu University College, Saudi Arabia

Stefano Squartini, Polytechnic University of Marche, Italy

Burkhard Stadlmann, University of Applied Sciences Wels, Austria

Chun-Yi Su, Concordia University, Canada

Ryszard Tadeusiewicz, AGH University of Science and Technology, Poland

PROGRAM COMMITTEE (CONT.)

Kazuya Takeda, Nagoya University, Japan

Daniel Thalmann, VR Lab EPFL, Switzerland

N. G. Tsagarakis, Istituto Italiano di Tecnologia, Italy

Avgoustos Tsinakos, University of Kavala Institute of Technology, Greece

Antonios Tsourdos, Cranfield University (Cranfield Defence and Security), U.K.

Angel Valera, Universidad Politécnica de Valencia, Spain

Eloisa Vargiu, University of Cagliari, Italy

Annamaria R. Varkonyi-koczy, Obuda University, Hungary

Ramiro Velazquez, Universidad Panamericana, Mexico

Damir Vrancic, Jožef Stefan Institute, Slovenia

Bernardo Wagner, Leibniz Universität Hannover, Germany

Dianhui Wang, La Trobe University, Australia

James Whidborne, Cranfield University, U.K.

Sangchul Won, Pohang University of Science and Technology, Korea, Republic of

Qishi Wu, University of Memphis, U.S.A.

Marek Zaremba, Université du Québec (UQO), Canada

Janan Zaytoon, University of Reims Champagne Ardennes, France

Primo Zingaretti, Università Politecnica delle Marche, Italy

AUXILIARY REVIEWERS

Lounis Adouane, LASMEA, UMR CNRS 6602, France

Iman Awaad, Bonn-Rhein-Sieg University, Germany

Yao Chen, Institute of Systems Sciences, Chinese Academy of Sciences, China

Rohit Chintala, Texas A and M University, India

Jonathan Courbon, LASMEA, France

Kun Deng, University of Illinois at Urbana Champaign, U.S.A.

Anibal Ferreira, University of Porto, Portugal

Leonardo Fischer, Universidade Federal do Rio Grande do Sul - UFRGS, Brazil

Fernando Fontes, Faculty of Engineering, University of Porto, Portugal

Ronny Hartanto, Bonn-Rhein-Sieg University of Applied Sciences, Germany

Laurent Harter, France

Nico Hochgeschwender, ESG GmbH, Germany

Ana Lopes, Institute of Systems and Robotics - University of Coimbra, Portugal

Jan Paulus, Bonn-Rhein-Sieg University oAS, Germany

Cristiano Presmebida, Institute of System and Robotics, Portugal

Michael Reckhaus, Bonn-Rhein-Sieg University, Germany

Lynda Seddiki, Paris 8 University, France

Renato Silveira, Universidade Federal do Rio Grande do Sul (UFRGS), Brazil

Yu Sun, University of Illinois, Urbana-Champaign, U.S.A.

C. Renato Vázquez, Universidad de Zaragoza, Spain

Huibing Yin, University of Illinois at Urbana-champaign, U.S.A.

SPECIAL SESSION PROGRAM COMMITTEE

SPECIAL SESSION ON INTELLIGENT VEHICLE CONTROLS & INTELLIGENT TRANSPORTATION SYSTEMS

T. J. Giuli, Ford Motor Company, U.S.A.

Oleg Gusikhin, Ford Research & Adv.
Engineering, U.S.A.

Erica Klampfl, Ford Research & Advanced
Engineering, U.S.A.

Ilya Kolmanovsky, University of Michigan, U.S.A.

Anatoli Koulinitch, ArvinMeritor, U.S.A.

Perry MacNeille, Ford Motor Company, U.S.A.

Danil Prokhorov, Toyota Tech Center, U.S.A.

Christian Ress, Ford Motor Company, Germany

SELECTED PAPERS BOOK

A number of selected papers presented at ICINCO 2010 will be published by Springer-Verlag in a LNEE Series book. This selection will be done by the Conference Chair and Program Co-chairs, among the papers actually presented at the conference, based on a rigorous review by the ICINCO 2010 Program Committee members.

FOREWORD

This book contains the proceedings of the 7th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2010) which was sponsored by the Institute for Systems and Technologies of Information, Control and Communication (INSTICC) and held in Funchal, Madeira - Portugal. ICINCO 2010 was co-sponsored by the International Federation for Automatic Control (IFAC) and held in cooperation with the Association for the Advancement of Artificial Intelligence (AAAI), the Workflow Management Coalition (WfMC), the Portuguese Association for Automatic Control (APCA) and the Association for Computing Machinery (ACM SIGART).

The ICINCO Conference Series has now consolidated as a major forum to debate technical and scientific advances presented by researchers and developers both from academia and industry, working in areas related to Control, Automation and Robotics that benefit from Information Technology.

In the Conference Program we have included oral presentations (full papers and short papers) and posters, organized in three simultaneous tracks: “Intelligent Control Systems and Optimization”, “Robotics and Automation” and “Systems Modeling, Signal Processing and Control”. We have included in the program six plenary keynote lectures, given by internationally recognized researchers, namely - José Santos-Victor (Instituto Superior Técnico, Portugal), Alícia Casals (Institute for Bioengineering of Catalonia.IBEC and Universitat Politècnica de Catalunya.UPC, Spain), Bradley Nelson (Institute of Robotics and Intelligent Systems at ETH-Zürich, Switzerland), Wisama Khalil (Ecole Centrale de Nantes, IRCCyN, France), Oleg Gusikhin (Ford Research & Adv. Engineering, U.S.A.) and John Hollerbach (University of Utah, U.S.A.).

The meeting is complemented with one satellite workshop, the International Workshop on Artificial Neural Networks and Intelligent Information Processing (ANNIIP), and one Special Session on Intelligent Vehicle Controls & Intelligent Transportation Systems (IVC & ITS).

ICINCO received 320 paper submissions, not including those of the workshop or the special session, from 57 countries, in all continents. To evaluate each submission, a double blind paper review was performed by the Program Committee. Finally, only 142 papers are published in these proceedings and presented at the conference. Of these, 94 papers were selected for oral presentation (27 full papers and 67 short papers) and 48 papers were selected for poster presentation. The full paper acceptance ratio was 8%, and the oral acceptance ratio (including full papers and short papers) was 29%. As in previous editions of the Conference, based on the reviewer’s evaluations and the presentations, a short list of authors will be invited to submit extended versions of their papers for a book that will be published by Springer with the best papers of ICINCO 2010.

Conferences are also meeting places where collaboration projects can emerge from social

contacts amongst the participants. Therefore, in order to promote the development of research and professional networks the Conference includes in its social program a Conference and Workshop Social Event & Banquet in the evening of June 17 (Thursday).

We would like to express our thanks to all participants. First of all to the authors, whose quality work is the essence of this Conference. Next, to all the members of the Program Committee and auxiliary reviewers, who helped us with their expertise and valuable time. We would also like to deeply thank the invited speakers for their excellent contribution in sharing their knowledge and vision. Finally, a word of appreciation for the hard work of the INSTICC team; organizing a conference of this level is a task that can only be achieved by the collaborative effort of a dedicated and highly capable team.

Commitment to high quality standards is a major aspect of ICINCO that we will strive to maintain and reinforce next year, including the quality of the keynote lectures, of the workshops, of the papers, of the organization and other aspects of the conference. We look forward to seeing more results of R&D work in Informatics, Control, Automation and Robotics at ICINCO 2011.

Joaquim Filipe

Polytechnic Institute of Setúbal / INSTICC, Portugal

Juan Andrade Cetto

Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Spain

Jean-Louis Ferrier

University of Angers, France

CONTENTS

INVITED SPEAKERS

KEYNOTE SPEAKERS

BIOINSPIRED ROBOTICS AND VISION WITH HUMANOID ROBOTS <i>José Santos-Victor</i>	IS-5
HUMAN - Robot Cooperation Techniques in Surgery <i>Alicia Casals</i>	IS-7
MAKING MICROROBOTS MOVE <i>Bradley Nelson</i>	IS-13
DYNAMIC MODELING OF ROBOTS USING RECURSIVE NEWTON-EULER TECHNIQUES <i>Wissama Khalil</i>	IS-19
EMOTIVE DRIVER ADVISORY SYSTEM <i>Oleg Gusikhin</i>	IS-33
FINGERTIP FORCE MEASUREMENT BY IMAGING THE FINGERNAIL <i>John Hollerbach</i>	IS-35

INTELLIGENT CONTROL SYSTEMS AND OPTIMIZATION

FULL PAPERS

A MORPHING WING USED SHAPE MEMORY ALLOY ACTUATORS NEW CONTROL TECHNIQUE WITH BI-POSITIONAL AND PI LAWS OPTIMUM COMBINATION - Part 1: Design Phase <i>Teodor Lucian Grigorie, Andrei Vladimir Popov, Ruxandra Mihaela Botez, Mahmoud Mamou and Youssef Mébarki</i>	5
A MORPHING WING USED SHAPE MEMORY ALLOY ACTUATORS NEW CONTROL TECHNIQUE WITH BI-POSITIONAL AND PI LAWS OPTIMUM COMBINATION - Part 2: Experimental Validation <i>Teodor Lucian Grigorie, Andrei Vladimir Popov, Ruxandra Mihaela Botez, Mahmoud Mamou and Youssef Mébarki</i>	13
PREDICTION OF TEMPERATURE INSIDE A REFRIGERATED CONTAINER IN THE PRESENCE OF PERISHABLE GOODS <i>Javier Palafox-Albarrán, Reiner Jedermann and Walter Lang</i>	20
DYNAMIC CONTROL OF MOBILE AD-HOC NETWORKS - Network Protocol Parameter Adaptation using Organic Network Control <i>Sven Tomforde, Björn Hurling and Jörg Hähner</i>	28
A PRACTICAL METHOD FOR SELF-ADAPTING GAUSSIAN EXPECTATION MAXIMIZATION <i>Nicola Greggio, Alexandre Bernardino and José Santos-Victor</i>	36
PATTERNS FOR TEMPORAL REQUIREMENTS ENGINEERING - A Level Crossing Case Study <i>A. Mekki, M. Ghazel and A. Toguyeni</i>	45

A PASSIVITY-BASED APPROACH TO DEPLOYMENT IN MULTI-AGENT NETWORKS <i>Heath LeBlanc, Emeka Eyisi, Nicholas Kottenstette, Xenofon Koutsoukos and Janos Sztipanovits</i>	53
 SHORT PAPERS	
CHARACTERISTICS OF DEFINING HYPERPLANES OF CONSTANT RETURNS TO SCALE TECHNOLOGY IN DEA <i>Sevan Sohraiee</i>	65
VERIFICATION OF AN INDUSTRIAL COMPUTER NETWORK OF HIGH RISK OPERATION PLANTS - A NPP APCS Example <i>Vitaly Promyslov and Stanislav Masolkin</i>	74
3D PATH PLANNING FOR UNMANNED AERIAL VEHICLES USING VISIBILITY LINE BASED METHOD <i>Rosli Omar and Dawei Gu</i>	80
NEW PROPOSAL FOR A MULTI-OBJECTIVE TECHNIQUE USING TRIBES AND TABU SEARCH <i>Nadia Smairi, Sadok Bouamama, Khaled Ghedira and Patrick Siarry</i>	86
ADAPTIVE FUZZY SLIDING MODE CONTROL FOR UNCERTAIN NONLINEAR SYSTEMS AGAINST ACTUATOR FAULTS <i>Meriem Benbrahim, Najib Essounbouli, Abdelaziz Hamzaoui and Ammar Betta</i>	92
A CONTEXT-AWARE ADAPTATION SYSTEM FOR SPATIAL AUGMENTED REALITY PROJECTIONS <i>Anne Wegerich, Julian Adenauer, Jeronimo Dzaack and Matthias Roetting</i>	98
EFFICIENT ORGANIZATION OF TRUCK PLATOONS BY MEANS OF DATA MINING - Application of the Data Mining Technique for the Planning and Organization of Electronically Coupled Trucks <i>Ralph Kunze, Richard Ramakers, Klaus Henning and Sabina Jeschke</i>	104
MORPHING WING REAL TIME OPTIMIZATION IN WIND TUNNEL TESTS <i>Andrei V. Popov, Lucian T. Grigorie, Ruxandra Botez, Mahmood Mamou and Youssef Mébarki</i>	114
GENETIC ALGORITHM VERSUS ANT COLONY OPTIMIZATION ALGORITHM - Comparison of Performances in Robot Path Planning Application <i>Nohaidda Binti Sariff and Norlida Buniyamin</i>	125
A MULTI-ESTIMATION SCHEME FOR CONTROLLING THE BEVERTON-HOLT EQUATION IN ECOLOGY <i>S. Alonso-Quesada and M. De La Sen</i>	133
STRONG STABILIZATION BY OUTPUT FEEDBACK CONTROLLERS FOR INPUT-DELAYED LINEAR SYSTEMS <i>Baozhu Du, James Lam and Zhan Shu</i>	141
SMART HOME - From User's Behavior to Prediction of Energy Consumption <i>Lamis Hawarah, Mirieille Jacomino and Stephane Ploix</i>	147
A SPATIAL ONTOLOGY FOR HUMAN-ROBOT INTERACTION <i>Lamia Belouaer, Maroua Bouzid and Abdel-Allah Mouaddib</i>	154
REINFORCEMENT LEARNING FOR ROBOT CONTROL USING PROBABILITY DENSITY ESTIMATIONS <i>Alejandro Agostini and Enric Celaya</i>	160

PARAMETER TUNING BY SIMPLE REGRET ALGORITHMS AND MULTIPLE SIMULTANEOUS HYPOTHESIS TESTING <i>Amine Bourki, Matthieu Coulm, Philippe Rolet, Olivier Teytaud and Paul Vayssière</i>	169
MASTERING INTELLIGENT CLOUDS - Engineering Intelligent Data Processing Services in the Cloud <i>Sergiy Nikitin, Vagan Terziyan and Michal Nagy</i>	174
EVALUATING MAXIMUM TRANSMISSION UNRELIABILITY IN PERSISTENT CSMA PROTOCOL <i>Dariusz Kościelnik and Marek Miskowicz</i>	182
STRUCTURE-PRESERVING ALGORITHMS FOR DISCRETE-TIME ALGEBRAIC MATRIX RICCATI EQUATIONS <i>Vasile Sima</i>	187
CONTROL OF NETWORKED SYSTEMS CONTAINING MULTIPLE AGENTS <i>Jose B. Cruz Jr., Gregory Tangonan, Raymond R. Tan, Nathaniel Libatique, Fabian M. Dayrit and Alvin B. Culaba</i>	193
USING MODELICA MODELLING LANGUAGE FOR PHYSICAL PLANT PARAMETERS EVALUATION AND OPTIMIZATION - A Case Study <i>Eurico Seabra and José Machado</i>	199
 POSTERS	
FUZZY FREQUENCY RESPONSE FOR UNCERTAIN DYNAMIC SYSTEMS <i>Carlos Cesar Teixeira Ferreira and Ginalber Luiz de Oliveira Serra</i>	209
HARDWARE-SOFTWARE CODESIGN OF FUZZY CONTROL SYSTEMS USING FPGAS <i>E. del Toro, S. Sánchez-Solano, M. Brox and A. J. Cabrera</i>	213
SEGMENTATION OF THE SOLID BED IN INFRARED IMAGE SEQUENCES OF ROTARY KILNS <i>Patrick Waibel, Jörg Matthes and Hubert Keller</i>	217
VOICE FEEDBACK CONTROL WITH SUBJECT-IN-THE-LOOP <i>Dan Neculescu, Lin Zhang, Elisha Pruner and Jerzy Sasiadek</i>	221
ON SOME PECULIARITIES OF CLUSTER ANALYSIS OF PERIODIC SIGNALS <i>V. Znak</i>	226
INTERACTION OF A FLEXIBLE ROBOT WITH ITS ENVIRONMENT <i>Michael Kastner, Hubert Gatringer, Hartmut Bremer, Martin Ramsauer and Paolo Ferrara</i>	230
EXPERIMENTING WITH AUTONOMOUS CALIBRATION OF A CAMERA RIG ON A VISION SENSOR NETWORK <i>Kyung min Han, Yuanqiang Dong and Guilherme DeSouza</i>	234
STOCHASTIC OPTIMIZATION FOR ENVIRONMENTALLY POWERED WSNS USING MDP MODELS WITH MULTI-EPOCH ACTIONS <i>Alexandru E. Şuşu</i>	238
A FRAMEWORK BASED ON A HIGH CONCEPTION LEVEL TO GENERATE CONFIGURATIONS IN PRODUCTION SYSTEMS <i>Marwa Kanso, Pascal Berruet and Jean-luc Philippe</i>	244

AN ADAPTABLE ARCHITECTURE FOR INTELLIGENT CONVEYORS <i>Dennis Ommen, Jens Kamenik, Carsten Beth, Jan C. Busch, Alexander Kulas, Eckhard Cramer and Axel Hahn</i>	249
ENGINEERING TIME IN AN ONTOLOGY FOR POWER SYSTEMS THROUGH THE ASSEMBLING OF MODULAR ONTOLOGIES <i>Jorge Santos, Luís Braga and Anthony G. Cohn</i>	255
ASSEMBLY SYSTEMS FOR LOW PRODUCT DEMAND - Estimation of Final Results <i>Waldemar Grzechca</i>	259
CONTROL STRATEGY OF CONSTANT MILLING FORCE SYSTEM AND METAL REMOVAL RATE MAXIMIZATION <i>Franc Cus, Joze Balic and Uros Zuperl</i>	265
THE VALUE OF INFORMATION SHARING IN A SERIAL SUPPLY CHAIN WITH CENTRALISED AND DECENTRALISED DECISION <i>Mansour Rached, Zied Bahroun, Belhassen Zouari, Armand Baboli and Jean-Pierre Campagne</i>	269
SPECIAL SESSION ON INTELLIGENT VEHICLE CONTROLS & INTELLIGENT TRANSPORTATION SYSTEMS	
FULL PAPERS	
WHEELED-ROBOT NAVIGATION WITH VELOCITY UPDATING ON ROUGH TERRAINS <i>Farid García and Matías Alvarado</i>	277
VEHICLE ROUTING TO MINIMIZE MIXED-FLEET FUEL CONSUMPTION AND ENVIRONMENTAL IMPACT <i>O. Gusikhin, P. MacNeille and A. Cohn</i>	285
POSTERS	
MODEL-BASED FEATURE EXTRACTION FOR ASSESSMENT OF DRIVER-RELATED FATIGUE <i>Damián Alvarez, Álvaro Orozco and Augusto Salazar</i>	295
AUTHOR INDEX	299

INVITED SPEAKERS

KEYNOTE SPEAKERS

BIOINSPIRED ROBOTICS AND VISION WITH HUMANOID ROBOTS

José Santos-Victor

Instituto Superior Técnico, Lisboa, Portugal

Abstract: In this talk, I will describe recent results on exploring recent results from neurophysiology and developmental psychology for the design of humanoid robot technologies. The outcome of this research is twofold: (i) using biology as an inspiration for more flexible and sophisticated robotic technologies and (ii) contribute to the understanding of human cognition by developing biologically plausible (embodied) models and systems.

One application area is the domain of video surveillance and human activity recognition. We will see how recent findings in neurophysiology (the discovery of the mirror neurons) suggest that both action understanding and execution are performed by the same brain circuitry. This might explain how humans can so easily (apparently) understand the actions of other individuals, which constitutes the building block of non-verbal communication first and then, language acquisition and social learning.

The second aspect to be addressed is the use of development as a methodological approach for building complex humanoid robots. This line of research is inspired after the human cognitive and motor development, a pathway that allows newborns to progressively acquire new skills and develop new learning strategies. In engineering terms, this may be a way not only to structure the sensed data but also to master the complexity of the interaction with the physical world with a sophisticated body (sensing and actuation).

During the talk, I will provide examples with several humanoid platforms used for this research: Baltazar is a humanoid torso we developed to study sensorimotor coordination and cognition; the latest results are implemented in the iCub humanoid robot, for which we designed the head, face and body covers as well as the attention and affordance learning system.

BRIEF BIOGRAPHY

José Santos-Victor received the PhD degree in Electrical and Computer Engineering in 1995 from Instituto Superior Técnico (IST - Lisbon, Portugal), in the area of Computer Vision and Robotics. He is an Associate Professor with "Aggregation" at the Department of Electrical and Computer Engineering of IST and a researcher of the Institute of Systems and Robotics (ISR) and heads the Computer and Robot Vision Lab - VisLab.

He is the scientific responsible for the participation of IST/ISR in various European and National research projects in the areas of Computer Vision and Robotics. His research interests are in the areas of Computer and Robot Vision, particularly in the relationship between visual perception and the control of action, biologically inspired vision and robotics, cognitive vision and visual controlled (land, air and underwater) mobile robots.

Prof. Santos-Victor was an Associated Editor of the IEEE Transactions on Robotics and the Journal of Robotics and Autonomous Systems.

HUMAN

Robot Cooperation Techniques in Surgery

Alicia Casals

Institute for Bioengineering of Catalonia (IBEC), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
alicia.casals@upc.edu

Keywords: Medical Robotics, Human Robot Interaction, Human Machine Interfaces, Surgical Robots.

Abstract: The growth of robotics in the surgical field is consequence of the progress in all its related areas, as: perception, instrumentation, actuators, materials, computers, and so. However, the lack of intelligence of current robots makes teleoperation an essential means for robotizing the Operating Room (OR), helping in the improvement of surgical procedures and making the best of the human-robot couple, as it already happens in other robotic application fields. The assistance a teleoperated system can provide is the result of the control strategies that can combine the high performance of computers with the surgeon knowledge, expertise and will. In this lecture, an overview of teleoperation techniques and operating modes suitable in the OR is presented, considering different cooperation levels. A special emphasis will be put on the selection of the most adequate interfaces currently available, able to operate in such quite special environments.

1 INTRODUCTION

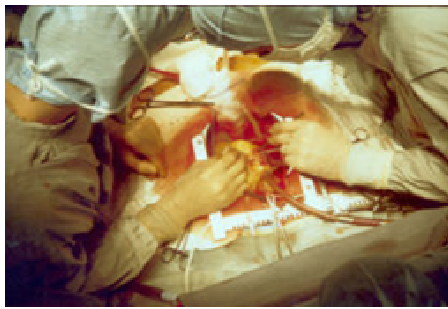
Technological evolution has continuously been introducing new equipments and changes in the Operating Room. Technology does not only affect real surgical interventions, but it also has a bear on all diagnosis and planning strategies, according to the diagnosed pathology. The history of surgery has suffered a continuous evolution through which three significant phases can be identified. They can be summarized in fig.1. From the practice of open surgery, fig. 1 a), in which surgeons get in touch directly with the patient organs or corresponding body parts, as purely manual actuation, the advent of new instruments and visualization techniques opened the era of minimally invasive surgery. Instruments with long handles allow entering the body through natural holes and small incisions over the patient fig. 1.b). This image shows the common scenario in laparoscopic surgery. At this stage, surgeons rely on instruments and specific equipment to perform surgery. The era of surgical robotics emerges as these instruments acquire new performances, or others appear in the scene, fig. 1.c).

New surgical procedures, not conceivable several decades ago, are more complex and require much higher performances. To face the challenges this kind of surgery relies on robots cooperating with

humans, so as to extract the best of both of them. Humans provide intelligence and decision making while robots contribute with their precision, computing capabilities and no tiredness. In this context, with human and machines sharing the working scene, and the task itself, more powerful interfaces and interaction means become necessary.

Both, cooperation and interface requirements will depend on the typology of surgery. Speaking in terms of robotics technology and considering that technological needs vary enormously with the kind of surgery, surgery can be classified in: microsurgery, neurosurgery, intracavity and orthopedic, and percutaneous and transcutaneous interventions. As significant distinction among them, some characteristics, as the kind of tissue (hard or soft) or the body parts undergoing surgery, are to be considered.

Since hard tissues are able to maintain its shape, when they can be immobilized some techniques applied in industry can be exported to surgery, otherwise, a tracking system to dynamically determine the changing reference frames is required. Soft tissues present the problem of deformability, making robot operation more complex. In this case, teleoperation is an alternative solution.



a)



b)



c)

Figure 1: Evolution of surgical procedures. a) open surgery, b) minimally invasive surgery, c) robot assisted surgery.

2 HUMAN ROBOT COOPERATION MODES

Human robot cooperation is implemented by means of teleoperation, therefore, a master device in the surgeon side controls a slave arm, a teleoperated robot, patient side. In between, a computer implements the required assistive functions that enhance human capabilities, resulting in a “super surgeon”. Such assistive functions can be a change of scale, defining constraints within the working space, tremor reduction and movement compensation (breathing or heart beating) and so. The surgeon can be located in a close position, or in

any other location, a few meters or some kilometers away. Fig. 2 shows a schema of such system.

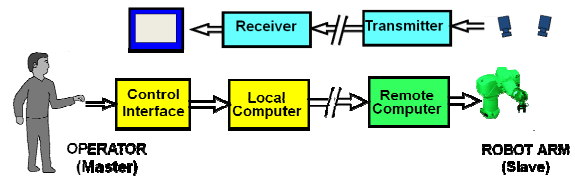


Figure 2: Schema of a teleoperation system.

However, the surgeon could also be in close contact with the robot, which guides and supervises his or her actions. In this cooperative mode, comanipulation, the surgeon hand and the robot end effector move simultaneously holding the surgical instrument. Working in these conditions, a change of scale or movement compensation is not an issue, but assisted teleoperation allows establishing constraints, virtual fixtures, operating over reference frames either fixed or floating over the patient anatomy. Comanipulation also allows directly perceiving both, images and the operating environment, what is especially useful in orthopedic surgery. The definition of virtual fixtures during the surgical planning facilitates a safer operation reducing the surgeon stress in critical interventions. Working with this configuration the robot itself behaves as a *haptic* device, Fig. 3.

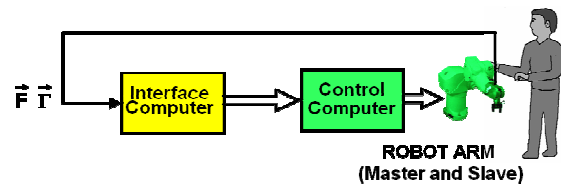


Figure 3: Schema of a comanipulation system.

The level of cooperation can also vary with the typology of the surgical interventions since the level of preplanning and programming varies depending on the predictability of the intervention. Three levels can be considered: manual guidance, supervised guidance and autonomous control.

The role of the interface in such cooperation systems is crucial as the surgeon cannot pay much attention to the robotic system, but to the patient and the own surgical procedure. A schema of the characteristics an interface should provide is shown in fig. 4.

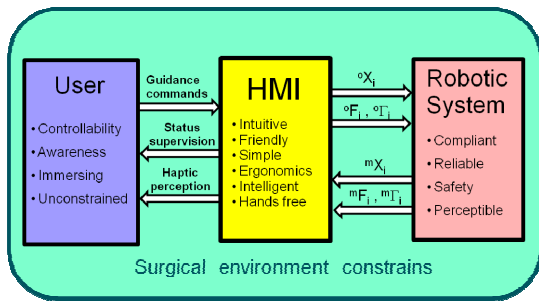


Figure 4: Schema of the characteristics of a teleoperation interface.

3 ROBOT ARCHITECTURES

Different surgical specialties work along different scale ranges and with different working requirements. Considering the variety of working conditions and from the robotics point of view, surgical procedures can be classified as follows: microsurgery, neurosurgery, transcutaneous, percutaneous, intracavities interventions and orthopedics. In what follows the main characteristics of each of them are described.

3.1 Microsurgery

Most interventions requiring microsurgery performances are related to sewing nerves in transplants or to ophthalmology. In this specialty, one or more high precision and high accessibility 6 Degrees of Freedom (DoF) arms are necessary. Additional DoF might be required to increase accessibility. As teleoperation assistance functions, a change of scale between the master and the slave increases the achievable precision as the application requires. Compared to former manipulators or holding devices used in classical manual practice, robots substitute them with advantage

3.2 Neurosurgery

Interventions in the skull also require high precision. However, its accessibility requirements are no so demanding since insertions are applied through incisions done just over the target area. In this case, movement compensation or floating reference frames are not needed as the skull can be fixed, with stereotaxis devices.

3.3 Transcutaneous

Some interventions can be performed through the

skin and soft tissues using radiation focused over the target area. This radiation can be of different types: RX, Gama, or High Intensity focused US (HIFU). In such minimally invasive technique, 5DoF are enough to focus the therapeutic beam over a point in a 3D space, with any orientation.

3.4 Percutaneous

Relatively simple interventions as biopsy, aspiration, ablations or releasing therapeutic payload, are more and more used due to the few invasiveness of the technique. This technique implies inserting needles with high precision, to produce the advance and drilling movements when the needle is manually oriented into the insertion point. 5 DoF are necessary to place and orient the needle with a robot.

Taking advantage of this minimally invasive surgery or intervention, deflection and guided probes can be used to reach areas not easily reachable. Operating through natural ways, as the arteries, this technique is being used successfully to treat brain aneurisms.

3.5 Intracavity Interventions

When the intervention cannot be carried out by means of needles and more versatile instruments are needed with two, three or four DoF, endoscopic techniques are required. There are two endoscopic techniques, one based on the use of natural orifices (NOTES) and the second based on small incisions to access the abdominal and thoracic cavity (laparoscopy) or the joints between bones (arthroscopy).

These techniques were introduced in the seventies operating the instruments manually. At present, these instruments can be guided by teleoperated robots, so as they can take advantage of assisted teleoperation techniques. These robots should be multiarm (2, 3 or 4), each with several DoF, not only for tool positioning but also to increase accessibility and to make the pose of each arm compatible with the patient in the operating table.

NOTES are used in intra vaginal interventions, ear, laparoscopy for abdominal, prostate, heart, gynecology or arthroscopy, knee specially.

3.6 Orthopedics

Orthopedic surgery uses as end effectors drilling, cutting or milling tools to operate over bone tissues. These techniques are oriented to bone repair either

with prosthesis implants, subjecting or immobilizing boards or to reconstruct bones with grafts after oncologic surgery, for example.

For these procedures CAD/CAM techniques can be used, with similar methods than those used in industry. Reference frames registration between anatomical elements in the operating table and the CAD model previously obtained from CT images are used to make task planning possible.

4 SENSOR REQUIREMENTS

Based on these different scenarios, robotics requires different kind of sensors to be able to implement the required control strategies. Two kinds of sensors are needed: 3D geometrical positioning sensors (navigators) and physical interaction characteristics (force and torque sensors).

Positioning anatomical parts in the 3D space is not simple, especially when dealing with not immobilized rigid elements, soft, deformable or rhythmically moving tissues. The success of surgical robots rely on their capability for adequately sensing positions either using physical contact sensors (optical or magnetic techniques) or remote sensing, specially vision. Current limitations of computer vision strongly condition its advances.

Apart from being able to control the robot, not only geometrical strategies are necessary to generate trajectories, but additional force control techniques are required to avoid injures, as for instance, necrosis. On the other hand, force sensors provide the information required to generate *haptic* information to be feedback to the surgeon. .

5 INTERFACES REQUIREMENTS

The requirements of an interface for surgical applications does not imply uniquely the interpretation of human will to control the teleoperated robot, but also to provide some feedback of the task going on to the surgeon. Thus, an interface constitutes a complete system, fig. 5, consisting of master devices adequate for every specific kind of surgery, actuators to feedback information to the surgeon, monitoring devices, and the computing power to process the information coming both, from the teleoperated system to provide the adequate information and from the

human operator to provide the adequate control orders.

The schema of fig.5 shows that an interface can be a complete system that in some environments should provide certain intelligence level and, as indicated in the schema, even generate synthetic information to improve human operator's perception.

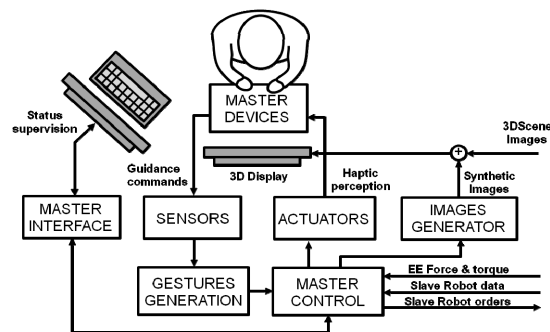


Figure 5: Schema of the master station in a teleoperated system.

Besides considering the best kind of master devices, either classical manual devices or hand free systems, the information coming from the robot side are key to achieve an efficient and smooth interaction. Feedback information can be either visual: direct images, augmented images or virtual, besides other kind of numerical and graphic data; or haptic, based on the force data measured in the robot working environment.

6 CONCLUSIONS

Interfaces are key components in teleoperated systems, or in robotic systems that work in close cooperation with humans. In the field of surgery, the surgeon faces the problem of dealing with complex systems while they are performing their own job, surgery. On the other side, their specialty is far from informatics and mechatronic systems. Thus, the design of a human robot interface in such fields should consider the three parts that compose the working environment, as shown in fig. 4, the user, the interface and the robotic system. While the surgeon has to pay complete attention to the intervention itself, the interface should provide the means of reacting to the humans' will, to interpret their needs, and to supply any kind of information that can help them to take decisions.

Teleoperation and its interface with the human operator can take different configurations according

to the application needs, considering both, the distance from the master to the slave and the typology of the application. In each case, the availability of the required teleoperation assistance functions to improve human and system performances is essential. Together with such assistance, the information fed back to the user, visual, haptic or even sound, can be intelligently processed to constitute a significant help for the whole process.

BRIEF BIOGRAPHY

Alicia Casals is professor at the Technical University of Catalonia (UPC), in the Automatic Control and Computer Engineering Department. She is currently leading the research group on Robotics and Medical imaging Program of the Institute for Biomedical Engineering of Catalonia, and is member of the research group GRINS: Intelligent Robotics and Systems at UPC. The research is oriented to improve human robot interaction through multimodal perception, focused mainly in the area of medical robotics. In this field she is working both in rehabilitation, assistance and surgical applications. Her background is in Electrical and Electronic Engineering and PhD in Computer Vision. From 2001 to 2008 she was the coordinator of the Education and Training key area within Euron, the Network of Excellence: European Robotics Network, and RAS Vice President for Membership in the period 2008-2009. From the developed research projects she won different awards, Award to a social invention (Mundo Electrónico), International Award *Barcelona'92* (Barcelona City_Hall), *Ciutat de Barcelona* Award 1998 (Barcelona City_Hall), and *Narcis Monturiol* Medal from the Catalan Government as recognition of the research trajectory 1999. From 2007 Prof. Casals is member of the *Institut d'Estudis Catalans*, the Academy of Catalonia.

MAKING MICROROBOTS MOVE

Bradley J. Nelson

Institute of Robotics and Intelligent Systems, ETH Zurich, Zurich, Switzerland

Keywords: Nanorobotics, Microrobotics, Nanocoils, Magnetic actuation.

Abstract: Our group has recently demonstrated three distinct types of microrobots of progressively smaller size that are wirelessly powered and controlled by magnetic fields. For larger scale microrobots, from 1mm to 500 μm , we microassemble three dimensional devices that precisely respond to torques and forces generated by magnetic fields and field gradients. In the 500 μm to 200 μm range, we have developed a process for microfabricating robots that harvest magnetic energy from an oscillating field using a resonance technique. At even smaller scales, down to micron dimensions, we have developed microrobots we call Artificial Bacterial Flagella (ABF) that are of a similar size and shape as natural bacterial flagella, and that swim using a similar low Reynolds number helical swimming strategy. ABF are made from a thin-film self-scrolling process. In this paper I describe why we want to do this, how each microrobot works, as well as the benefits of each strategy.

1 INTRODUCTION

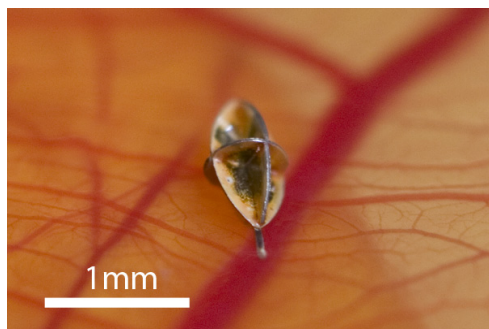
Micro and nanorobotics have the potential to dramatically change many aspects of medicine by navigating bodily fluids to perform targeted diagnosis and therapy and by manipulating cells and molecules. In the past few years, we have developed three new approaches to wirelessly controlling microscale structures with high precision over long distances in liquid environments (Figure 1). Because the distance from which these structures can be controlled is relatively large, the structures can not only be used as tools for manipulating other micro and nanoscale structures, similar to particle trapping techniques, but can also serve as vehicles for targeted delivery to locations deep within the human body. The microrobots we have developed are non-spherical. Therefore, both their position and orientation can be precisely controlled, removing another limitation of particle trapping. Unprecedented control in multiple degrees of freedom has been achieved with field strengths as low as 1 mT.

2 MEDICAL MICROROBOTS

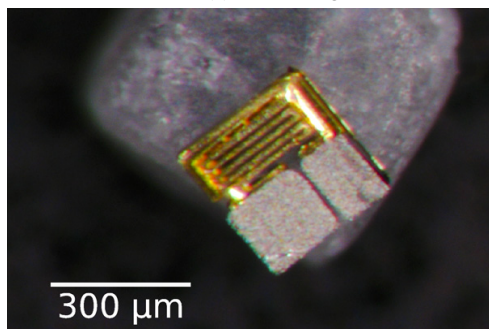
Minimally invasive medical techniques are linked with a variety of patient oriented benefits ranging

from reduction of recovery time, medical complications, infection risks, and post-operative pain, to lower hospitalization costs, shorter hospital stays, and increased quality of care [1-4]. Microrobotic devices have the potential for improved accessibility compared to current clinical tools, and medical tasks performed by them can even become practically noninvasive. They will perform tasks that are either difficult or impossible with current methods. Rather than acting as autonomous agents that navigate the body diagnosing and solving problems, microrobots will more likely act as new technical tools for clinicians, continuing to capitalize on the clinicians cognitive skill, which is their greatest asset.

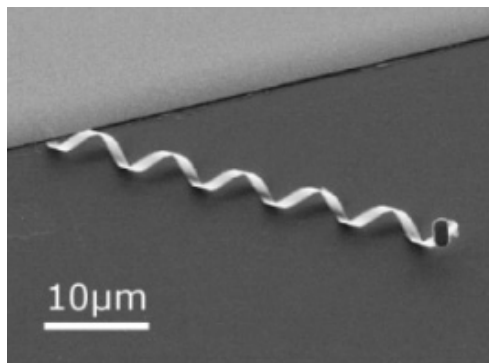
In recent years there has been significant progress on robot-assisted colonoscopy and on wireless miniature robots for use in the GI tract (Kazzim et al., 2006). Motivated by capsule endoscopes that are already in clinical use, a number of technologies have been explored to expand the capabilities of these devices, ranging from wireless GI pressure monitoring systems and lab-on-a-chip devices equipped with pH and temperature sensors (Johannessen et al., 2006) to the addition of legs and other mechanisms for controlled locomotion (Menciassi et al., 2007). The size of these devices approaches a few centimeters, capitalizing on the relatively large size of the GI tract. By further reducing device size and creating microrobots with a



(a) Octomag



(b) Magmite



(c) ABF

Figure 1: Three different types of microrobots at varying orders of magnitude. a) The Octomag robot (shown puncturing a small vein) is controlled using magnetic fields and field gradients. b) The Magmite (sitting on a grain of salt) is powered by oscillating magnetic fields that excite a spring mass system to resonance that harvests the impact energy. c) Artificial Bacterial Flagella (ABF) are propelled through fluid by rotating a magnetic field to generate torques on the magnetic metal head of the device.

maximum dimension of only a few millimeters or less, additional locations in the human body become available for wireless intervention. Natural pathways such as the circulatory system, the urinary system, and the central nervous system become available, enabling intervention with minimal trauma.

As we downscale robots to submillimeter

dimensions, the relative importance of physical effects changes (Wautelet, 2001). As device size is reduced, surface effects and fluid viscosity dominate over inertia and other volumetric effects, and power storage becomes a key issue. Furthermore, microrobots, like microorganisms, swim in a low-Reynolds-number regime, requiring swimming methods that differ from macroscale swimmers (Purcell, 1977). This places strong constraints on the development of medical microrobots. In traditional robotics, it is often easy to compartmentalize aspects of robot design such as kinematics, power, and control. In the design of wireless microrobots, fabrication is fundamentally limited by scaling issues, and power and control are often inextricably linked. Engineers must give up intuition gained from observing and designing in the macroscale physical world, and instead rely on analysis and simulation to explore microrobot design. Even then, only experimental results will demonstrate the efficacy of a given microrobot strategy, as the world experienced by the microrobot may be quite difficult to accurately model.

3 OCTOMAG MICROROBOTS

The Octomag microrobot (Yesin et al., 2006), shown in Figure 1, was the first microrobot we developed and is primarily intended for ophthalmic surgery. An external magnetic field acts to align the robot along the long (“easy”) axis, and a field gradient is generated to pull or push the microrobot. The winged shape acts to reduce the side-ways drift of the microrobot by increasing the fluid drag along the axes perpendicular to the long axis (Abbott et al., 2007). The relatively large size of the device is compatible with using gradient fields for propulsion at distances suitable for use within the body and is targeted at controlling the device in a viscous medium.

The robot is a three-dimensional structure built by microassembling individual parts, which allows for the combination of incompatible materials and processes for the integration of MEMS based sensors and actuators. The principle advantage of the hybrid design is that the individual parts of the assembly can be produced with standard MEMS manufacturing processes that create planar geometries. In this way, different subsystems of the robot can be manufactured using the most suitable process for the purpose. Robot parts have been made with electroplated nickel, single crystal silicon, polymer, and laser cut steel.

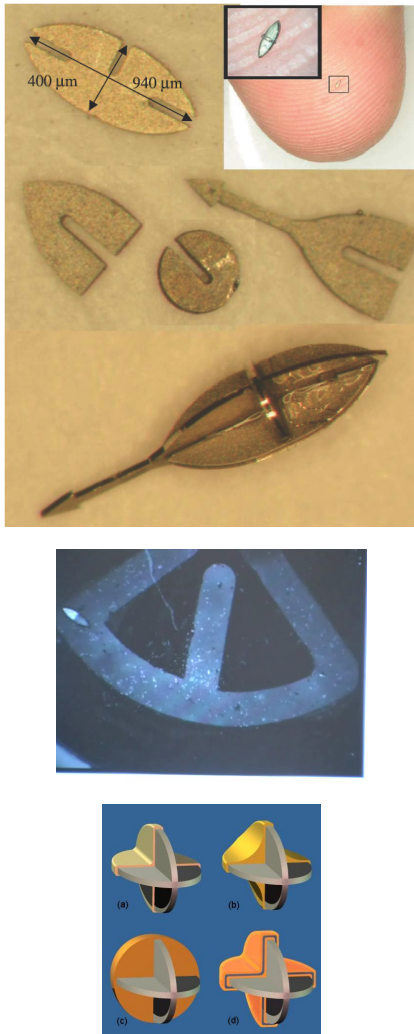


Figure 2: (a) Octomag microrobots are microassembled from 50μm thick electroplated nickel. (b) The sub-mm devices can be precisely controlled to navigate through tiny mazes, and (c) can be coated to ensure biocompatibility and for transport of cargo, such as drugs to be unloaded by diffusion (Yesin et al., 2006).

4 MAGMITES

With decreasing size, gradient propulsion becomes infeasible due to the force generated being related to the magnetic medium's volume, which decreases rapidly with size. To overcome this limitation, we have developed a second propulsion mechanism that harnesses the interactive forces between small magnetic bodies in a uniform magnetic field to drive a spring mechanism to resonance (Vollmers et al., 2008). This energy is then rectified to move the robot through its environment.

The resonant nature of the actuator enables the device to move with fields below 2 mT which is roughly 50x that of the Earth's magnetic field. This locomotion mechanism has been demonstrated on both structured and unstructured surfaces and is controllable enough to repeatedly and precisely follow trajectories. The frequency selectivity of the spring mass resonating structure allows multiple robotic agents to be used on the same substrate to perform tasks. Although this propulsion method was initially designed for operation in air, it has demonstrated its ability to perform in aqueous environments and manipulate glass microspheres on the order of 50 μm.

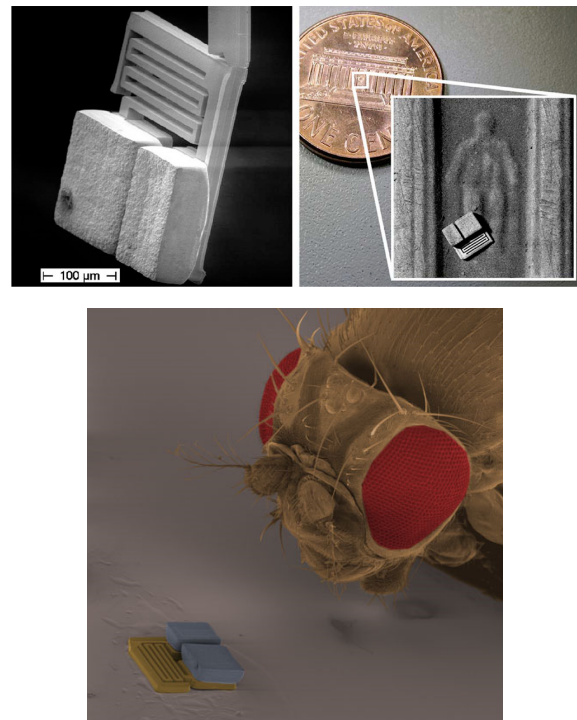


Figure 3: Magmite robots consist of two Ni masses separated by a gold spring. The robot shown measures 300μm square, 70μm thick, and is dwarfed by *Drosophila melanogaster* (Vollmers et al., 2008).

5 ARTIFICIAL BACTERIAL FLAGELLA

As sizes decrease further, the resonant frequencies of the mechanical structures required for the resonant magnetic actuator increase to tens of kHz and become difficult to generate at sufficient strength. At this scale, torque on the magnetic bodies becomes one of the predominant forces that can be generated.

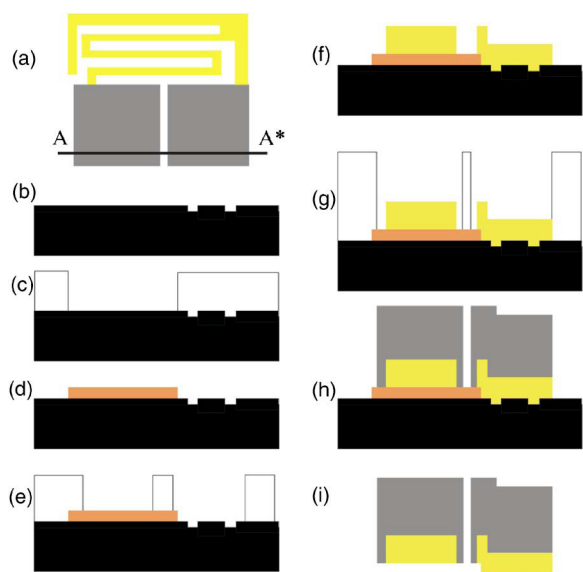


Figure 4: Fabrication sequence with cross section shown along line A-A* (a). Holes for dimples (b) are etched in a wafer before a Ti/Cu adhesion/seed layer is evaporated onto the surface. Photoresist is applied (c) to define thick electroplated copper islands (d). The springs and frame are defined (e) and plated (f) before a final layer of photoresist (g) defines the nickel bodies (h). The device is released from the wafer by etching the sacrificial copper layer (i).

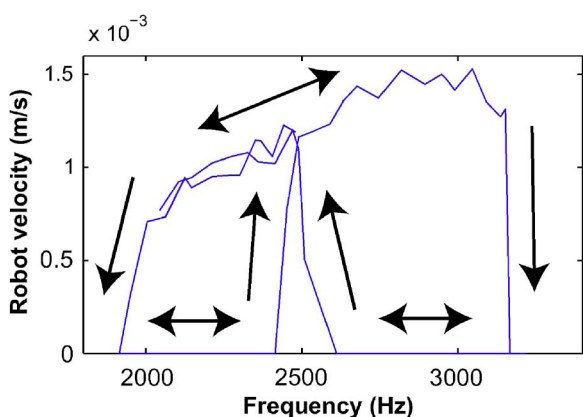


Figure 5: Robot velocity as a function of frequency with a driving field of 2.2 mT. When moving, the robot never comes to a complete rest to allow static friction can take effect. Driving with a frequency too far from resonance reduces the system energy and at some point the robot sticks to the substrate. Moving back toward resonance increases the absorbed energy, allowing the robot to begin moving again.

Taking inspiration from nature, this torque can be leveraged to create artificial bacterial flagella (Zhang et al., 2009).

The helical swimming robot consists of two parts: a helical tail and a magnetic metal head. The tails are

27 to 42 nm thick, less than 2 μm wide, and coil into diameters smaller than 3 μm . The robots are fabricated by a self-scrolling technique (Zhang et al., 2006). The helical swimming microbots are propelled and steered precisely in water by a rotating magnetic field on the order of 1 to 2 mT. As the robot's principle dimensions approach those of individual cells, many of the experimental methods used with the robots parallel those used by their biological counterparts.

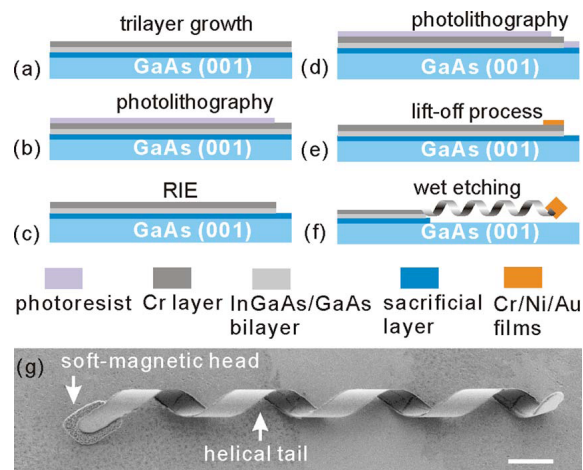


Figure 6: [(a)-(f)] Fabrication procedure of the ABF with InGaAs/GaAs/Cr helical tail. (g) FESEM image of an untethered ABF. The scale bar is 4 μm (Zhang et al., 2009).

6 SUMMARY

Recent advances in microbotics have demonstrated new capabilities in wirelessly controlling microscale structures with high precision over long distances in liquid environments. These breakthroughs make it possible to experimentally investigate the use of these microrobots for manipulating micro and nano size structures in as many as six degrees-of-freedom. Applications to nanomedicine in areas related to targeted medical therapies and molecular manipulation are clear, though many challenges must be addressed. To functionalize these devices and to improve their performance capabilities, fundamental issues in the role surface forces play must be addressed; biocompatibility must be ensured; loading and diffusion of biomolecules must be investigated; and interactions with and manipulation of tissue and macromolecules must be considered. There is a lot yet to do.

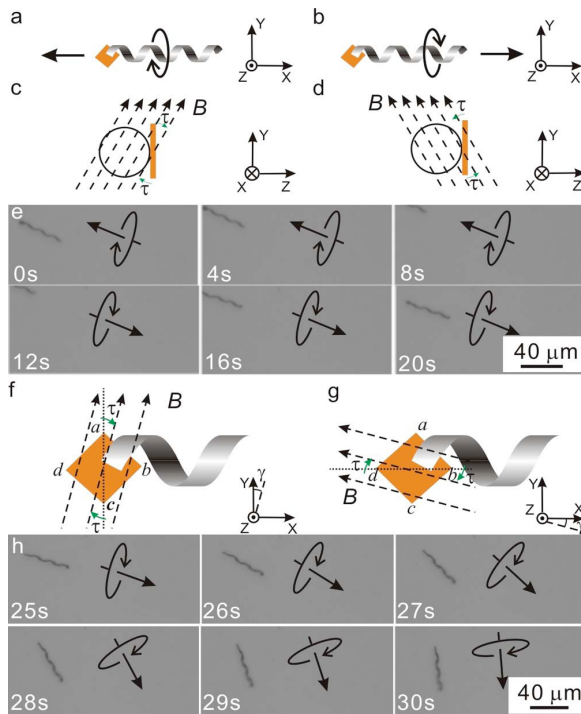


Figure 7: ABF swimming motion controlled by magnetic fields with field strength of 2.0 mT. [(a)-(d)] Schematic of a left-handed ABF swimming forward and backward. With the field B continuously rotating perpendicular to the X axis of the ABF, a misalignment angle between the field and the thin magnetic head will induce a magnetic torque (τ) that attempts to align the ABF head with the field, resulting in rotation and propulsion of the ABF. (e) Optical microscope images of the forward/backward motion of an ABF controlled by magnetic fields. The commanded translation and rotation directions of the ABF are indicated by the arrows. (f) If the field is rotated about the Z axis by an angle $|\gamma| < 45^\circ$ with respect to the easy axis ac of the head, then the ABF is steered as it is propelled, as the easy axis ac attempts to align with the field. This is the steering principle used during normal operation of the ABF. (g) If the field is rotated about the Z axis by an angle $|\gamma| < 45^\circ$ with respect to the easy axis bd , the ABF will instantaneously attempt to rotate perpendicular to the helix axis. However, steering using the bd easy axis is not possible simultaneously with forward/backward propulsion. (h) Optical microscope images of the turning motion of an ABF controlled by magnetic fields. The commanded translation and rotation directions of the ABF are indicated by the arrows.

REFERENCES

F. Tendick, S. S. Sastry, R. S. Fearing, and M. Cohn., "Applications of micromechatronics in minimally invasive surgery," *IEEE/ASME Transactions on Mechatronics*, vol. 3, pp. 34-42, 1998.

P. Dario, M. C. Carrozza, A. Benvenuto, and A. Menciasci, "Micro-systems in biomedical applications," *Journal of Micromechanics and Microengineering*, vol. 10, pp. 235-244, 2000.

M. J. Mack, "Minimally invasive and robotic surgery," *Journal of American Medical Association*, vol. 285, pp. 568-572, 2001.

M. C. Carrozza, P. Dario, and L. P. S. Jay, "Micromechatronics in surgery," *Transactions of the Institute of Measurement and Control*, vol. 25, pp. 309-327, 2003.

I. Kassim, L. Phee, W. S. Ng, F. Gong, P. Dario, and C. A. Mosse, "Locomotion techniques for robotic colonoscopy," *IEEE Engineering in Medicine and Biology Magazine*, pp. 49-56, May/June 2006.

E. A. Johannessen, L. Wang, S. W. J. Reid, D. R. S. Cumming, and J. M. and Cooper, "Implementation of radiotelemetry in a lab-in-a-pill format," *Lab on a Chip*, vol. 6, pp. 39-45, 2006.

A. Menciasci, D. Accoto, S. Gorini, and P. Dario, "Development of a biomimetic miniature robotic crawler," *Autonomous Robots*, vol. 21, pp. 155-163, 2007.

M. Wautelet, "Scaling laws in the macro-, micro- and nanoworlds," *European Journal of Physics*, vol. 22, pp. 601-611, Nov 2001.

E. M. Purcell, "Life at low Reynolds numbers," *American Journal of Physics*, vol. 45, pp. 3-11, 1977.

K. B. Yesin, K. Vollmers and B.J. Nelson, "Modeling and control of untethered biomicrobots in a fluidic environment using electromagnetic fields," *International Journal of Robotics Research*, vol. 25, pp. 527-536, 2006.

J. J. Abbott, O. Ergeneman, M. Kummer, A. M. Hirt, and B. J. Nelson, "Modeling magnetic torque and force for controlled manipulation of soft-magnetic bodies," *IEEE Transactions on Robotics*, vol. 23, pp. 1247-1252, December 2007.

K. Vollmers, D. Frutiger, B. E. Kratochvil, and B. J. Nelson, "Wireless resonant magnetic actuation for untethered microrobots," *Applied Physics Letters*, vol. 92, April 2008.

L. Zhang, J.J. Abbott, L.X. Dong, B.E. Kratochvil, D.J. Bell, D.J. and B.J. Nelson, "Artificial bacterial flagella: Fabrication and magnetic control," *Applied Physics Letters*, vol. 94, February 2009.

L. Zhang, L. X. Dong, D. J. Bell, B. J. Nelson, C. Schoenenberger, and D. Gruetzmacher, "Fabrication and characterization of freestanding Si/Cr micro- and nanospirals," *Microelectron. Eng.*, vol. 83, pp. 1237-1240, Apr-Sep 2006.

BRIEF BIOGRAPHY

Brad Nelson is the Professor of Robotics and Intelligent Systems at ETH Zürich. His primary research focus is on microrobotics and nanorobotics with an emphasis on applications in biology and

medicine. He received a B.S.M.E. from the University of Illinois at Urbana-Champaign and an M.S.M.E. from the University of Minnesota. He has worked as an engineer at Honeywell and Motorola and served as a United States Peace Corps Volunteer in Botswana, Africa, before obtaining a Ph.D. in Robotics from Carnegie Mellon University in 1995. He was an Assistant Professor at the University of Illinois at Chicago (1995-1998) and an Associate Professor at the University of Minnesota (1998-2002). He became a Full Professor at ETH Zürich in 2002.

Prof. Nelson has been awarded a McKnight Land-Grant Professorship and is a recipient of the Office of Naval Research Young Investigator Award, the National Science Foundation Faculty Early Career Development (CAREER) Award, the McKnight Presidential Fellows Award, and the Bronze Tablet. He was elected as a Robotics and Automation Society Distinguished Lecturer in 2003 and 2008 and won Best Paper Awards at major robotics conferences and journals in 2004, 2005, 2006, 2007, 2008 and 2009. He was named to the 2005 "Scientific American 50," Scientific American magazine's annual list recognizing fifty outstanding acts of leadership in science and technology from the past year for his efforts in nanotube manufacturing. His laboratory won the 2007 and 2009 RoboCup Nanogram Competition, both times the event has been held. He serves on the editorial boards of several journals, has served as the head of the Department of Mechanical and Process Engineering from 2005–2007, and is currently the Chairman of the ETH Electron Microscopy Center (EMEZ).

DYNAMIC MODELING OF ROBOTS USING RECURSIVE NEWTON-EULER TECHNIQUES

Wisama Khalil

Ecole Centrale de Nantes, IRCCyN UMR CNRS 6597, 1 Rue de la Noë, 44321 Nantes, France

Wisama.khalil@irccyn.ec-nantes.fr

Keywords: Dynamic Modelling, Newton-Euler, Recursive Calculation, Tree Structure, Parallel Robots, Flexible Joints, Mobile Robots.

Abstract: This paper present the use of recursive Newton-Euler to model different robotics systems. The main advantages of this technique are the facility of implementation by numerical or symbolical programming and providing models with reduced number of operations. In this paper the inverse and direct dynamic models of different robotics systems will be presented. At first we start by rigid tree structure robots, then these algorithms will be generalized for closed loop robots, parallel robots, and robots with lumped elasticity. At the end the case of robots with moving base will be treated.

1 INTRODUCTION

The dynamic modelling of robots is an important topic for the design, simulation, and control of robots. Different techniques have been proposed and used by the robotics community. In this paper we show that the use of Newton-Euler recursive technique for different robotics systems is easy to develop and programme. The proposed algorithm can be extended to many types of structures; serial, tree structure, closed, parallel, with a fixed base or with moving platform. The same technique can be used for robots with lumped elasticity or flexible links.

In section 2 we will recall the method used to describe the kinematics of the structure, and then in section 3 we present the inverse and the direct dynamic modeling of tree structure rigid robots which are considered as the base methods. The following sections present the generalization to the other systems.

2 DESCRIPTION OF THE KINEMATICS OF ROBOTS

The geometry of the structures will be described using the Modified Denavit and Hartenberg method as proposed in (Khalil and Kleinfinger, 1986). This method can take into account tree structures and closed loop robots. Its use facilitates the calculation

of the base inertial parameters of robots (Gautier and Khalil, 1988, Khalil W., Bennis F., 1994, Khalil and Bennis, 1995).

2.1 Geometric Description of Tree Structure Robots

A tree structure robot is composed of $n+1$ links and n joints. Link 0 is the base and link n is a terminal link. The joints are either revolutes or prismatic, rigid or elastic. The links are numbered consecutively from the base, link 0, to the terminal links. Joint j connects link j to link $a(j)$, where $a(j)$ denotes the link antecedent to link j . A frame R_i is attached to each link i such that (Figure 1):

- z_i is along the axis of joint i ;
- x_i is taken along the common normal between z_i and one of the succeeding joint axes, which are fixed on link i .

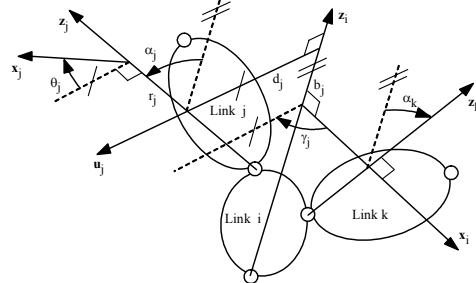


Figure 1: Geometric parameters for a link i .

In general the homogeneous transformation matrix ${}^i\mathbf{T}_j$, which defines the frame R_j relative to frame R_i is obtained as a function of six geometric parameters $(\gamma_j, b_j, \alpha_j, d_j, \theta_j, r_j)$. Thus ${}^i\mathbf{T}_j$ is obtained as:

$${}^i\mathbf{T}_j = \mathbf{Rot}(z, \gamma_j) \mathbf{Tran}(z, b_j) \mathbf{Rot}(x, \alpha_j) \mathbf{Tran}(x, d_j) \mathbf{Rot}(z, \theta_j) \mathbf{Tran}(z, r_j)$$

After developing, this matrix can be partitioned as follows:

$${}^i\mathbf{T}_j = \begin{bmatrix} {}^i\mathbf{R}_j & {}^i\mathbf{P}_j \\ \mathbf{0}_{1 \times 3} & 0 \end{bmatrix} \quad (1)$$

Where \mathbf{R} defines the (3×3) rotation matrix and \mathbf{P} defines the (3×1) vector defining the position of the origin of frame j with respect to frame i .

If \mathbf{x}_i is along the common normal between \mathbf{z}_i and \mathbf{z}_j , the parameters γ_j and b_j will be equal to zero.

The joint variable of joint j is denoted by:

$$q_j = \bar{\sigma}_j \theta_j + \sigma_j r_j$$

where $\sigma_j = 0$ if joint j is revolute, $\sigma_j = 1$ if joint j is prismatic, and $\bar{\sigma}_j = 1 - \sigma_j$. We set $\sigma_j = 2$ to define a frame R_j fixed with respect to frame $a(j)$. In this case, q_j and $\bar{\sigma}_j$ are not defined.

The serial structure is a special case of a tree structure where $a(j)=j-1$, $\gamma_j=0$, and $b_j=0$ for all $j=1, \dots, n$.

2.2 Description of Closed Loop Structure

The system is composed of L joints and $n+1$ links, where link 0 is the fixed base and $L > n$. The number of independent closed loops is equal to:

$$B = L - n$$

The joints are either active (motorized) or passive. The number of active joints is denoted N . The position and orientation of all the links can be determined as a function of the active joint variables.

To determine the geometric parameters of a mechanism with closed chains, we proceed as follows:

a) Construct an equivalent tree structure having n joints by virtually cutting each closed chain at one of its passive joints. Define the geometric parameters of the tree structure as given in section 2.1.

b) For each cut joint define two supplementary frames on one of the links connected by this joint. Assuming that a cut joint is numbered k (where $k=n+1, \dots, L$) and that the links connected by joint k

are numbered i and j (where i and $j < n$) the frames will be defined as follows (Figure 2):

- frame R_k is defined fixed on link j such that $a(k)=i$, the axis \mathbf{z}_k is along the axis of joint k , and \mathbf{x}_k is along the common normal between \mathbf{z}_k and \mathbf{z}_j . The matrix ${}^i\mathbf{T}_k$ will be determined using the general parameters $\gamma_k, b_k, \alpha_k, d_k, \theta_k, r_k$.

- frame R_{k+B} is aligned with R_k that is to say it is fixed on link j , but $a(k+B)=j$. The geometric parameters defining R_{k+B} are constant, we note that r_{k+B} and θ_{k+B} are zero since \mathbf{x}_{k+B} is normal to \mathbf{z}_j .

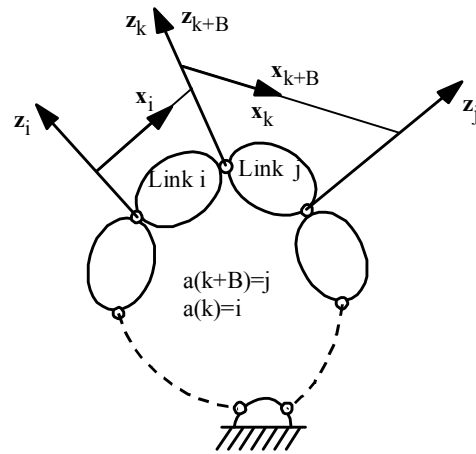


Figure 2: Frames of a cut joint k .

The joint variables are denoted as:

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_{tr} \\ \mathbf{q}_c \end{bmatrix}, \mathbf{q}_{tr} = \begin{bmatrix} \mathbf{q}_a \\ \mathbf{q}_p \end{bmatrix} \quad (2)$$

- \mathbf{q}_{tr} vector containing the tree structure joint variables;
- \mathbf{q}_a vector containing the N active joint variables;
- \mathbf{q}_p vector containing the $p=n-N$ passive joint variables of the equivalent tree structure;
- \mathbf{q}_c vector containing the B variables of the cut joints.

Only the N active variables \mathbf{q}_a are independent. Since R_k and R_{k+B} are aligned, the geometric constraint equations for each loop, which can be used to calculate the passive joint variables in terms of the active joint variables, can be written as:

$${}^{k+B}\mathbf{T}_j \dots {}^i\mathbf{T}_k = \mathbf{I}_4 \quad (3)$$

The kinematic constraint equations are obtained by using the fact that the screw of frame k is equal to that of frame $k+B$:

$$\begin{aligned} {}^0\mathbb{V}_k &= {}^0\mathbb{V}_{k+B} \\ \mathbf{J}_k \dot{\mathbf{q}}_{b1} &= \mathbf{J}_{k+B} \dot{\mathbf{q}}_{b2} \end{aligned} \quad (4)$$

\mathbb{V}_j (6×1) kinematic screw vector of frame j , given by:

$$\mathbb{V}_j = \begin{bmatrix} \mathbf{V}_j^T & \boldsymbol{\omega}_j^T \end{bmatrix}^T \quad (5)$$

\mathbf{V}_j linear velocity of the origin of frame R_j ,

$\boldsymbol{\omega}_j$ angular velocity of frame j ,

$\dot{\mathbf{q}}_{b1}$ joint velocities from the base to frame k , through branch 1,

$\dot{\mathbf{q}}_{b2}$ joint velocities from the base to frame k through branch 2.

3 DYNAMIC MODELING OF TREE STRUCTURE ROBOTS

3.1 Introduction

The most common in use methods to calculate the dynamic models are the Lagrange equations and the Newton Euler Equations (Craig 1986, Khalil and Dombre 2002, Angeles 2006).

The Lagrange equation is given as:

$$\boldsymbol{\Gamma} = \frac{d}{dt} \left[\frac{\partial L}{\partial \dot{\mathbf{q}}} \right]^T - \left[\frac{\partial L}{\partial \mathbf{q}} \right]^T \quad (6)$$

where $\boldsymbol{\Gamma}$ is the joint torques and forces, L is the Lagrangian of the robot defined as the difference between the kinetic energy E and the potential energy U of the system: $L = E - U$. After developing we obtain:

$$\boldsymbol{\Gamma} = \mathbf{A}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \quad (7)$$

where \mathbf{A} is the inertia matrix of the robot and \mathbf{H} is the Coriolis, Centrifuge and gravity torques.

Solving the previous equation to find $\boldsymbol{\Gamma}$ in terms of $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is known as the inverse dynamic problem, and solving it to obtain $\ddot{\mathbf{q}}$ in terms of $(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\Gamma})$ is known as the direct dynamic model. The inverse dynamic model is obtained by substituting $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ into (7), whereas the direct model needs to inverse the inertia matrix.

$$\ddot{\mathbf{q}} = -(\mathbf{A})^{-1} (\boldsymbol{\Gamma} - \mathbf{H}) \quad (8)$$

The calculation of the Lagrange equations for systems with big number of degrees of freedom using developed symbolic methods is time consuming, and the obtained model will need more

time to execute with respect to that of recursive methods. The recursive Newton-Euler algorithms have been shown to be an excellent tool to model rigid robots (Khalil and Kleinfinger, 1987, Khalil and Creusot, 1987, Khosla, 1987). In (Hollerbach, 1980) an efficient recursive Lagrange algorithm is presented but without achieving better performances than that of Newton-Euler.

The Newton-Euler equations giving the external forces and moments on a link j about the origin of frame j are written as:

$${}^j\mathbb{F}_j = {}^j\mathbb{J}_j {}^j\dot{\mathbb{V}}_j + \begin{bmatrix} {}^j\boldsymbol{\omega}_j \times ({}^j\boldsymbol{\omega}_j \times {}^j\mathbf{M}\mathbf{S}_j) \\ {}^j\boldsymbol{\omega}_j \times ({}^j\mathbf{J}_j {}^j\boldsymbol{\omega}_j) \end{bmatrix} \quad (9)$$

where

$${}^j\mathbb{F}_j = \begin{bmatrix} {}^j\mathbf{F}_j \\ {}^j\mathbf{M}_j \end{bmatrix} \quad (10)$$

$\boldsymbol{\omega}_j$ the angular velocity of link j ;

$\dot{\mathbb{V}}_j$ the linear acceleration of the origin of frame j ;

\mathbb{F}_j total external wrench on link j ;

\mathbf{F}_j total external forces on link j ;

\mathbf{M}_j total external moments on link j about O_j ;

\mathbb{J}_j (6×6) inertia matrix of link j :

$${}^j\mathbb{J}_j = \begin{bmatrix} M_j \mathbf{I}_3 & -{}^j\mathbf{M}\mathbf{S}_j \\ {}^j\mathbf{M}\hat{\mathbf{S}}_j & {}^j\mathbf{J}_j \end{bmatrix} \quad (11)$$

Where M_j , $\mathbf{M}\mathbf{S}_j$ and \mathbf{J}_j are the standard inertial parameters of link j . They are respectively, the mass, the first moments, and the inertia matrix about the origin.

3.2 Calculation of the Inverse Dynamics using Recursive NE Algorithm

The algorithm consists of two recursive computations (Luh, Walker and Paul, 1980): forward recursion and backward recursion. The forward equations, from link 1 to link n , compute the link velocities and accelerations and consequently the dynamic wrench on each link. The backward equations, from link n to the base, provide the reaction wrenches on the links and consequently the joint torques.

This method gives the joint torques in terms of the joint positions, velocities and accelerations without explicitly computing the matrices \mathbf{A} and \mathbf{H} . That is to say the algorithm will be denoted by:

$$\boldsymbol{\Gamma} = \text{NE}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathbf{f}_c, \mathbf{m}_c) \quad (12)$$

Where \mathbf{f}_e and \mathbf{m}_e are the external forces and moments of the links of the robot on the environment.

The forward recursive equations are based on the following equations (Khalil and Dombre 2002):

$${}^j\mathbf{V}_j = {}^j\mathbb{T}_i {}^i\mathbf{V}_i + \dot{\mathbf{q}}_j {}^j\mathbf{a}_j \quad (13)$$

$${}^j\dot{\mathbf{V}}_j = {}^j\mathbb{T}_i {}^i\dot{\mathbf{V}}_i + {}^j\boldsymbol{\gamma}_j + \dot{\mathbf{q}}_j {}^j\mathbf{a}_j \quad (14)$$

$${}^j\boldsymbol{\gamma}_j = \begin{bmatrix} {}^j\mathbf{R}_i \left[{}^i\boldsymbol{\omega}_i \times ({}^i\boldsymbol{\omega}_i \times {}^i\mathbf{P}_j) \right] + 2\sigma_j ({}^i\boldsymbol{\omega}_i \times \dot{\mathbf{q}}_j {}^j\mathbf{a}_j) \\ \bar{\sigma}_j {}^j\boldsymbol{\omega}_i \times \dot{\mathbf{q}}_j {}^j\mathbf{a}_j \end{bmatrix} \quad (15)$$

where

${}^j\mathbb{A}_j$ is the (6x1) column matrix given as :

$${}^j\mathbb{A}_j = [0 \ 0 \ \sigma_j \ 0 \ 0 \ \bar{\sigma}_j]^T \quad (16)$$

${}^j\mathbb{T}_i$ and the screw transformation matrix is:

$${}^j\mathbb{T}_i = \begin{bmatrix} {}^j\mathbf{R}_i & -{}^j\mathbf{R}_i {}^i\hat{\mathbf{P}}_j \\ \mathbf{0}_{3 \times 3} & {}^j\mathbf{R}_i \end{bmatrix} \quad (17)$$

The forward algorithm is given for $j=1, \dots, n$, with $i = a(j)$, as follows:

$${}^j\boldsymbol{\omega}_i = {}^j\mathbf{R}_i {}^i\boldsymbol{\omega}_i \quad (18)$$

$${}^j\boldsymbol{\omega}_j = {}^j\boldsymbol{\omega}_i + \bar{\sigma}_j \dot{\mathbf{q}}_j {}^j\mathbf{a}_j \quad (19)$$

$${}^j\dot{\boldsymbol{\omega}}_j = {}^j\mathbf{R}_i {}^i\dot{\boldsymbol{\omega}}_i + \bar{\sigma}_j (\dot{\mathbf{q}}_j {}^j\mathbf{a}_j + {}^j\boldsymbol{\omega}_i \times \dot{\mathbf{q}}_j {}^j\mathbf{a}_j) \quad (20)$$

$${}^j\dot{\mathbf{V}}_j = {}^j\mathbf{R}_i ({}^i\dot{\mathbf{V}}_i + {}^i\mathbf{U}_i {}^i\mathbf{P}_j) + \sigma_j (\ddot{\mathbf{q}}_j {}^j\mathbf{a}_j + 2{}^j\boldsymbol{\omega}_i \times \dot{\mathbf{q}}_j {}^j\mathbf{a}_j) \quad (21)$$

$${}^j\mathbf{F}_j = \mathbf{M}_j {}^j\dot{\mathbf{V}}_j + {}^j\mathbf{U}_j {}^j\mathbf{M}\mathbf{S}_j \quad (22)$$

$${}^j\mathbf{M}_j = {}^j\mathbf{J}_j {}^j\dot{\boldsymbol{\omega}}_j + {}^j\boldsymbol{\omega}_j \times ({}^j\mathbf{J}_j {}^j\boldsymbol{\omega}_j) + {}^j\mathbf{M}\mathbf{S}_j \times {}^j\dot{\mathbf{V}}_j \quad (23)$$

with

$${}^j\mathbf{U}_j = {}^j\hat{\boldsymbol{\omega}}_j + {}^j\hat{\boldsymbol{\omega}}_j {}^j\hat{\boldsymbol{\omega}}_j$$

and where \mathbf{a}_j is the unit vector along the \mathbf{z}_j axis which is the axis of joint j .

The matrix $\hat{\mathbf{W}}$ defines the 3×3 vector product matrix associated to the (3×1) vector \mathbf{W} such that:

$$\hat{\mathbf{W}} = \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix} \quad (24)$$

$$\mathbf{w} \times \mathbf{v} = \hat{\mathbf{W}} \mathbf{v}$$

These equations are initialized by $\boldsymbol{\omega}_0 = \mathbf{0}$, $\dot{\boldsymbol{\omega}}_0 = \mathbf{0}$, $\dot{\mathbf{V}}_0 = -\mathbf{g}$, ${}^0\mathbf{U}_0 = \mathbf{0}$, with \mathbf{g} is the acceleration of gravity.

Initialising the linear acceleration $\dot{\mathbf{V}}_0$ by $-\mathbf{g}$ will take

automatically the effect of gravity forces on all the links of the structure.

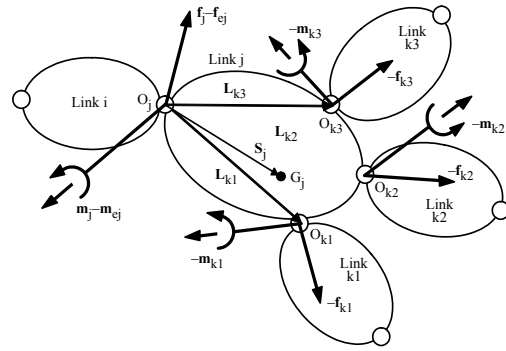


Figure 3: Forces and moments acting on a link j .

The backward recursive equations are deduced from the resultant forces and moments on link j around the origin of link j (Figure 3).

$${}^j\mathbf{f}_j = {}^j\mathbb{F}_j + \sum_k {}^k\mathbb{T}_j^T {}^k\mathbf{f}_k + {}^j\mathbf{f}_{e_j} \quad (25)$$

(25)

Where $a(k)=j$,

The backward equations can be calculated for $j=n, \dots, 1$:

$${}^j\mathbf{f}_j = {}^j\mathbf{F}_j + {}^j\mathbf{f}_{e_j} \quad (26)$$

$${}^j\mathbf{m}_j = {}^j\mathbf{M}_j + {}^j\mathbf{m}_{e_j} \quad (27)$$

$${}^i\mathbf{f}_j = {}^i\mathbf{R}_j {}^j\mathbf{f}_j \quad (28)$$

$${}^i\mathbf{f}_{e_i} = {}^i\mathbf{f}_{e_i} + {}^i\mathbf{f}_j \quad (29)$$

$${}^i\mathbf{m}_{e_i} = {}^i\mathbf{m}_{e_i} + {}^i\mathbf{R}_j {}^j\mathbf{m}_j + {}^i\mathbf{P}_j \times {}^i\mathbf{f}_j \quad (30)$$

$$\Gamma_j = (\sigma_j {}^j\mathbf{f}_j + \bar{\sigma}_j {}^j\mathbf{m}_j)^T {}^j\mathbf{a}_j + I_{a_j} \ddot{\mathbf{q}}_j + F_{s_j} \text{sign}(\dot{\mathbf{q}}_j) + F_{v_j} \dot{\mathbf{q}}_j \quad (31)$$

Where:

\mathbf{f}_j and \mathbf{m}_j are the reaction forces and moments of link $a(j)$ on link j respectively, I_{a_j} is the inertia of the rotor and transmission gears of the motor of joint j , F_{s_j} and F_{v_j} are the coulomb and viscous friction parameters respectively, ${}^j\mathbf{f}_{e_j}$ and ${}^j\mathbf{m}_{e_j}$ are the external forces and moments of link j on the environment.

This algorithm is easy to program numerically or symbolically. The computational cost is linear with the number of degrees of freedom of the robot. To reduce the number of operations of the calculation of this model the base inertial parameters can be used instead of the standard inertial parameters and the technique of customized symbolic method can be applied (Khosla 1986, Khalil and Kleinfinger 1987, Khalil and Creusot 1997).

3.3 Computation of the Direct Dynamic Model

The computation of the direct dynamic model is employed to carry out simulations for the purpose of testing the robot performances and studying the control laws. During simulation, the dynamic equations are solved for the joint accelerations given the input torques and the state of the robot (joint positions and velocities). Through integration of the joint accelerations, the robot trajectory is then determined.

The direct dynamic model can be obtained from Lagrange equation (8) as follows:

$$\ddot{\mathbf{q}} = \mathbf{A}^{-1} [\boldsymbol{\Gamma} - \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}})]$$

Two methods based on Newton-Euler methods can be used to obtain the dynamic model: the first is based on calculating the \mathbf{A} and \mathbf{H} matrices using Newton-Euler inverse dynamic model in order to calculate the joint accelerations by (8); the second method is based on a recursive Newton-Euler algorithm that does not explicitly calculate the matrix \mathbf{A} and has a computational cost that varies linearly with the number of degrees of freedom of the robot. For tree structure robots, the second method is more efficient, but the first method can be used for closed loop robots and other complicated systems. That is why we will present both methods.

3.3.1 Using the Inverse Dynamic Model to Calculate the Direct Dynamic Model

In this method the matrices $\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{A}(\mathbf{q})$ are calculated using the inverse model by giving special values for the joint accelerations, joint velocities, external forces, friction, gravity (Walker and Orin 1982).

By comparing equations (7) and (12) we deduce that $\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}})$ is equal to $\boldsymbol{\Gamma}$ if $\ddot{\mathbf{q}} = \mathbf{0}$, and that the i^{th} column of \mathbf{A} is equal to $\boldsymbol{\Gamma}$ if:

$$\ddot{\mathbf{q}} = \mathbf{u}_i, \dot{\mathbf{q}} = \mathbf{0}, \mathbf{g} = \mathbf{0}, \mathbf{f}_{ej} = \mathbf{0}, \mathbf{m}_{ej} = \mathbf{0}$$

where \mathbf{u}_i is the $(n \times 1)$ unit vector whose i^{th} element is equal to 1, and the other elements are zeros. Iterating the procedure for $i = 1, \dots, n$ leads to the construction of the entire inertia matrix.

To reduce the computational complexity of this algorithm, we can make use of the base inertial parameters and the customized symbolic techniques. Moreover, we can take advantage of the fact that the inertia matrix \mathbf{A} is symmetric.

3.3.2 Recursive NE Computation of the Direct Dynamic Model

This method is based on the recursive Newton-Euler equations and does not use explicitly the inertia matrix of the robot (Armstrong 1979, (Featherstone 1983, (Brandl, Johanni and Otter, 1986).

Using (9) and (25) the equilibrium equations of link j can be written as:

$${}^j\mathbb{J}_j {}^j\dot{\mathbf{V}}_j = {}^j\mathbf{f}_j + {}^j\boldsymbol{\beta}_j - \sum_k {}^k\mathbb{T}_j^T {}^k\mathbf{f}_k \quad (32)$$

where k denote the links articulated on link j such that $a(k)=j$, and

$${}^j\boldsymbol{\beta}_j = -{}^j\mathbf{f}_{ej} - \begin{bmatrix} {}^j\boldsymbol{\omega}_j \times ({}^j\boldsymbol{\omega}_j \times {}^j\mathbf{M}\mathbf{S}_j) \\ {}^j\boldsymbol{\omega}_j \times ({}^j\mathbf{J}_j {}^j\boldsymbol{\omega}_j) \end{bmatrix} \quad (33)$$

The joint accelerations are obtained as a result of three recursive computations:

i) first forward computations for $j = 1, \dots, n$: in this step, we compute the screw transformation matrices ${}^j\mathbb{T}_i$, the link angular velocities ${}^j\boldsymbol{\omega}_j$ as well as ${}^j\boldsymbol{\gamma}_j$ and ${}^j\boldsymbol{\beta}_j$ vectors, which appear in the link accelerations and the link wrenches equations respectively when $\ddot{\mathbf{q}} = \mathbf{0}$;

$${}^j\boldsymbol{\gamma}_j = \begin{bmatrix} {}^j\mathbf{R}_i \left[{}^i\boldsymbol{\omega}_i \times ({}^i\boldsymbol{\omega}_i \times {}^i\mathbf{P}_j) \right] + 2\sigma_j ({}^j\boldsymbol{\omega}_j \times \dot{q}_j {}^j\mathbf{a}_j) \\ \bar{\sigma}_j {}^j\boldsymbol{\omega}_j \times \dot{q}_j {}^j\mathbf{a}_j \end{bmatrix} \quad (34)$$

$${}^j\boldsymbol{\beta}_j = -{}^j\mathbf{f}_{ej} - \begin{bmatrix} {}^j\boldsymbol{\omega}_j \times ({}^j\boldsymbol{\omega}_j \times {}^j\mathbf{M}\mathbf{S}_j) \\ {}^j\boldsymbol{\omega}_j \times ({}^j\mathbf{J}_j {}^j\boldsymbol{\omega}_j) \end{bmatrix} \quad (35)$$

ii) backward recursive computation: in this step we calculate the elements $H_j, {}^j\mathbb{J}_j, {}^j\boldsymbol{\beta}_j, {}^j\mathbb{K}_j, {}^j\boldsymbol{\alpha}_j$ which express \ddot{q}_j and ${}^j\mathbf{f}_j$ in terms of ${}^i\dot{\mathbf{V}}_i$ in the third recursive equations. These equations are demonstrated in the following sub-section.

For $j = n \dots 1$, compute:

$$\mathbf{H}_j = ({}^j\mathbb{J}_j^T {}^j\mathbb{J}_j^* {}^j\mathbb{J}_j + \mathbf{I}a_j) \quad (36)$$

$${}^j\mathbb{K}_j = {}^j\mathbb{J}_j^* - {}^j\mathbb{J}_j^* {}^j\mathbb{J}_j {}^j\mathbb{J}_j^{-1} {}^j\mathbb{J}_j^T {}^j\mathbb{J}_j^* \quad (37)$$

$${}^j\boldsymbol{\alpha}_j = {}^j\mathbb{K}_j {}^j\boldsymbol{\gamma}_j + {}^j\mathbb{J}_j^* {}^j\mathbb{J}_j {}^j\mathbb{J}_j^{-1} (\tau_j + {}^j\mathbb{J}_j^T {}^j\boldsymbol{\beta}_j^*) - {}^j\boldsymbol{\beta}_j^* \quad (38)$$

If $a(j) \neq 0$, calculate also:

$${}^i\boldsymbol{\beta}_i^* = {}^i\boldsymbol{\beta}_i^* - {}^i\mathbb{T}_i^T {}^j\boldsymbol{\alpha}_j \quad (39)$$

$${}^i\mathbb{J}_i^* = {}^i\mathbb{J}_i^* + j\mathbb{T}_i^T j\mathbb{K}_j j\mathbb{T}_i \quad (40)$$

These equations are initialized by

$${}^j\mathbb{J}_j^* = j\mathbb{J}_j^* \text{ and } j\beta_j^* = j\beta_j.$$

iii) *second forward recursive computations.* Since the acceleration of the base is known ($\dot{\mathbf{V}}_0 = -\mathbf{g}$, $\dot{\boldsymbol{\omega}}_0 = \mathbf{0}$ for fixed base), the third recursive computation gives $\ddot{\mathbf{q}}_j$ and $j\mathbf{f}_j^*$ (if needed) for $j = 1 \dots n$. as follows:

$$\ddot{\mathbf{q}}_j = H_j^{-1} [-j\mathbb{a}_j^T j\mathbb{J}_j^* (j\mathbb{T}_i^T i\dot{\mathbf{V}}_i + j\boldsymbol{\gamma}_j) + \tau_j + j\mathbb{a}_j^T j\beta_j^*] \quad (41)$$

$$j\mathbf{f}_j^* = \begin{bmatrix} j\mathbf{f}_j \\ j\mathbf{m}_j \end{bmatrix} = j\mathbb{K}_j j\mathbb{T}_i^T i\dot{\mathbf{V}}_i + j\boldsymbol{\alpha}_j \quad (42)$$

$$j\dot{\mathbf{V}}_j = j\mathbb{T}_i^T i\dot{\mathbf{V}}_i + j\mathbb{a}_j \ddot{\mathbf{q}}_j + j\boldsymbol{\gamma}_j \quad (43)$$

where

$$\tau_j = \Gamma_j - F_{sj} \text{sign}(\dot{\mathbf{q}}_j) - F_{vj} \dot{\mathbf{q}}_j \quad (44)$$

Calculation of the elements of the backward recursive equations

To simplify the notations, we consider the case of a serial structure of n joints. Expressing the acceleration of link n in terms of the acceleration of link $n-1$, and since ${}^{n+1}\mathbf{f}_{n+1} = \mathbf{0}$, we obtain:

$${}^n\mathbb{J}_n ({}^n\mathbb{T}_{n-1}^T {}^{n-1}\dot{\mathbf{V}}_{n-1} + \ddot{\mathbf{q}}_n {}^n\mathbb{a}_n + {}^n\boldsymbol{\gamma}_n) = {}^n\mathbf{f}_n + {}^n\boldsymbol{\beta}_n \quad (45)$$

Since:

$$j\mathbb{a}_j^T j\mathbf{f}_j^* = \tau_j - I\mathbf{a}_j \ddot{\mathbf{q}}_j$$

$$\tau_j = \Gamma_j - F_{sj} \text{sign}(\dot{\mathbf{q}}_j) - F_{vj} \dot{\mathbf{q}}_j$$

We obtain the joint acceleration of joint n :

$$\ddot{\mathbf{q}}_n = H_n^{-1} (-{}^n\mathbb{a}_n^T {}^n\mathbb{J}_n ({}^n\mathbb{T}_{n-1}^T {}^{n-1}\dot{\mathbf{V}}_{n-1} + {}^n\boldsymbol{\gamma}_n) + \tau_n + {}^n\mathbb{a}_n^T {}^n\boldsymbol{\beta}_n) \quad (46)$$

where H_n is a scalar given as:

$$H_n = ({}^n\mathbb{a}_n^T {}^n\mathbb{J}_n {}^n\mathbb{a}_n + I\mathbf{a}_n) \quad (47)$$

Substituting for $\ddot{\mathbf{q}}_n$ from (46) and (45), we obtain the dynamic wrench ${}^n\mathbf{f}_n$ as:

$${}^n\mathbf{f}_n = \begin{bmatrix} {}^n\mathbf{f}_n \\ {}^n\mathbf{m}_n \end{bmatrix} = {}^n\mathbb{K}_n {}^n\mathbb{T}_{n-1}^T {}^{n-1}\dot{\mathbf{V}}_{n-1} + {}^n\boldsymbol{\alpha}_n \quad (48)$$

where:

$${}^n\mathbb{K}_n = {}^n\mathbb{J}_n - {}^n\mathbb{J}_n {}^n\mathbb{a}_n H_n^{-1} {}^n\mathbb{a}_n^T {}^n\mathbb{J}_n \quad (49)$$

$${}^n\boldsymbol{\alpha}_n = {}^n\mathbb{K}_n {}^n\boldsymbol{\gamma}_n + {}^n\mathbb{J}_n {}^n\mathbb{a}_n H_n^{-1} (\tau_n + {}^n\mathbb{a}_n^T {}^n\boldsymbol{\beta}_n) - {}^n\boldsymbol{\beta}_n \quad (50)$$

We now have $\ddot{\mathbf{q}}_n$ and ${}^n\mathbf{f}_n$ in terms of ${}^{n-1}\dot{\mathbf{V}}_{n-1}$. Iterating the procedure for $j = n-1$, we obtain:

$${}^{n-1}\mathbb{J}_{n-1} {}^{n-1}\dot{\mathbf{V}}_{n-1} = {}^{n-1}\mathbf{f}_{n-1} + {}^{n-1}\mathbb{T}_{n-1}^T {}^n\mathbf{f}_n + {}^{n-1}\boldsymbol{\beta}_{n-1} \quad (51)$$

which can be rewritten as:

$${}^{n-1}\mathbb{J}_{n-1}^* ({}^{n-1}\mathbb{T}_{n-2}^T {}^{n-2}\dot{\mathbf{V}}_{n-2} + \ddot{\mathbf{q}}_{n-1} {}^{n-1}\mathbb{a}_{n-1} + {}^{n-1}\boldsymbol{\gamma}_{n-1}) = {}^{n-1}\mathbf{f}_{n-1}^* + {}^{n-1}\boldsymbol{\beta}_{n-1}^* \quad (52)$$

where:

$${}^{n-1}\mathbb{J}_{n-1}^* = {}^{n-1}\mathbb{J}_{n-1} + {}^{n-1}\mathbb{T}_{n-1}^T {}^n\mathbb{K}_n {}^n\mathbb{T}_{n-1} \quad (53)$$

$${}^{n-1}\boldsymbol{\beta}_{n-1}^* = {}^{n-1}\boldsymbol{\beta}_{n-1} - {}^{n-1}\mathbb{T}_{n-1}^T {}^n\boldsymbol{\alpha}_n \quad (54)$$

Equation (52) has the same form as (45). Thus, we can express $\ddot{\mathbf{q}}_{n-1}$ and ${}^{n-1}\mathbf{f}_{n-1}$ in terms of ${}^{n-2}\dot{\mathbf{V}}_{n-2}$. Iterating this procedure for $j = n-2, \dots, 1$, we obtain $\ddot{\mathbf{q}}_j$ and $j\mathbf{f}_j^*$ in terms of ${}^{j-1}\dot{\mathbf{V}}_{j-1}$ for $j = n-1, \dots, 1$ as given by equations (41) and (43) which represent the general case.

4 INVERSE DYNAMIC MODELING OF CLOSED LOOP ROBOTS

The computation of the Inverse dynamic model of closed loop robots can be obtained by first calculating the inverse dynamic model of the equivalent tree structure robot, in which the joint variables satisfy the constraints of the loop. Then the closed loop torques of the active joints Γ_c are obtained by projecting the tree structure torques Γ_{tr} on the motorized joints using the transpose of the Jacobian matrix of the tree structure variables (or velocities) in terms of the active joint variables (or velocities).

$$\Gamma_c = \mathbf{G}^T \Gamma_{tr} (\mathbf{q}_{tr}, \dot{\mathbf{q}}_{tr}, \ddot{\mathbf{q}}_{tr}) \quad (55)$$

where:

$$\mathbf{G} = \frac{\partial \mathbf{q}_{tr}}{\partial \mathbf{q}_a} = \frac{\partial \dot{\mathbf{q}}_{tr}}{\partial \dot{\mathbf{q}}_a} \quad (56)$$

It can be written also as:

$$\Gamma_c = \Gamma_a + \frac{\partial \dot{\mathbf{q}}_p}{\partial \dot{\mathbf{q}}_a} \Gamma_p \quad (57)$$

Where:

Γ_a and Γ_p are the torque of actuated and passive joints of the tree structure.

The kinematics Jacobian matrix can be obtained from (4) representing the kinematics closed loop constraints.

There is no recursive method to obtain the direct dynamic model of closed loop robots. It can be

computed using the inverse dynamic model by a procedure similar to that given in section (3.3.1) in order to obtain the matrices \mathbf{A}_c and \mathbf{H}_c of the following relation:

$$\mathbf{\Gamma}_c = \mathbf{A}_c(\mathbf{q}_{tr})\ddot{\mathbf{q}}_a + \mathbf{H}_c(\mathbf{q}_{tr}, \dot{\mathbf{q}}_{tr}) \quad (58)$$

5 INVERSE DYNAMIC MODELING OF PARALLEL ROBOTS

A parallel robot is a complex multi-body system having several closed loops. It is composed of a moving platform connected to a fixed base by parallel legs. The dynamic model can be obtained as described in the previous section, but in this section we present a method that takes into account the parallel structure. To simplify the notations we will present her the case of parallel robots with six degrees of freedom. Examples concerning reduced mobility robots are given in (Khalil and Ibrahim 2007).

The robot is composed of a fixed base and a mobile platform. They are connected using m parallel legs.

The inverse dynamic model gives the forces and torques of motorized joints as a function of the desired trajectory of the mobile platform.

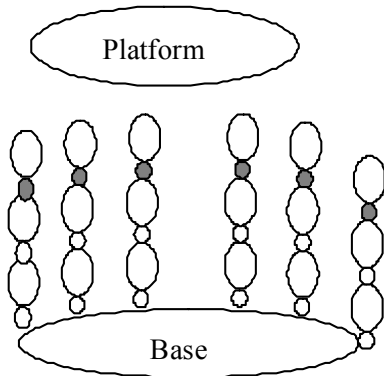


Figure 4: Parallel structure after separating the platform.

To obtain the dynamic models of parallel robots, we exploit their structural characteristics by decomposing the system into two subsystems: the platform and the legs.

The dynamics of the platform is calculated as a function of the Cartesian variables (spatial Cartesian position, velocity and acceleration of the platform), whereas the dynamics of the legs are calculated as a function of the joint variables of the legs

$(\mathbf{q}_i, \dot{\mathbf{q}}_i, \ddot{\mathbf{q}}_i)$ for $i=1, \dots, m$. The active joint torques are obtained by the sum of these dynamics and projecting them on the active joint axes.

To project the dynamics of the platform on the active joint space we multiply it by the transpose of the robot Jacobian matrix, which gives the platform screw \mathbb{V}_p in terms of the motorized joint velocities $\dot{\mathbf{q}}_a$, and to project the leg dynamics on the active joint space we use the Jacobian between these two spaces. Thus the dynamic model of the parallel structure is given by the following equation:

$$\mathbf{\Gamma} = \mathbf{J}_p^T \mathbb{F}_p + \sum_{i=1}^m \left(\frac{\partial \dot{\mathbf{q}}_i}{\partial \dot{\mathbf{q}}_a} \right)^T \mathbf{\Gamma}_i \quad (59)$$

where

\mathbb{F}_p is the total forces and moments on the platform,

\mathbf{J}_p is the $(6 \times n)$ kinematics Jacobian matrix of the robot, which gives the platform velocity \mathbf{V}_p (translational and angular) as a function of the active joint velocities:

$$\mathbf{V}_p = \mathbf{J}_p \dot{\mathbf{q}}_a \quad (60)$$

$\mathbf{\Gamma}_i$ is the inverse dynamic model of leg i , it is a function of $(\mathbf{q}_i, \dot{\mathbf{q}}_i, \ddot{\mathbf{q}}_i)$, which can be obtained in terms of the platform location, velocity and acceleration, using the inverse kinematic models of the legs. We note that \mathbf{q}_i does not include the passive joint variables connecting the legs to the platform.

In this section we suppose $n=6$, thus \mathbf{J}_p is (6×6) matrix.

The calculation of \mathbf{J}_p is obtained by inverting \mathbf{J}_p^{-1} , which is easy to obtain for most parallel structures.

\mathbb{F}_p is calculated by the Newton-Euler equation (9).

The calculation of $\partial \dot{\mathbf{q}}_i / \partial \dot{\mathbf{q}}_a$ is carried out by the following relation, which exploits the parallel structure of the robot:

$$\frac{\partial \dot{\mathbf{q}}_i}{\partial \dot{\mathbf{q}}_a} = \frac{\partial \dot{\mathbf{q}}_i}{\partial \mathbf{v}_i} \frac{\partial \mathbf{v}_i}{\partial \mathbf{V}_p} \frac{\partial \mathbf{V}_p}{\partial \dot{\mathbf{q}}_a} \quad (61)$$

with: \mathbf{v}_i is the Cartesian velocity transferred from leg i to the platform.

We can rewrite (61) as:

$$\frac{\partial \dot{\mathbf{q}}_i}{\partial \dot{\mathbf{q}}_a} = \mathbf{J}_i^{-1} \mathbf{J}_{vi} \mathbf{J}_r \quad (62)$$

\mathbf{J}_i is the kinematic Jacobian matrix of leg i such that:

$$\mathbf{v}_i = \mathbf{J}_i \dot{\mathbf{q}}_i \quad (63)$$

\mathbf{J}_{v_i} gives \mathbf{v}_i as a function of \mathbb{V}_p :

$$\mathbf{v}_i = \mathbf{J}_{v_i} \mathbb{V}_p \quad (64)$$

For the Gough-Stewart platform (where the mobile platform is connected to the legs using spherical joints), we obtain:

$$\mathbf{J}_{v_i} = \frac{\partial \mathbf{v}_i}{\partial \mathbb{V}_p} = \begin{bmatrix} \mathbf{I}_3 & \hat{\mathbf{P}}_i \end{bmatrix} \quad (65)$$

Where \mathbf{P}_i is the vector between the origin of the platform frame and the centre of the spherical joint linking the platform with leg i .

Finally the inverse dynamic model of the robot is given by the following form:

$$\mathbf{\Gamma} = \mathbf{J}_p^T \left[\mathbb{F}_p + \sum_{i=1}^m \mathbf{J}_{v_i}^T \mathbf{J}_i^T \mathbf{\Gamma}_i \right] \quad (66)$$

We note that the term between the brackets in (66) represents the dynamic model of the robot expressed in the Cartesian space of the platform frame (Khalil and Guegan, 2002).

6 INVERSE DYNAMIC MODELING OF ROBOTS WITH ELASTIC JOINTS

In this section we treat structure robots with lumped elasticity or flexible joints. The system can be described using Modified Denavit and Hartenberg method presented in section 2. Each joint could be either elastic or rigid (Khalil and Gautier, 2000).

6.1 Lagrange Dynamic Form

The general form of the dynamic model of a system with flexible joints has the same form as (7). It can be rewritten as:

$$\mathbf{\Gamma} = \mathbf{A}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \quad (67)$$

It can be partitioned as follows:

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{\Gamma}_r \\ \mathbf{\Gamma}_f \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{12}^T & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_r \\ \ddot{\mathbf{q}}_f \end{bmatrix} + \begin{bmatrix} \mathbf{H}_r \\ \mathbf{H}_f \end{bmatrix} \quad (68)$$

Where $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$ are the $(n \times 1)$ vectors of positions, velocities, and accelerations of rigid and elastic joints;

$\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}})$ is the $(n \times 1)$ vector of Coriolis, centrifugal and gravity forces,

$\mathbf{A}(\mathbf{q})$ is the $(n \times n)$ inertia matrix of the system,

$\mathbf{\Gamma}_r$ is the vector of rigid joint torques,

$\mathbf{\Gamma}_f$ is the vector of elastic joint torques.

If joint j is flexible:

$$\mathbf{\Gamma}_j = -\Delta q_j K_j \quad (69)$$

where K_j is the stiffness of the elastic joint,

$$\Delta q_j = q_j - q_{0j} \quad (70)$$

q_{0j} is the joint position corresponding to zero elasticity force.

In the case of a system with elasticity, the direct dynamic model has the same outputs as in the case of rigid bodies; it gives the joint accelerations as a function of the joint torques and of the system state variables $(\mathbf{q}, \dot{\mathbf{q}})$. It can be calculated using (68) by calculation the inverse of \mathbf{A} .

In the case of a system with elasticity, the inverse dynamic model calculates the input torques and the elastic accelerations as a function of the joint positions, velocities and rigid joint accelerations. It is to be noted that the accelerations of the elastic variables cannot be specified independently. Using (68) to calculate the inverse model, we have first to calculate the acceleration elastic accelerations from the second row:

$$\mathbf{\Gamma}_f = \begin{bmatrix} \mathbf{A}_{12}^T & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_r \\ \ddot{\mathbf{q}}_f \end{bmatrix} + \mathbf{H}_f \quad (71)$$

then we can calculate the rigid joint torques from the first row.

6.2 Direct Dynamics of Systems with Flexible Joints using Recursive NE

The direct dynamic model of system with flexible joints can be calculated using the recursive direct dynamic model algorithm of rigid joints presented in section (3.3) after putting $\mathbf{\Gamma}_j = -\Delta q_j K_j$ for the elastic joints.

Remark: We note that in case of rigid non motorized joint, the same algorithm can be used after putting $\mathbf{\Gamma}_j = 0$.

6.3 Inverse Dynamics of Systems with Flexible Joints using Recursive NE

The recursive inverse dynamic algorithm of rigid links cannot be used for system with flexible joints since the accelerations of the flexible joints are unknown. On the contrary it can be used to obtain the \mathbf{A} and \mathbf{H} matrices as explained in section (3.3.1),

then we can proceed as explained in 6.1 for the calculation of $\dot{\mathbf{q}}_j$ and Γ_j .

We propose here a recursive algorithm to solve this problem (Khalil and Gautier 2000). This algorithm consists of three recursive steps.

i) The first forward iteration is exactly the same as that of the direct dynamic model (section 3.3).

ii) The second backward recursive equations calculate the matrices giving the elastic accelerations $\ddot{\mathbf{q}}_j$ and $\dot{\mathbf{f}}_j$ as a function of ${}^a(j)\dot{\mathbf{V}}_a(j)$. These matrices can be defined using a similar procedure as in section (3.3). They can be calculated for $j=n, \dots, 1$, as follows:

- If joint j is elastic:

$$\mathbf{H}_j = {}^j\mathbf{a}_j^T {}^j\mathbb{J}_j^* {}^j\mathbf{a}_j \quad (72)$$

$${}^j\mathbb{K}_j = {}^j\mathbb{J}_j^* - {}^j\mathbb{J}_j^* {}^j\mathbf{a}_j \mathbf{H}_j^{-1} {}^j\mathbf{a}_j^T {}^j\mathbb{J}_j^* \quad (73)$$

$${}^j\boldsymbol{\alpha}_j = {}^j\mathbb{K}_j {}^j\boldsymbol{\gamma}_j + {}^j\mathbb{J}_j^* {}^j\mathbf{a}_j \mathbf{H}_j^{-1} (-\mathbf{K}_j \Delta \mathbf{q}_j + {}^j\mathbf{a}_j^T {}^j\boldsymbol{\beta}_j^*) - {}^j\boldsymbol{\beta}_j^* \quad (74)$$

- If joint j is rigid:

$${}^j\mathbb{K}_j = {}^j\mathbb{J}_j^* \quad (75)$$

$${}^j\boldsymbol{\alpha}_j = {}^j\mathbb{K}_j {}^j\boldsymbol{\gamma}_j + {}^j\mathbb{J}_j^* {}^j\mathbf{a}_j \ddot{\mathbf{q}}_j - {}^j\boldsymbol{\beta}_j^* \quad (76)$$

if $a(j) \neq 0$, calculate:

$${}^i\boldsymbol{\beta}_i^* = {}^i\boldsymbol{\beta}_i^* - {}^i\mathbb{T}_i^T {}^j\boldsymbol{\alpha}_j \quad (77)$$

$${}^i\mathbb{J}_i^* = {}^i\mathbb{J}_i^* + {}^i\mathbb{T}_i^T {}^j\mathbb{K}_j {}^i\mathbb{T}_i \quad (78)$$

The previous equations are initialized by:

$${}^j\mathbb{J}_j^* = {}^j\mathbb{J}_j, \text{ and } {}^j\boldsymbol{\beta}_j^* = {}^j\boldsymbol{\beta}_j.$$

The third recursive equations (for $j = 1, \dots, n$) calculate $\ddot{\mathbf{q}}_j$ for the elastic joints and the joint torques for the rigid joints using the following equation:

$$\dot{\mathbf{f}}_j = \begin{bmatrix} \dot{\mathbf{f}}_j \\ \dot{\mathbf{m}}_j \end{bmatrix} = {}^j\mathbb{K}_j {}^j\mathbb{T}_i^T {}^i\dot{\mathbf{V}}_i + {}^j\boldsymbol{\alpha}_j \quad (79)$$

- if j is elastic:

$$\ddot{\mathbf{q}}_j = \mathbf{H}_j^{-1} [-{}^j\mathbf{a}_j^T {}^j\mathbb{J}_j^* ({}^i\mathbb{T}_i^T {}^i\dot{\mathbf{V}}_i + {}^j\boldsymbol{\gamma}_j) - \mathbf{K}_j \Delta \mathbf{q}_j + {}^j\mathbf{a}_j^T {}^j\boldsymbol{\beta}_j^*] \quad (80)$$

$${}^j\dot{\mathbf{V}}_j = {}^j\mathbb{T}_i^T {}^i\dot{\mathbf{V}}_i + {}^j\mathbf{a}_j \ddot{\mathbf{q}}_j + {}^j\boldsymbol{\gamma}_j \quad (81)$$

- if j is rigid

$$\Gamma_j = (\sigma_j \dot{\mathbf{f}}_j + \bar{\sigma}_j \dot{\mathbf{m}}_j)^T {}^j\mathbf{a}_j + I_{a_j} \ddot{\mathbf{q}}_j \quad (82)$$

7 DYNAMIC MODELING OF ROBOTS WITH MOVING BASE

The structure treated in this section includes a big

number of systems such as: cars, mobile robots, mobile manipulators, walking robots, Humanoid robots, eel like robots (Khalil W., G. Gallot G., Boyer F., 2007), snakes like robots, flying robots, spatial vehicle, etc. The difference between all of these systems will be in the calculation of the interaction forces with the environment. In the previous sections the base is fixed thus the acceleration of the base is equal to zero, whereas in the case of a mobile base system the acceleration of the base must be determined in both direct and inverse dynamic models. The proposed recursive dynamic models are easy to implement and calculate using numerical calculation. The inverse dynamic model, which is used in general in the control problems, can be used in simulation too when the objective is to study the evolution of the base giving joint positions, velocities and accelerations of the other joints. The direct dynamic model can be used in simulation when the joint torques are specified.

We use the same notations of section 2 to describe the structure. The base fixed frame R_0 is defined wrt the world fixed frame R_w by the transformation matrix ${}^w\mathbf{T}_0$. This matrix is supposed known at $t = 0$, it will be updated by integrating the base acceleration. The velocity and acceleration of the base are represented by the (6×1) vectors \mathbf{V}_0 and $\dot{\mathbf{V}}_0$ respectively.

The Cartesian velocities and accelerations of the links are calculated using the recursive equations (13)-(17).

7.1 General form of the Dynamic Models

The dynamic model of a robot with moving base can be represented by the following relation:

$$\begin{bmatrix} \mathbf{0}_{6 \times 1} \\ \Gamma \end{bmatrix} = \mathbf{A} \begin{bmatrix} {}^0\dot{\mathbf{V}}_0 \\ \dot{\mathbf{q}} \end{bmatrix} + \mathbf{H} \quad (83)$$

Γ ($n \times 1$) vector of joint torques,

\mathbf{q} ($n \times 1$) vector of joint positions,

\mathbf{A} is the $(6+n) \times (6+n)$ inertia matrix of the robot, it can be partitioned as follows:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{12}^T & \mathbf{A}_{22} \end{bmatrix} \quad (84)$$

\mathbf{A}_{11} is the (6×6) inertia matrix of the composed link 0, which is composed of the inertia of all the links referred to frame R_0 (the base).

\mathbf{A}_{22} is the $(n \times n)$ inertia matrix of the other links when the head is fixed,

\mathbf{A}_{12} is the $(6 \times n)$ coupled inertia matrix of the joints and the base. It reflects the effect of the joint accelerations on the base motion, and the dual effect of base accelerations on the joint motions.

\mathbf{H} is the $(n+6) \times 1$ vector representing the Coriolis, centrifugal, gravity and external forces effect on the robot. Its elements are functions of the base and joint velocities and the external forces. This vector can be partitioned as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \end{bmatrix} \quad (85)$$

where:

\mathbf{H}_1 the Coriolis, centrifugal, gravity and external forces on the base.

\mathbf{H}_2 the Coriolis, centrifugal, gravity and external forces on the links $1, \dots, n$.

The inverse dynamic model gives the joint torques and the base acceleration in terms of the desired trajectory (position, velocity and acceleration) of the articulated system (links 1 to n) and the base position and velocity. Using equation (83) and (84), the inverse dynamic model is solved by using the first row of equation (83) to obtain the base acceleration:

$${}^0\dot{\mathbf{V}}_0 = -(\mathbf{A}_{11})^{-1} (\mathbf{H}_1 + \mathbf{A}_{12}\ddot{\mathbf{q}}) \quad (86)$$

Then the second row of (83), can be used to find the joint torques:

$$\Gamma = \mathbf{A}_{12}^T \dot{\mathbf{V}}_0 + \mathbf{A}_{22} \ddot{\mathbf{q}} + \mathbf{H}_2 \quad (87)$$

The direct dynamic model gives the joint accelerations and the base acceleration in terms of the position and velocity of the base and the articulated system and the joint input torques. Thus using (83), the direct dynamic model is solved as follows:

$$\begin{bmatrix} \dot{\mathbf{V}}_0 \\ \ddot{\mathbf{q}} \end{bmatrix} = \mathbf{A}^{-1} \begin{bmatrix} -\mathbf{H}_1 \\ \Gamma - \mathbf{H}_2 \end{bmatrix} \quad (88)$$

The calculation of \mathbf{A} and \mathbf{H} can be done by Lagrange method. They can also be calculated using the inverse dynamic model of tree structure of section (3.2) and using the procedure of section (3.3.1). The base can be taken into account by either of the following methods:

- The velocity and acceleration of the base will be the initial conditions $\dot{\mathbf{V}}_0$ and $\boldsymbol{\omega}_0$ for the forward

recursive calculation. The backward recursive calculation must continue to $j=0$, where this new iteration will obtain the 6 equations of Newton-Euler equations of the base.

- We can assign link 1 to be the base, and suppose that link 0 is a virtual link whose inertial parameters are equal to zero but has the velocity and acceleration of the base. This can be done by putting $\sigma_2=2$. The six equations of the base will be those of $\mathbf{f}_1 = 0$;

Solving the inverse and direct dynamic problems using \mathbf{A} and \mathbf{H} may be very time consuming for systems with big number of degrees of freedom (as the eel like robot). Therefore, we propose here to use a recursive method, which is easy to programme, and its computational complexity is linear wrt the number of degrees of freedom.

The recursive Newton-Euler algorithm is based on the kinematic equations presented in section 3.

7.2 Recursive NE Calculation of the Inverse Dynamic Model of Robots with Mobile Base

The inverse dynamic algorithm in this case consists of three recursive equations (a forward, then a backward, then a forward).

i) Forward recursive calculation:

In this step we calculate the screw transformation matrices, link velocities, and the elements of the accelerations and external wrenches on the links, which are independent of the acceleration of the robot base ($\dot{\mathbf{V}}_0, \boldsymbol{\omega}_0$). Thus we calculate for $j=1, \dots, n$: ${}^j\mathbb{T}_i$, ${}^j\mathbf{V}_j$ and ${}^j\boldsymbol{\gamma}_j$ using equations (13)-(17). We calculate also ${}^j\boldsymbol{\beta}_j$ representing the elements of the Newton-Euler equations, which are independent of the base acceleration in equations (14) and (15) such that:

$${}^j\boldsymbol{\zeta}_j = {}^j\boldsymbol{\gamma}_j + \ddot{\mathbf{q}}_j {}^j\mathbf{a}_j \quad (89)$$

$${}^j\boldsymbol{\beta}_j = -{}^j\mathbf{f}_{e_j} - \begin{bmatrix} {}^j\boldsymbol{\omega}_j \times ({}^j\boldsymbol{\omega}_j \times {}^j\mathbf{M}\mathbf{S}_j) \\ {}^j\boldsymbol{\omega}_j \times ({}^j\mathbf{J}_j {}^j\boldsymbol{\omega}_j) \end{bmatrix} \quad (90)$$

ii) Backward recursive equations:

In this step we obtain the base acceleration using the inertial parameters of the composite link 0, where the composite link j consists of the links $j, j+1, \dots, n$.

We note that (32), giving the equilibrium equation of link j , can be rewritten using (90) as:

$${}^j\mathbf{f}_j = {}^j\mathbb{J}_j {}^j\dot{\mathbf{v}}_j - {}^j\boldsymbol{\beta}_j + \sum_k {}^k\mathbb{T}_j^T {}^k\mathbf{f}_k \quad (91)$$

Applying the Newton-Euler equations on the composite link j , we obtain:

$${}^j\mathbf{f}_j = {}^j\mathbb{J}_j {}^j\dot{\mathbf{v}}_j - {}^j\boldsymbol{\beta}_j + \sum_{s(j)} {}^{s(j)}\mathbb{T}_j^T \left({}^{s(j)}\mathbb{J}_{s(j)} {}^{s(j)}\dot{\mathbf{v}}_{s(j)} - {}^{s(j)}\boldsymbol{\beta}_{s(j)} \right) \quad (92)$$

Where $s(k)$ means all the links succeeding joint j , that is to say joining j to any terminal link.

Substituting for ${}^{s(j)}\dot{\mathbf{v}}_{s(j)}$ in terms of ${}^j\dot{\mathbf{v}}_j$ using (14), we obtain:

$${}^{s(j)}\dot{\mathbf{v}}_{s(j)} = {}^{s(j)}\mathbb{T}_j {}^j\dot{\mathbf{v}}_j + \sum_r {}^{s(j)}\mathbb{T}_r {}^r\boldsymbol{\zeta}_r \quad (93)$$

Where r denotes all links between j and $s(j)$.

From (92), we obtain:

$${}^j\mathbf{f}_j = {}^j\mathbb{J}_j^c {}^j\dot{\mathbf{v}}_j - {}^j\boldsymbol{\beta}_j^c \quad (94)$$

with:

$${}^j\mathbb{J}_j^c = {}^j\mathbb{J}_j^c + \sum_k {}^k\mathbb{T}_j^T {}^k\mathbb{J}_k^c {}^k\mathbb{T}_j \quad (95)$$

$${}^j\boldsymbol{\beta}_j^c = {}^j\boldsymbol{\beta}_j^c - \sum_k {}^k\mathbb{T}_j^T {}^k\boldsymbol{\beta}_k^c + {}^k\mathbb{T}_j^T {}^k\mathbb{J}_k^c {}^k\boldsymbol{\zeta}_k \quad (96)$$

${}^j\mathbb{J}_j^c$ is the inertial matrix of the composite link j .

For $j = 0$, and supposing ${}^0\mathbf{f}_0$ is equal to zero, we obtain using (94):

$${}^0\dot{\mathbf{v}}_0 = \left({}^0\mathbb{J}_0^c \right)^{-1} {}^0\boldsymbol{\beta}_0^c \quad (97)$$

To conclude, the recursive equations of this step consist of initialising ${}^n\mathbb{J}_n^c = {}^n\mathbb{J}_n$, ${}^n\boldsymbol{\beta}_n^c = {}^n\boldsymbol{\beta}_n$ and then calculating (95)-(96) for $j = n, \dots, 0$. At the end ${}^0\dot{\mathbf{v}}_0$ is calculated by (97).

Comparing (97) with (68) we can deduce that \mathbf{A}_{11} is equal to ${}^0\mathbb{J}_0^c$, whereas ${}^0\boldsymbol{\beta}_0^c$ is equal to $(\mathbf{H}_1 + \mathbf{A}_{12}\dot{\mathbf{q}})$.

iii) Forward recursive equations:

After calculating ${}^0\dot{\mathbf{v}}_0$, the wrench ${}^j\mathbf{f}_j$ and the joint torques are obtained using equations (6) and (22) for $j = 1, \dots, n$ as:

$${}^j\dot{\mathbf{v}}_j = {}^j\mathbb{T}_i {}^i\dot{\mathbf{v}}_i + {}^j\boldsymbol{\zeta}_j \quad (98)$$

$${}^j\mathbf{f}_j = \begin{bmatrix} {}^j\mathbf{f}_j \\ {}^j\mathbf{m}_j \end{bmatrix} = {}^j\mathbb{J}_j^c {}^j\dot{\mathbf{v}}_j - {}^j\boldsymbol{\beta}_j^c \quad (99)$$

The joint torque is calculated by projecting ${}^j\mathbf{f}_j$ on the joint axis, and by taking into account the friction and the actuators inertia:

$$\Gamma_j = {}^j\mathbf{f}_j^T {}^j\mathbf{a}_j + F_{sj} \text{sign}(\dot{q}_j) + F_{vj} \dot{q}_j + I_{aj} \ddot{q}_j \quad (100)$$

It is to be noted that the inverse dynamic model algorithm can be used in the dynamic simulation of the mobile robot when the objective is to study the effect of the joint motions on the base. In this case the joint positions, velocities and accelerations trajectories are given. At each sampling time the acceleration of the base will be integrated to provide the angular and linear velocities for the next sampling time.

7.3 Recursive Direct Dynamic Model

The direct dynamic model consists of three recursive calculations in the same order as those of the inverse dynamic model (forward, backward and forward):

i) Forward recursive equations:

We calculate the link Cartesian velocities using (13) and the terms of Cartesian accelerations and equilibrium equations of the links that are independent of the accelerations of the base and of the joints. We calculate the following recursive equations for $j = 1, \dots, n$:

$${}^j\boldsymbol{\gamma}_j = \begin{bmatrix} {}^j\mathbf{R}_i \left[{}^i\boldsymbol{\omega}_i \times ({}^i\boldsymbol{\omega}_i \times {}^i\mathbf{P}_j) \right] + 2\sigma_j ({}^i\boldsymbol{\omega}_i \times \dot{q}_j {}^i\mathbf{a}_j) \\ \bar{\sigma}_j {}^i\boldsymbol{\omega}_i \times \dot{q}_j {}^i\mathbf{a}_j \end{bmatrix} \quad (101)$$

$${}^j\boldsymbol{\beta}_j = -{}^j\mathbf{f}_{ej} - \begin{bmatrix} {}^j\boldsymbol{\omega}_j \times ({}^j\boldsymbol{\omega}_j \times {}^j\mathbf{M}\mathbf{S}_j) \\ {}^j\boldsymbol{\omega}_j \times ({}^j\mathbf{J}_j {}^j\boldsymbol{\omega}_j) \end{bmatrix} \quad (102)$$

ii) Backward recursive equations:

In this second step, we first initialise ${}^n\mathbb{J}_n^* = {}^n\mathbb{J}_n$, ${}^n\boldsymbol{\beta}_n^* = {}^n\boldsymbol{\beta}_n$ and then we calculate for $j = n, \dots, 1$ the following elements, which permit to calculate ${}^j\mathbf{f}_j$ and \ddot{q}_j in terms of ${}^i\dot{\mathbf{v}}_i$ and will be used in the third recursive equations (these matrices can be obtained using a similar procedure as for the direct dynamic model of rigid links):

$$\mathbf{H}_j = {}^j\mathbf{a}_j^T {}^j\mathbb{J}_j^* {}^j\mathbf{a}_j + I_{aj} \quad (103)$$

$${}^j\mathbb{K}_j = {}^j\mathbb{J}_j^* - {}^j\mathbb{J}_j^* {}^j\mathbf{a}_j \mathbf{H}_j^{-1} {}^j\mathbf{a}_j^T {}^j\mathbb{J}_j^* \quad (104)$$

$${}^i\mathbb{J}_i^* = {}^i\mathbb{J}_i + {}^j\mathbb{T}_i^T {}^j\mathbb{K}_j {}^j\mathbb{T}_i \quad (105)$$

$$\boldsymbol{\tau}_j = \boldsymbol{\Gamma}_j - \mathbf{F}_{sj} \text{sign}(\dot{\mathbf{q}}_j) - \mathbf{F}_{vj} \dot{\mathbf{q}}_j \quad (106)$$

$${}^j\boldsymbol{\alpha}_j = {}^j\mathbb{K}_j {}^j\boldsymbol{\gamma}_j + {}^j\mathbb{J}_j^* {}^j\mathbf{a}_j \mathbf{H}_j^{-1} (\boldsymbol{\tau}_j + {}^j\mathbf{a}_j^T {}^j\boldsymbol{\beta}_j^*) - {}^j\boldsymbol{\beta}_j^* \quad (107)$$

$${}^i\boldsymbol{\beta}_i^* = {}^i\boldsymbol{\beta}_i - {}^j\mathbb{T}_i^T {}^j\boldsymbol{\alpha}_j \quad (108)$$

iii) Forward recursive equations:

At first, the base acceleration is calculated by the following relation:

$${}^0\dot{\mathbf{V}}_0 = ({}^0\mathbb{J}_0^*)^{-1} {}^0\boldsymbol{\beta}_0^* \quad (109)$$

We note that ${}^0\boldsymbol{\beta}_0^*$ is a function of $\boldsymbol{\tau}$, whereas ${}^0\boldsymbol{\beta}_0^c$

(used in the inverse model) is a function of $\ddot{\mathbf{q}}$.

$\ddot{\mathbf{q}}_j$ and ${}^j\mathbf{f}_j$ (if desired) are calculated for $j=1, \dots, n$ using the following equations:

$$\ddot{\mathbf{q}}_j = \mathbf{H}_j^{-1} \left[-{}^j\mathbf{a}_j^T {}^j\mathbb{J}_j^* ({}^j\dot{\mathbf{V}}_{j-1} + {}^j\boldsymbol{\gamma}_j) + \boldsymbol{\tau}_j + {}^j\mathbf{a}_j^T {}^j\boldsymbol{\beta}_j^* \right] \quad (110)$$

$${}^j\mathbf{f}_j = {}^j\mathbb{K}_j {}^j\mathbb{T}_i^T {}^i\dot{\mathbf{V}}_i + {}^j\boldsymbol{\alpha}_j \quad (111)$$

where:

$${}^j\dot{\mathbf{V}}_j = {}^j\dot{\mathbf{V}}_{j-1} + {}^j\mathbf{a}_j \ddot{\mathbf{q}}_j + {}^j\boldsymbol{\gamma}_j \quad (112)$$

8 CONCLUSIONS

This paper presents the inverse and direct dynamic modeling of different robotics systems. The dynamic models are developed using the recursive Newton-Euler formalism. The inverse model provides the torque of the joint and the acceleration of the free degrees of freedom such as the elastic joints, or the acceleration of the base in case of mobile base.

The direct model provides the joint acceleration of the joints including those of the free degrees of freedom.

These algorithms constitute the generalization of the algorithms of articulated manipulators to the other cases.

The proposed methods have been applied on more complicated systems such as:

- flexible link robots (Boyer and Khalil, 1998),
- Micro continuous system (Boyer, Porez and Khalil, 2006),
- hybrid structure, where the robot is composed of parallel modules, which are connected in serie, (Ibrahim, Khalil 2010).

REFERENCES

- Angeles J. 2002. *Fundamentals of Robotic Mechanical Systems*. Second edition, Springer-Verlag, New York.
- Armstrong W.W., 1979. Recursive solution to the equation of motion of an N-links manipulator. In *Proc. 5th World Congress on Theory of Machines and Mechanisms*, p. 1343-1346.
- Boyer, F., Khalil, W., 1998. An efficient calculation of flexible manipulator inverse dynamic. In, *Int. Journal of Robotics Research*, vol. 17, No.3, pp.282-293
- Boyer, F., Porez M., Khalil, W. 2006. Macro-continuous torque algorithm for a three-dimensional eel-like robot. In *IEEE Robotics transaction*, vol.22, No.4, 2006, pp.763-775.
- Brandl H., Johanni R., Otter M., .1986. A very efficient algorithm for the simulation of robots and multibody systems without inversion of the mass matrix. In *Proc. IFAC Symp. on Theory of Robots*, Vienne, p. 365-370.
- Craig J.J., 1986. *Introduction to robotics: mechanics and control*. Addison Wesley Publishing Company, Reading.
- Featherstone R., 1983. The calculation of robot dynamics using articulated-body inertias. In the *Int. J. of Robotics Research*, Vol. 2(3), p. 87-101.
- Gautier M., Khalil W. 1990. Direct calculation of minimum set of inertial parameters of serial robots. In *IEEE Trans. on Robotics and Automation*, Vol. RA-6(3), p. 368-373.
- Ibrahim, O., Khalil, W., 2010. Inverse and direct dynamic models of Hybride robots. In *Mechanism and machine theory*, Volume 45, Issue 4, p. 627-640.
- Luh J.Y.S., Walker M.W., Paul R.C.P., 1980. On-line computational scheme for mechanical manipulators. In *Trans. of ASME, J. of Dynamic Systems, Measurement, and Control*, Vol. 102(2) p. 69-76.
- Khalil W., Kleinfinger J.-F., 1986. A new geometric notation for open and closed-loop robots. In *Proc. IEEE Int. Conf. on Robotics and Automation*, San Francisco, p. 1174-1180.
- Khalil W., Kleinfinger J.-F., 1987. Minimum operations and minimum parameters of the dynamic model of tree structure robots. In *IEEE J. of Robotics and Automation*, Vol. RA-3(6), p. 517-526.
- Khalil W., Bennis F., 1994. Comments on Direct Calculation of Minimum Set of Inertial Parameters of Serial Robots. In *IEEE Trans. on Rob. & Automation*, Vol. RA-10(1), p. 78-79.
- Khalil W., Creusot D., 1997. SYMORO+: a system for the symbolic modelling of robots. In *Robotica*, Vol. 15, p. 153-161.
- Khalil W., Gautier M., 2000. Modeling of mechanical systems with lumped elasticity", In *Proc. IEEE Int. Conf. on Robotics and Automation*, San Francisco, p. 3965-3970.
- Khalil, W., Dombre, E. 2002. *Modeling identification and control of robots*. Hermes, Penton-Sciences, London.
- Khalil W. and Guegan S., 2004. Inverse and Direct Dynamic Modeling of Gough-Stewart Robots. In

- IEEE Transactions on Robotics and Automation*, 20(4), p. 754-762.
- Khalil W., G. Gallot G., Boyer F., 2007. Dynamic Modeling and Simulation of a 3-D Serial Eel-Like Robot. In *IEEE Transactions on Systems, Man and Cybernetics, Part C: Application and reviews, Vol. 37, N° 6*.
- Khalil W., Ibrahim O., 2007. General solution for the Dynamic modeling of parallel robots. In *Journal of Intelligent and Robotic Systems*, Vol.49, pp.19-37.
- Khosla P.K., 1986. Real-time control and identification of direct drive manipulators. Ph. D. Thesis, Carnegie Mellon.
- Walker M.W., Orin D.E., 1982. Efficient dynamic computer simulation of robotics mechanism. In *Trans. of ASME, J. of Dynamic Systems, Measurement, and Control*, Vol. 104, p. 205-211.

BRIEF BIOGRAPHY

Wisama Khalil received the Ph.D. and the “Doctorat d’Etat” degrees in robotics and control engineering from the University of Montpellier, France, in 1976 and 1978, respectively. Since 1983, he has been a Professor at the Automatic Control and Robotics Department, Ecole Centrale de Nantes, France. He is the coordinator of Erasmus Mundus master course EMARO “European Master in Advanced Robotics”. He is carrying out his research within the Robotics team, Institut de Recherche en Communications et Cybernétique de Nantes (IRCCyN). His current research interests include modeling, control, and identification of robots. He has more than 100 publications in journals and international conferences.

EMOTIVE DRIVER ADVISORY SYSTEM

Oleg Gusikhin

*Ford Motor Company, Research and Innovation Center
2101 Village Road, Dearborn, MI 48121, U.S.A.*

EXTENDED ABSTRACT

In 2007, Ford, in cooperation with Microsoft, introduced an in-car communication and entertainment system, SYNC. This system enables Bluetooth and USB connectivity for consumer phones and MP3 players and allows hands-free voice-activated control of brought-in devices. Since its initial introduction, there has been rapid growth of SYNC-enabled services, such as remote monitoring of vehicle health, personalized traffic reports, weather, news, and turn-by-turn directions utilizing data-over-voice technology. Furthermore, SYNC takes advantage of existing networking capabilities of smart phones/PDAs by providing a SYNC API to mobile application developers.

The Emotive Driver Advisory System (EDAS) is a Ford Research project that fills the technology pipeline for future SYNC versions, exploiting advances in information technology and consumer electronics to enhance the driver's experience. EDAS was inspired by recent developments in affective computing, open mic grammar-based speech recognition, embodied conversational agents, and humanoid robotics focusing on personalization and context-aware adaptive and intelligent behavior. The EDAS concept was revealed at the 2009 Consumer Electronics Show and the 2009 North American International Auto Show as EVA, Emotive Voice Activation.

The core elements of EDAS include an emotive and natural spoken dialogue system and an AVATAR-based visual interface integrated with adaptive vehicle controls and cloud-based infotainment. The system connects the vehicle, the driver, and the environment, while providing the dialogue strategy best suited for the given driving context and emotive status of the driver.

Voice interaction is the prevalent method for the driver to interface with vehicle systems for hands-free, eyes-free communication. The effectiveness of such communication depends on the quality and sophistication of both speech recognition and speech generation. The EDAS spoken dialogue system allows recognition of the driver's commands in an

open mic, natural, non-hierarchical manner. In turn, the system response depends on the driving environment, as well as the driver's status. The responses can be more extensive and engaging in open road conditions, while concise in high traffic situations. The ability to recognize the driver's emotions and generate emotions in response can further improve such communication. The spoken interface is augmented by the AVATAR as a universal intelligent gauge. The AVATAR supplements the emotive intent in delivering system messages, as well as providing non-verbal cues into the status of the active task.

The system leverages cloud-based infotainment, allowing for personalized, context-aware and interactive delivery of infotainment services. The ability to maintain connectivity between vehicle systems and the internet not only gives access to a vast amount of up-to-date information, but also allows outsourcing of computationally intensive tasks to a remote server, tapping into the power of cloud computing. Specifically, we demonstrate how EDAS enhances four most common in-vehicle infotainment activities: points of interest, news radio, music and refueling notification and advice.

BRIEF BIOGRAPHY

Dr. Oleg Gusikhin is a Technical Leader at Ford Manufacturing, Vehicle Design and Safety Research Laboratory. He received his Ph.D. from the St. Petersburg Institute of Informatics and Automation of Russian Academy of Sciences and an MBA from the Ross Business School at the University of Michigan. For over 15 years, he has been working at Ford Motor Company in different functional areas including Information Technology, Advanced Electronics Manufacturing, and Research & Advanced Engineering. During his tenure at Ford, Dr. Gusikhin has been involved in the design and implementation of advanced information technology and intelligent controls for manufacturing and vehicle systems. Dr. Gusikhin is a recipient of 2004 Henry Ford Technology Award and two Ford

Research and Advanced Engineering Technical Achievement Awards. He holds 2 patents and is a co-author of 8 patent applications on advanced vehicle infotainment technology.

FINGERTIP FORCE MEASUREMENT BY IMAGING THE FINGERNAIL

John Hollerbach

University of Utah, U.S.A.

Abstract: Shear and normal forces from fingertip contact with a surface are measured by external camera images of the fingernail. Due to mechanical interaction between the surface, fingertip bone, and fingernail, regions of tension or compression are set up that result in reddening or whitening due to blood flow. The effect is quantitative enough to serve as a transducer of fingertip force. Due to individual differences, calibration is required for the highest accuracy. Automated calibration is achieved by use of a magnetically levitated haptic interface probe.

BRIEF BIOGRAPHY

John M. Hollerbach is Professor of Computing, and Research Professor of Mechanical Engineering, at the University of Utah. He also directs the Robotics Track, a joint graduate program between the School of Computing and Department of Mechanical Engineering. From 1989-1994 he was the Natural Sciences and Engineering/Canadian Institute for Advanced Research Professor of Robotics at McGill University, jointly in the Departments of Mechanical Engineering and Biomedical Engineering. From 1982-1989 he was on the faculty of the Department of Brain and Cognitive Sciences and a member of the Artificial Intelligence Laboratory at MIT; from 1978-1982 he was a Research Scientist. He received his BS in chemistry ('68) and MS in mathematics ('69) from the University of Michigan, and SM ('75) and PhD ('78) from MIT in Computer Science. He is presently the Vice President for Technical Activities of the IEEE Robotics and Automation Society, and Editor of the International Journal of Robotics Research.

INTELLIGENT CONTROL SYSTEMS AND OPTIMIZATION

FULL PAPERS

A MORPHING WING USED SHAPE MEMORY ALLOY ACTUATORS NEW CONTROL TECHNIQUE WITH BI-POSITIONAL AND PI LAWS OPTIMUM COMBINATION

Part 1: Design Phase

Teodor Lucian Grigorie, Andrei Vladimir Popov, Ruxandra Mihaela Botez
École de Technologie Supérieure, Montréal, Québec H3C 1K3, Canada
lgrigore@elth.ucv.ro, andrei-vladimir.popov.1@ens.etsmtl.ca, ruxandra.botez@etsmtl.ca

Mahmoud Mamou, Youssef Mébarki
National Research Council, Ottawa, Ontario K1A 0R6, Canada
{Mahmoud.Mamou, Youssef.Mebarki}@nrc-cnrc.gc.ca

Keywords: Morphing Wing, Shape Memory Alloy Actuators, Bi-positional and PI Control Design, Numerical Simulations.

Abstract: The paper presents the design phase of the actuators control system development for a morphing wing application. Some smart materials, like Shape Memory Alloy (SMA), are used as actuators to modify the upper surface of the wing made of a flexible skin. The actuations lines control is designed and validated using a numerical simulation model developed in Matlab/Simulink. The finally adopted control law is a combination of a bi-positional law and a PI law; the control must behave like a switch between cooling phase and heating phase, situations where the output current is 0 A, or is controlled by a law of PI type. The PI controller, for the heating phase, is optimally tuned using the Ziegler-Nichols criterion and the linear model obtained using the System Identification Toolbox of Matlab. The controlled linearized system for heating phase is numerically tested in terms of time response, stability, controllability and the observability. In the actuation control design final phase, numerical simulations, based on SMA non-linear analytical model, were used for validation.

1 INTRODUCTION

Many researches are made around the world in the new challenge field related to the morphing aircraft, with the purpose to improve operational efficiency, particularly by reducing fuel consumption (Chang, 2009, Smith, 2007, Hinshaw, 2009, Gonzalez, 2005, Namgoong, 2006, Majji, 2007, and Ruotsalainen, 2009). Therefore, a lot of architecture were and are still imagined, designed, studied and developed, for this new concept application. One of these is our team project including the numerical simulations and experimental multidisciplinary studies using the wind tunnel for a morphing wing equipped with a flexible skin, smart material actuators and pressure sensors. The aim of these studies is to develop an automatic system that, based on the information related to the pressure distribution along the wing chord, moves the transition point from the laminar to

the turbulent regime closer to the trailing edge in order to obtain a larger laminar flow region, and, as a consequence, a drag reduction.

The objective of here presented research work was to develop an actuation control concept for a new morphing mechanism using smart materials, like Shape Memory Alloy (SMA), as actuators. These actuators modify the flexible upper surface of the wing, changing the airfoil shape. The morphing wing project was developed by Ecole de Technologie Supérieure in Montréal, Canada, in collaboration with Ecole Polytechnique in Montreal and the Institute for Aerospace Research at the National Research Council Canada (IAR-NRC).

To achieve the aerodynamic imposed purpose, a first phase of the studies involved the determination of some optimized airfoils available for 35 different flow conditions (five Mach numbers and seven angles of attack combinations). The optimized

airfoils were derived from a laminar WTEA-TE1 reference airfoil (Khalid, 1993, and Khalid, 1993), and were used as a starting point for the actuation system design. The transition point position estimation is made using the information received from a pressure system sensors (optical and Kulite types) equipping the upper face of the wing. Two architectures were developed for morphing system: open loop and closed loop. The difference between the two architectures is given by using or not using the position of transition point as a feedback signal for the actuation lines control. Here described work was developed in the open loop phase; in this phase were made numerical and experimental studies related to the aerodynamics of the morphed wing, to the flexible skin realization, to the actuation system, to the control of the actuation system, and, also, to the real-time determination and visualization of the transition point position using the pressure sensor system. Here, the pressure sensors using is limited to the monitoring of the pressure distribution and of the RMS pressure distribution in the boundary layer.

2 ARCHITECTURE OF THE CONTROLLED MORPHING WING SYSTEM

The chosen wing model was a rectangular one, with a reference airfoil WTEA-TE1, a chord of 0.5 m and a span of 0.9 m. The model was equipped with a flexible skin made of composite materials (layers of carbon and Kevlar fibers in a resin matrix) morphed by two actuation lines (Fig. 1). Each actuation line uses SMA wires as actuators. In the same time, 32 pressure sensors (16 optical sensors and 16 Kulite sensors), were disposed on the flexible skin in different positions along of the chord. The sensors are positioned on two diagonal lines at an angle of 15 degrees from centerline. The rigid lower structure was made from Aluminum, and was designed to allow space for the actuation system and wiring.

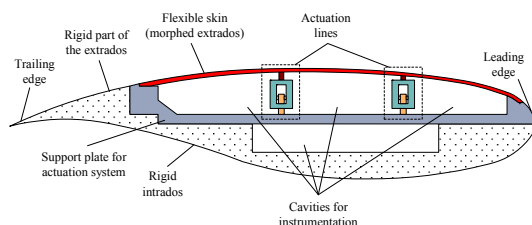


Figure 1: General architecture of the mechanical model.

Starting from the reference airfoil, depending on

different flow conditions, 35 optimized airfoils were calculated for the desired morphed positions of the airfoil. The flow conditions were established as combinations of seven incidence angles (-1° , -0.5° , 0° , 0.5° , 1° , 1.5° , 2°) and five Mach numbers (0.2, 0.225, 0.25, 0.275, 0.3). Each of the calculated optimized airfoils must be able to keep the transition point as much as possible near the trailing edge.

The SMA actuator wires are made of nickel-titanium, and contract like muscles when electrically driven. Also, these have the ability to personalize the association of deflections with the applied forces, providing in this way a variety of shapes and sizes extremely useful to achieve actuation system goals. How the SMA wires provide high forces with the price of small strains, to achieve the right balance between the forces and the deformations, required by the actuation system, a compromise must be established. Therefore, the structural components of the actuation system must be designed to respect the capabilities of actuators to accommodate the required deflections and forces.

Each of our actuation lines uses three shape memory alloys wires (1.8 m in length) as actuators, and contains a cam, which moves in translation relative to the structure (on the x -axis in Fig. 2). The cam causes the movement of a rod related on the roller and on the skin (on the z -axis). The recall used is a gas spring. So, when the SMA is heating the actuator contracts and the cam moves to the right, resulting in the rise of the roller and the displacement of the skin upwards. In contrast, the cooling of the SMA results in a movement of the cam to the left, and thus a movement of the skin down. The horizontal displacement of each actuator is converted into a vertical displacement at a rate 3:1 (results a cam factor $c_f=1/3$). From the optimized airfoils, an approximately 8 mm maximum vertical displacement was obtained for the rods, so, a 24 mm maximum horizontal displacement must be actuated.

3 SMA ACTUATORS CONTROL DESIGN AND NUMERICAL SIMULATION

The control of SMA actuators can be achieved, in principle, using any method of position control, but the specific properties of SMA actuators, such as hysteresis, the first cycle effect and the long term changes must always be considered. Starting from the established concept of the actuation system the operating schema of the controller can be organized

as is presented in Fig. 3. Based on the 35 studied flight conditions a database of the 35 optimized airfoils can be built. Therefore, for each flight condition results a pair of optimal vertical deflections (dY_{1opt} , dY_{2opt}) for the two actuation lines. The SMA actuators morph the airfoil until the obtained vertical deflections of the two actuation lines (dY_{1real} , dY_{2real}) become equals with the required deflections (dY_{1opt} , dY_{2opt}). The morphed airfoil vertical deflections in the actuation points are measured using two position transducers. The role of the controller is to elaborate an electrical current command signal for the SMA actuators on the base of the error signals (e) between the required vertical displacements and obtained displacements. Because the two actuation lines are identical the designed controller will be valid for both of them.

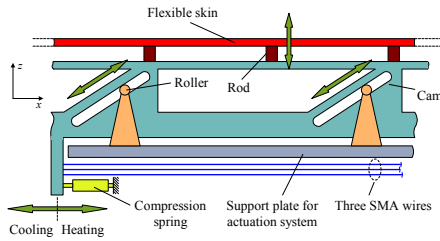


Figure 2: The actuation mechanism concept.

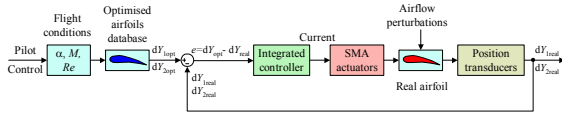


Figure 3: Operating schema of the SMA actuators control.

The first phase of the controller design supposes the numerical simulation of the controlled actuation system. Therefore, a model of SMA actuator was required. In our system a non-linear model was used (a numerical finite element one) build by Prof. P. Terriault using the theoretical model of Lickhatchev (Terriault, 2006). The SMA model has as inputs the initial temperature of the alloy, the electrical current that heats the alloy and the applied force; the outputs are the displacement of the actuator and the temperature of the alloy during functioning. According to this model, to use the shape-changing characteristics the SMA needs to be initialized by an external force, which obliges it to go initially through the transformation phase and further to revert to the initial phase through the cooling phase. Before these two phases, the control can't be realized, due to the intrinsic behavior of the SMA (Terriault, 2006, and Popov, 2008).

Looking the wing as an object moving through

the atmosphere, aerodynamic forces are generated between the air and the wing; these forces vary in function of the airflow characteristics (Mach number, Reynolds number and α - angle of attack). Since the aerodynamic forces are suction forces, it tends to lift the skin and to shorten the SMA wire. Against the aerodynamic forces action the elastic force of the flexible skin. A gas spring is needed in order to counteract the aerodynamic forces, so that the resultant force that acts on the SMA wire is given by equation

$$F_{SMA} = F_{spring} + (F_{skin} - F_{aero}) \cdot c_f. \quad (1)$$

To have the premises necessary to initialize the SMA actuators in any conditions, they are loaded by the gas spring even if there are no aerodynamic forces applied on the flexible skin. So, the equation (1) becomes

$$F_{SMA} = (F_{pretension} + k_{spring} \cdot \delta_h) + (k_{skin} \cdot \delta_v - F_{aero}) \cdot c_f, \quad (2)$$

where

$$F_{spring} = F_{pretension} + k_{spring} \cdot \delta_h, \quad F_{skin} = k_{skin} \cdot \delta_v. \quad (3)$$

F_{SMA} is the SMA resultant force, F_{spring} - gas spring elastic force, F_{skin} - elastic force produced by the flexible skin, F_{aero} - aerodynamic force, $F_{pretension}$ - pretension force of the spring, c_f - cam factor (1/3), k_{spring} and k_{skin} are the elastic coefficients of the spring, and of the skin, respectively, δ_h and δ_v are the horizontal and vertical actuated displacements.

Implementing the SMA actuators model in a Matlab S-function, the simulation model in Fig. 4 was obtained. As can be observed, to control the SMA actuators, an adequate electrical current must supply it. The length of the SMA wires is a complex function of the SMA load force and temperature, the last one being influenced by the supplying current in time and by the interaction of the wires with the environment in their cooling phase (when the electrical supply is removed) (Grigorie, 2009).

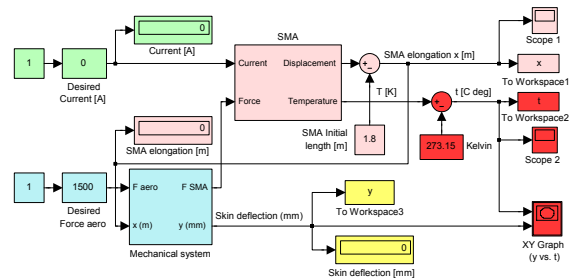


Figure 4: SMA actuators Simulink model.

The block “Mechanical system” in Fig. 4 was modeled accordingly with the equations (1) to (3). As is shown in Fig. 2, the horizontal and the vertical actuated distances are correlated by using the “cam factor” $c_f=1/3$. Therefore, the aerodynamic and the skin forces (F_{aero} and F_{skin}) are reflected in the SMA load force (F_{SMA}) with the same rate. The gas spring has a preloaded force of 1500 N and a linear elastic coefficient of 2.95 N/mm. In simulations a linear elastic coefficient of approximate 100 N/mm was considered for the skin.

The envelope of the SMA actuator, obtained through numerical simulation for different aerodynamic load cases, is shown in Fig. 5. As can be observed from Fig. 5, to obtain a skin maximum vertical displacement (8 mm) in absence of aerodynamic force, it is required a high temperature (approximately 162°C) in order to counteract the spring force. Because the ability of the SMA wires to contract is dependent upon Joule heating to produce the transformation temperature required, the need in higher temperature is reflected by a need in higher electrical current. Due to the fact that the aerodynamic forces reduce the actuators load the required current and temperature values are decreased; i.e. for $F_{aero}=1800$ N the need in temperature for the maximum vertical displacement obtaining is approximately 90°C. From other point of view, the ability of the SMA wires to return to their original configuration is dependant upon the ability of the system to cool the wires. The simulated SMA model offers just summary information about this subject, the proper heating and cooling of the wires being observed only in the moment of a thermodynamic analysis of the physical morphing wing. The system architecture play a big role in the wire cooling by the convection process, and also the performances of the system can be negatively influenced by heat transfer from actuators to the other components.

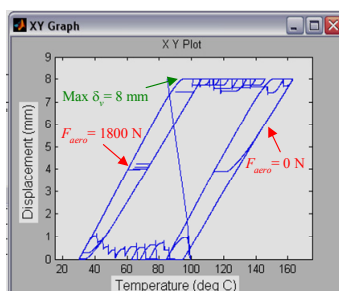


Figure 5: Simulated envelope of the SMA actuator.

According with Fig. 3, the integrated controller purpose is to control the SMA actuators in terms of

supply electrical current so that to cancel the deviation e between the required values for vertical displacements (corresponding to the optimized airfoils) and the real values, obtained from position transducers. As mentioned previously, the design of such controller is difficult considering the strong nonlinearities of the SMA actuators characteristics, nonlinearities significantly influenced by the forces with which they are tense. The chosen design procedure consisted of the following steps:

Step 1: numerical simulation of the SMA model actuators for certain values of the forces in the system;

Step 2: approximation of the system with linear systems in the heating and cooling phases using the System Identification Toolbox of Matlab and the numerical values obtained at the Step 1;

Step 3: the choice of the controller type and its tuning for each of the two SMA actuators phase – heating and cooling;

Step 4: the integration of the two obtained controller in a single one followed by its validation for the general model of the system (non-linear).

Because the team that established the actuation line architecture (Georges, 2009) suggested that the pretension force of the gas spring must have the value $F_{pretension}=1500$ N, $F_{aero}=1500$ N value was chosen in numerical simulations for the aerodynamic force. Simulating a cooling phase followed by a heating phase with the model in Fig. 4, the blue characteristics depicted in Fig. 6 were obtained. In the first graphical window of the figure is presented the SMA wire length changing in time (δ_h), while in the second window the SMA wire temperature values in the two phases are shown. One observes that a SMA wire dilatation results in the cooling phase, and a wire contraction is obtained in the heating phase. For a horizontal actuation distance of approximately 24 mm the wire temperature reaches a value near by 100°C. Note are the transient time to reach the steady-state values for the two phases: approximate 60 s for the cooling phase and approximate 40 s for the heating phase. For the steady-state, after the cooling phase, the numerical simulation obtained forces were: $F_{SMA}=1000$ N, $F_{skin}=0$ N and $F_{spring}=1500$ N. In this steady-state the system is relaxed in terms of mechanical and the vertical displacement of the actuator is null. For the steady-state, after the heating phase, the numerical simulation obtained forces were: $F_{SMA}=1337$ N, $F_{skin}=266.1$ N and $F_{spring}=1571$ N. This steady-state corresponds to the actuation system maximal vertical displacement of approximately 8 mm.

Using the Matlab System Identification Toolbox

and the numerical values characterizing the δ_h response at a series of successive step inputs, two transfer functions were found for the SMA phases:

$$H_h(s) = \frac{0.0177388 \cdot s^2 + 0.004017 \cdot s + 0.0241958}{s^3 - 1.43582 \cdot s^2 + 0.64742 \cdot s - 0.001018}, \quad (4)$$

$$H_c(s) = \frac{0.3535 \cdot s + 0.2672}{s^2 - 1.9386 \cdot s + 0.011242},$$

where $H_h(s)$ and $H_c(s)$ are the transfer functions for heating and cooling phases. The displacements δ_h , corresponding to the linear systems obtained through the two phases identification, are depicted with red line in Fig. 6. A very good approximation can be also observed for the two phases through the identification in simulated conditions. The previously established transfer functions help to the controller type choice for each phase and to its tuning.

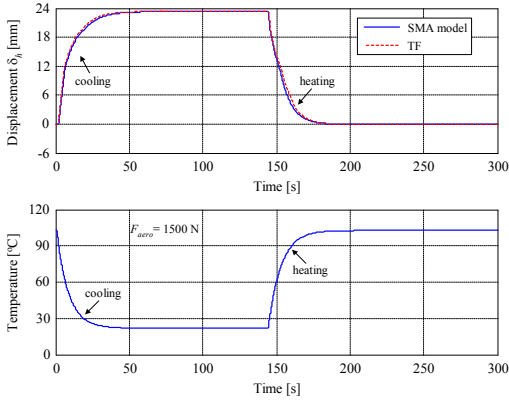


Figure 6: Actuator displacement and temperature vs time.

Considering the significance of physical controlled phenomenon, that the SMA wire must be heated to contract and then cooled to dilate by providing an appropriate electrical current by the control block, it is normal that in the cooling phase the actuators not be powered. This phase of cooling may occur in controlling not only a long-term phase, when it ordered a switch between two values of the actuator displacements, but also as a short-lived phase, which occurs when the real value of the deformation exceeds its desired value and is need to cool the actuator wires. On the other hand, it is imperative that in the heating phase actuators to be controlled so that the stationary error of the automatic system to be zero. Therefore, for this phase one opted for a simple architecture of the controller of PI type (proportional-integral). It combines the advantages of proportional type controller, which reduces substantially the overshoot and lead to a short transient time, with the benefits

of the integral controller, which cancels the steady-state system error. As a consequence, the resulted controller must behave like a switch between cooling phase and heating phase, situations where the output current is 0 A, or is controlled by a law of PI type. The two phase's interconnection leads to an integrated controller, which can be viewed as a combination of a bi-positional controller (an on-off one) and a PI (proportional-integral) controller.

The input-output characteristic of a bi-positional (on-off) controller can be described by the equation

$$i(t) = \begin{cases} -i_m, & \text{if } e \leq 0, \\ i_m, & \text{if } e > 0, \end{cases} \quad (5)$$

where $i(t)$ is the command variable (electrical current in our case) in time, i_m reflects the value of the command and e is the operating error (Fig. 3). The PI controller law is given by

$$i(t) = K_p \cdot e(t) + K_I \cdot \int e(t) \cdot dt, \quad (6)$$

with K_p - the proportional gain, and K_I - the integral gain. Combining the two controllers in a single one, based on the rules previously mentioned results the control law of the integrated controller as the form

$$i(t) = \begin{cases} 0, & \text{if } e \leq 0, \\ K_p \cdot e(t) + K_I \cdot \int e(t) \cdot dt, & \text{if } e > 0. \end{cases} \quad (7)$$

The optimal tuning of the controller in heating phase was realized using an integral criterion, the error minimum surface criterion, very well known in the literature as Ziegler-Nichols criterion (Mihoc, 1980); the tuning methodology is: a) the regulator is considered as a proportional one (P) and it is tuned with respect to the K_p parameter; b) the amplification factor K_p is increased until the response of the automatic system will be self-sustained oscillatory. One memorizes the value K_{p0} of K_p for which the system has an oscillatory behavior and the value of oscillations semi-period (T_0). The optimal values for the parameters of the PI regulator are determined using the relations:

$$K_p = 0.45 \cdot K_{p0}, \quad K_I = K_p / (0.85 \cdot T_0). \quad (8)$$

Follows the controller tuning steps the next numerical values for the PI controller parameters were obtained and/or were calculated:

$$K_{p0} = 3984, \quad T_0 = 2.68s, \\ K_p = 1792.8, \quad K_I = 787.0061. \quad (9)$$

As a consequence, the controlled system in hea-

ting phase can be modeled with an approximate linear system with the block schema in Fig. 7. The parameters $a_0 \div a_2$ and $b_0 \div b_3$ in the schema are the coefficients of the $H_h(s)$ transfer function nominator and denominator in ascending power of s (eq. (4)). The open loop transfer function of the controlled heating phase is

$$H_{ol}(s) = C_{PI}(s) \cdot H_h(s) = \frac{q_3 s^3 + q_2 s^2 + q_1 s + q_0}{b_3 s^4 + b_2 s^3 + b_1 s^2 + b_0 s}, \quad (10)$$

while the closed loop transfer function is

$$H_{cl}(s) = C_{PI}(s) \cdot H_h(s) = \frac{q_3 s^3 + q_2 s^2 + q_1 s + q_0}{r_4 s^4 + r_3 s^3 + r_2 s^2 + r_1 s + r_0}. \quad (11)$$

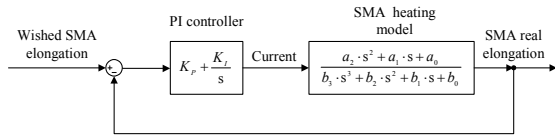


Figure 7: The block schema with transfer functions of the heating phase linear model.

The included coefficients are

$$\begin{aligned} q_3 &= K_p a_2 = 31.8021, & q_2 &= K_p a_1 + K_I a_2 = 21.1622, \\ q_1 &= K_p a_0 + K_I a_1 = 46.5396, & q_0 &= K_I a_0 = 19.0422, \end{aligned} \quad (12)$$

respectively

$$\begin{aligned} r_4 &= b_3 = 1, & r_3 &= b_2 + K_p a_2 = 30.3663, \\ r_2 &= b_1 + K_p a_1 + K_I a_2 = 21.8096, \\ r_1 &= b_0 + K_p a_0 + K_I a_1 = 46.5386, & r_0 &= K_I a_0 = 19.0422. \end{aligned} \quad (13)$$

$C_{PI}(s)$ is the transfer function of the PI controller. The poles of the close loop transfer function $H_{cl}(s)$ result with the values

$$\begin{aligned} p_1 &= -29.6837, & p_1 &\in \mathbf{R}_-, & p_4 &= -0.4453, & p_4 &\in \mathbf{R}_- \\ p_2 &= -0.1187 + 1.1943 \cdot i, & p_2 &\in \mathbf{C}, & \text{Re}(p_2) &\in \mathbf{R}_-, \\ p_3 &= -0.1187 - 1.1943 \cdot i, & p_3 &\in \mathbf{C}, & \text{Re}(p_3) &\in \mathbf{R}_-. \end{aligned} \quad (14)$$

One can observe that all poles of the transfer function are placed in the left-hand side of the s -plane, and the obtained system is stable.

In the state-space representation

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t), \end{aligned} \quad (15)$$

the state matrix A , the input matrix B , the output matrix C and the feed-forward matrix D , were obtained by the forms

$$\begin{aligned} A &= \begin{bmatrix} -30.3663 & -21.8096 & -46.5386 & -19.0422 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, & (16) \\ B^T &= [1 \ 0 \ 0], & C &= [31.8 \ 21.1 \ 46.5 \ 19], & D &= 0. \end{aligned}$$

Evaluating the controllability and observability of the system (P and Q matrices) results

$$P = \begin{bmatrix} 1 & -30.3663 & 900.3025 & -26723.119 \\ 0 & 1 & -30.3663 & 900.3025 \\ 0 & 0 & 1 & -30.3663 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (17)$$

$$Q = \begin{bmatrix} 31.8 & 21.1 & 46.5 & 19 \\ -944.5 & -647 & -1460.9 & -605.5 \\ 28035.4 & 19139.3 & 43352.4 & 17986.3 \\ -832193.6 & -568090.6 & -1286744.7 & -533857.8 \end{bmatrix}, \quad (18)$$

$$\text{rank}(P) = \text{rank}(Q) = \text{system order} = 4. \quad (19)$$

As a consequence, the system is completely controllable and observable.

Based on the previously considerations, the final form of the integrated controller law is

$$i(t) = \begin{cases} 0, & \text{if } e \leq 0, \\ 1792.8 \cdot e(t) + 787.0061 \cdot \int e(t) \cdot dt, & \text{if } e > 0. \end{cases} \quad (20)$$

Introducing the controller in a general block schema, with the non-linear SMA model, the Simulink model in Fig. 8 was obtained for the SMA actuators control (see Fig. 3). The input variable of the schema is the desired skin deflection and the output is the real skin deflection.

The ‘‘Integrated controller’’ block contains the implementation of the law described by equation (20) and of the preliminary observations related to the SMA actuators physical limitations in terms of temperature and supplying currents. Its schema is shown in Fig. 9. The inputs of the block are the control error (difference between the desired and the obtained displacements) and the wires temperature, and the output is controlled electrical current applied on the SMA actuators. There are two switches in the schema; the first one chooses one of the two options in the control law (20) and the second one switching the electrical current value to 0A when the SMA temperature value is over the imposed limit. Also, a current saturation block is used to prevent the current increase over the physical limit supported by the actuation SMA wires.

decreased; i.e. for $F_{aero}=1800$ N the need in temperature for the maximum vertical displacement obtaining is approximately 90°C.

The final configuration of the integrated controller was a combination of a bi-positional controller (particularly an on-off one) and a PI (proportional-integral) controller, due to the two phases (heating and cooling) of the SMA wires interconnection. The resulted controller must behave like a switch between cooling phase and heating phase, situations where the output current is 0 A, or is controlled by a law of PI type.

Using an integral criterion, the error minimum surface criterion (Ziegler-Nichols), the PI controller for the heating phase was optimal tuned, the resulted values are $K_p=1792.8$ and $K_i=787.0061$. Evaluating the systems' performances one observed that the poles of closed loop transfer function of the controlled heating phase resulted with the values (14) are all placed in the left-hand side of the s-plane, so the obtained system is stable. On the other way, the system was found to be completely controllable and observable based on the values established in equations (17)-(19). So, the final form of the integrated controller law was (20).

Loading the numerically simulated general model (the non-linear one with $F_{pretension}=1500$ N) in Fig. 8 with aerodynamic force $F_{aero} = 1500$ N, the obtained characteristics in Fig. 10 confirm that the controller works good, the transition to the desired steady-state being significantly improved through the integration of the two control law in the equation (20): 1) the amplitudes of oscillations were reduced and the observed oscillations in the SMA temperatures around the steady-state are due only to the thermal inertia of the smart material; 2) the values of the transition time from 0mm to the steady-state values decrease from 20÷25 to approximate 5 s.

As second and third validation methods a bench test and a wind tunnel test were performed and will be presented in the second part of the paper, related to the experimental validation.

ACKNOWLEDGEMENTS

We would like to thank the Consortium of Research in the Aerospace Industry in Quebec (CRIAQ), Thales Avionics, Bombardier Aerospace, and the National Sciences and Engineering Research Council (NSERC) for the support that made this research possible. We would also like to thank George Henri Simon for initiating the CRIAQ 7.1

project and Philippe Molaret from Thales Avionics and Eric Laurendeau from Bombardier Aeronautics for their collaboration on this work.

REFERENCES

- Chang, P., Shah, A., Singhee, M., 2009, *Parameterization of the Geometry of a Blended Wing Body Morphing Wing*, Project report, Georgia Institute of Technology, April 2009, Atlanta, Georgia, USA
- Georges, T., Brailovski, V., Morellon, E., Coutu, D., Terriault, P., 2009, *Design of Shape Memory Alloy Actuators for Morphing Laminar Wing With Flexible Extradors*, Journal of Mechanical Design, Vol. 31, N° 9
- Gonzalez, L., 2005, *Morphing Wing Using Shape Memory Alloy: a concept proposal*, Final research paper, Texas A&M University, College Station, Texas, USA
- Grigorie, T. L., Botez, R. M., 2009, *Adaptive neuro-fuzzy inference system-based controllers for smart material actuator modeling*, Journal of Aerospace Engineering, Vol. 223, No. 6, pp. 655-668
- Hinshaw, T. L., 2009, *Analysis and Design of a Morphing Wing Tip using Multicellular Flexible Matrix Composite Adaptive Skins*, Master of Science Thesis, Virginia Polytechnic Institute and State University, Virginia, USA
- Khalid, M., Jones, D. J., 1993, *Navier Stokes Investigation of Blunt Trailing Edge Airfoils using O-Grids*, AIAA Journal of Aircraft, vol.30, no.5, pp. 797-800
- Khalid, M., Jones, D. J., 1993, *A CFD Investigation of the Blunt Trailing Edge Airfoils in Transonic Flow*, Inaugural Conference of the CFD Society of Canada.
- Majji, M., Rediniotis, O. K., Junkins, J.L., 2007, *Design of a Morphing Wing: Modeling and Experiments*, AIAA Atmospheric Flight Mechanics Conference and Exhibit, Hilton Head, South Carolina, USA
- Mihoc, D., 1980, *Teoria si elementele sitemelor de reglare automata*. Editura Didactica si Pedagogica, Bucuresti
- Namgoong, H., Crossley, W. A., Lyrintzis, A. S., 2006, *Aerodynamic Optimization of a Morphing Airfoil Using Energy as an Objective*, 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, USA
- Popov, A. V., Labib, M., Fays, J., Botez, R. M., 2008, *Closed-Loop Control Simulations on a Morphing Wing*, Journal of Aircraft, Vol. 45, pp. 1794-1803
- Ruotsalainen, P., et. al., 2009, *Shape Control of a FRP Airfoil Structure Using SMA-Actuators and Optical Fiber Sensors*. Journal of Solid State Phenomena, Volume 144, pp. 196-201
- Smith, K., Butt, J., Spakovsky, M. R., Moorhouse, D., 2007, *A Study of the Benefits of Using Morphing Wing Technology in Fighter Aircraft Systems*, 39th AIAA Thermophysics Conference, Miami, Florida, USA
- Terriault, P., Viens, F., Brailovski, V., 2006, *Non-isothermal Finite Element Modeling of a Shape Memory Alloy Actuator Using ANSYS*, Computational Materials Science, Vol. 36, No. 4, pp. 397-410

A MORPHING WING USED SHAPE MEMORY ALLOY ACTUATORS NEW CONTROL TECHNIQUE WITH BI-POSITIONAL AND PI LAWS OPTIMUM COMBINATION

Part 2: Experimental Validation

Teodor Lucian Grigorie, Andrei Vladimir Popov, Ruxandra Mihaela Botez
École de Technologie Supérieure, Montréal, Québec H3C 1K3, Canada
lgrigore@elth.ucv.ro, andrei-vladimir.popov.1@ens.etsmtl.ca, ruxandra.botez@etsmtl.ca

Mahmoud Mamou, Youssef Mébarki
National Research Council, Ottawa, Ontario K1A 0R6, Canada
Mahmoud.Mamou, Youssef.Mebarki@nrc-cnrc.gc.ca

Keywords: Morphing Wing, Shape Memory Alloy Actuators Control, Experimental Validation, Bench Test, Wind Tunnel Test.

Abstract: The paper represents the second part of a study related to the development of an actuators control system for a morphing wing application, and describes the experimental validation of the control designed in the first part. After a short presentation of the finally adopted control architecture, the physical implementation of the control is done. To implement the controller on the physical model two Programmable Switching Power Supplies AMREL SPS100-33 and a Quanser Q8 data acquisition card, were used. The inputs of the data acquisition were two signals from Linear Variable Differential Transformer potentiometers, indicating the positions of the actuators, and six signals from thermocouples installed on the SMA wires. The acquisition board outputs channels were used to control power supplies in order to obtain the desired skin deflections. The control validation was made in two experimental ways: bench test and wind tunnel test. All 35 optimized airfoil cases, used in the design phase, were converted into actuators vertical displacements which were used as inputs reference for the controller. In the wind tunnel tests a comparative study was realized around of the transition point position for the reference airfoil and for each optimized airfoil.

1 INTRODUCTION

The spectacular and continuous evolution of the aerospace engineering domain was highlighted in the last years especially through the boarded equipments and systems technology development, mainly those of avionics. But, in the same time, the two related sub-domains, propulsion systems and aircraft structures, in parallel registered very important discoveries, sometimes notified to the general public too little. Thus, the concept of green aircraft launched major trends in the aerospace field research, of which can be mentioned reduction of noise and chemical pollution of the atmosphere, reduction of fuel consumption and increase of aircraft flight autonomy. This concept is a consequence of the predictions for future according with that the air traffic is seen to more than double in

the next 20 years. Therefore, both environmental and economic pressures will strongly increase and significant progress will need to be achieved in both improving the efficiency and minimizing the environmental impact of aircraft. In order to provide these required changes, aircrafts in new concepts are designed and will be developed. These suppose the validation and after that the integration of new technologies and solutions at the level of all major aircraft components: cabin, wing, power plant system, and fuselage; multidisciplinary investigations already explore the different associated aspects of aero-dynamics, acoustics, materials, structure, engines and systems. The aims of these investigations are to ensure an improved quality and affordability, whilst meeting the tightening environmental constraints (emission and noise), with a vision of global efficiency of the air

transport system.

Within this context are developed our research related to the morphing aircraft new challenge field, precisely to the morphing wing concept in this field (Chang, 2009, Smith, 2007, Hinshaw, 2009, Gonzalez, 2005, Namgoong, 2006, Majji, 2007, and Ruotsalainen, 2009). The presented work objective is to develop an actuation control concept for a new morphing mechanism using smart materials, like Shape Memory Alloy (SMA), as actuators. These smart actuators modify the upper surface of a wing made of a flexible skin so the laminar to turbulent transition point moves close to the wing airfoil trailing edge. The final purpose of the research project is to obtain a drag reduction as a function of flow condition, by changing the wing shape.

The chosen wing model was a rectangular one, with a reference airfoil WTEA-TE1, a chord of 0.5 m and a span of 0.9 m. The model was equipped with a flexible skin made of composite materials morphed by two actuation lines. Each actuation line uses shape memory alloys wires as actuators.

In the first part of this paper a control for the actuation lines of the morphing wing system was designed. In this way, 35 optimized airfoils available for 35 different flow conditions (five Mach numbers (0.2 to 0.3) and seven angles of attack (-1° to 2°) combinations) were used.

From the developed actuation mechanism results that each actuation line uses three SMA wires (1.8 m in length) as actuators, and contains a cam, which moves in translation relative to the structure (on the x-axis in Fig. 1). The cam causes the movement of a rod related on the roller and on the skin (on the z-axis). The recall used is a gas spring. The horizontal displacement of each actuator is converted into a vertical displacement at a rate 3:1, which makes that the horizontal stroke of x mm to be converted into a vertical stroke $z=x/3$; results a cam factor $c_f=1/3$, therefore, for the approximately 8 mm maximum vertical displacement, obtained from the optimized airfoils numerical data, a 24 mm maximum horizontal displacement must be actuated.

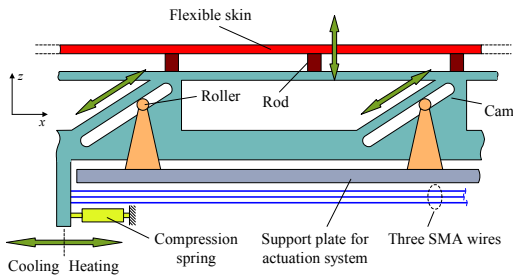


Figure 1: The actuation mechanism concept.

The designed controller controls the SMA actuators in terms of supply electrical current so that to cancel the deviation between the required values for vertical displacements (corresponding to the optimized airfoils) and the real values, obtained from two position transducers (Fig. 2).

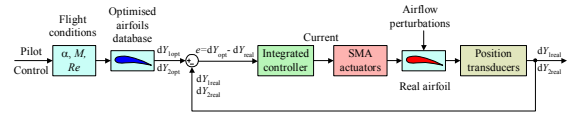


Figure 2: Operating schema of the SMA actuators control.

The finally numerical validated configuration (in the first part of the paper) of the integrated controller was a combination of a bi-positional controller (particularly an on-off one) and a PI (proportional-integral) controller, due to the two phases (heating and cooling) of the SMA wires interconnection. The resulted controller must behave like a switch between cooling phase and heating phase, situations where the output current is 0 A, or is controlled by a law of PI type

$$i(t) = \begin{cases} 0, & \text{if } e \leq 0, \\ 1792.8 \cdot e(t) + 787.0061 \cdot \int e(t) \cdot dt, & \text{if } e > 0. \end{cases} \quad (1)$$

e is the operating error (see Fig. 2).

2 PHYSICAL MORPHING WING CONTROL IMPLEMENTATION

Starting from the theoretical and numerical simulation resulted considerations to implement the controller on the physical model two Programmable Switching Power Supplies AMREL SPS100-33, controlled by Matlab through a Quanser Q8 data acquisition card, were used (Fig. 3) (Kirianaki, 2002, Park, 2003, and Austerlitz, 2003).

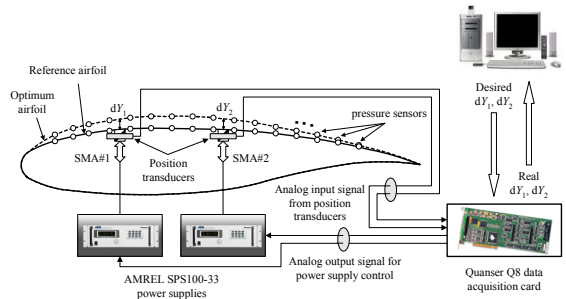


Figure 3: Physical model operating schema.

The power supplies have RS-232 and GPIB IEEE-488 as standard features and the technical characteristics: Power 3.3kW, Voltage (dc) 0÷100 V, Current (dc) 0÷33 A. The Q8 data acquisition card has 8 single-ended analog inputs with 14-bit resolution. All 8 channels can be sampled simultaneously at 100 kHz, with A/D conversion times of 2.4 μ s/channel, simultaneous sampling and sampling frequencies of up to 350 kHz for 2 channels. Also, the Q8 card is equipped with 8 analog outputs, with software programmable voltage ranges and simultaneous update capability with an 8 μ s settling time over full scale (20V).

The acquisition board was connected to a PC and programmed through Matlab/Simulink R2006b and WinCon 5.2 (Fig. 4).

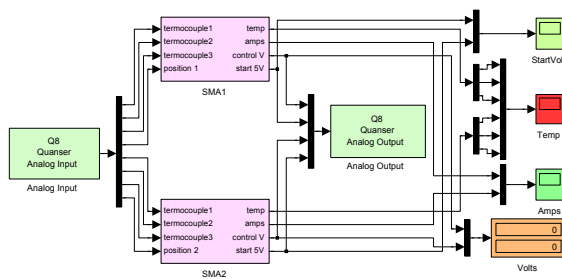


Figure 4: Simulink actuators control.

The input signals were two signals from Linear Variable Differential Transformer potentiometers that indicate the positions of the SMA actuators, and six signals from thermocouples installed on all the SMA wires components. The acquisition sampling time was set to 0.01 second. The outputs channels of the acquisition board were used to control each power supply through analog/external control by use of a DB-15 I/O connector. The current supplied to the actuator was set to be limited at 10 A, and the control signal was set to be 0÷0.6061V (maximum voltage for the power supply is 2 V for 33 A current supply).

The operation principle of the physically implemented controller is relative simple. The initial input, which is the optimized airfoil for any flow condition, is chosen manually by the operator from the computer database through a user interface. Then the displacements (dY_1 , dY_2) that are required to be reproduced by the two control points on the flexible skin are sent to the controller. This controller sends an analog signal 0 – 2 V to the power supply that provides a current to the SMA. The SMA will respond accordingly and change its length according to the temperature of the wire. This will result in a change of the actuators positions, which are sensed

by the linear variable differential transducer (LVDT). The signal position received from the LVDT is compared to the desired position and the error obtained is fed back to the controller. If the realized position is greater than the desired position the controller will disconnect the control current letting the SMA wire to cool down. During the cooling down process the SMA will maintain its length due to the hysteretic behavior. This effect is taken into account for actuators displacement. Also the controller uses three thermocouples signals from each SMA wire to monitor the temperature of the wires and maintain it below 130°C, as an upper limit.

3 SMA ACTUATORS CONTROL BENCH TEST VALIDATION

The morphing wing system in the bench test runs is shown in Fig. 5.

The gas springs that maintain the SMA wires in tension had a preloaded value of 225 lbs (1000 N) since in the laboratory condition there is no aerodynamic force.

After an initial calibration test the calibration gains and constants were established for the two LVDT potentiometers and for the six thermocouples. The calibration test for LVDT potentiometers consisted of several scans of airfoil using a laser beam. On the calibration, the SMA actuators were in “zero setting position” with no power supplied and the skin coordinates were measured using the laser beam that scanned the center line of the wing model. The laser was set to scan the chord of the model on a 370 mm length with a speed of 5 mm per second.

In the bench test, the 35 optimized airfoil cases were converted into SMA actuator #1 and #2 vertical displacements which were used as inputs reference for the controller. A typical test run history is shown in Fig. 6 for $\alpha=1^\circ$, Mach=0.3 flight condition ($dY_1=5.22$ mm, $dY_2=7.54$ mm – vertical displacements of the skin in the actuation points).

On observe that the controllers, in the two actuation lines, work even in zero values of the desired signal because of the gas springs pretension. Also, small oscillations of the obtained deflection are observed around the desired values of the deflections. The amplitude of the oscillations in this phase is due to the LVDT potentiometers mechanical link and to the inertia of the SMA wires, being smallest than 0.05 mm. The heating phase is approximately 9 times more rapidly than the cooling phase; heating time equals 8 s while the cooling time

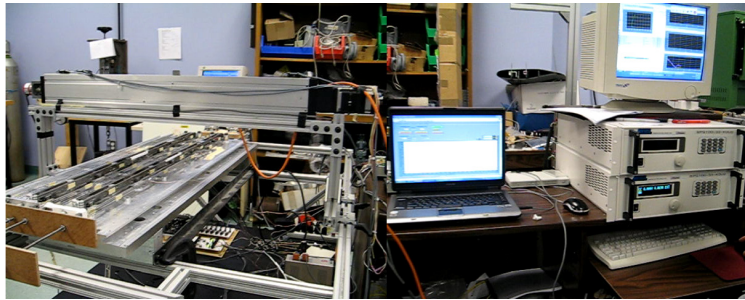


Figure 5: Morphing wing system in the bench test runs.

equals 70 s. There can be observed the differences between the numerical model of the SMA actuators and the physical model.

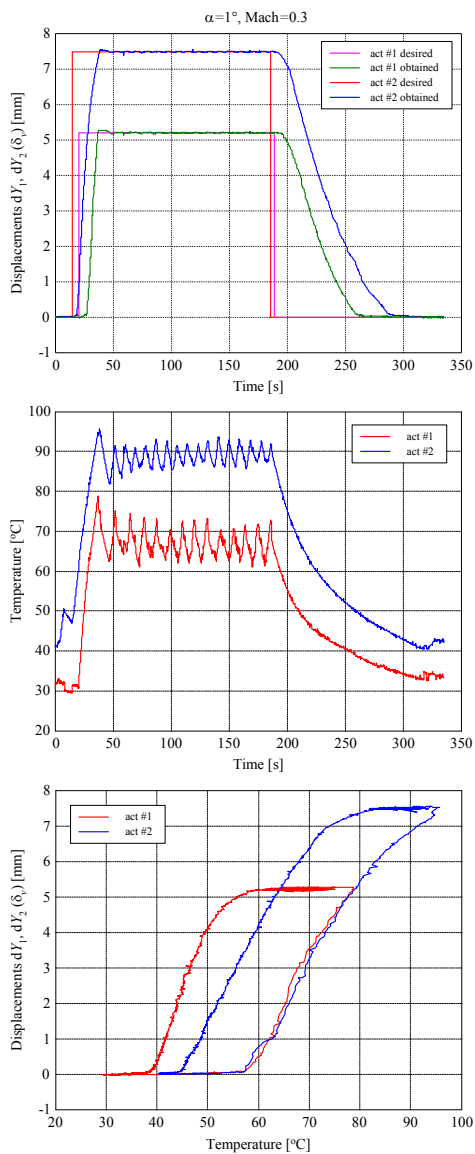


Figure 6: Bench test for $\alpha=1^\circ$, $M=0.3$ flight condition.

The bench test results confirmed that the experimental version of the designed integrated controller works well even in the lab conditions, where no aerodynamic forces are loaded and the preloaded gas springs force is 1000N.

4 SMA ACTUATORS CONTROL VALIDATION IN WIND TUNNEL TESTS

Once confirmed the well working of the designed integrated controller through bench test, the next step in our morphing wing project was to validate the controller in a wind tunnel test simultaneously with the transition point real time detection and visualization for all 35 optimized airfoils. The model was tested for all 35 theoretical studied flight conditions, a comparative study being realized around of the transition point position for the reference airfoil and for each optimized airfoil. So, simultaneously with the controller testing, a validation study for the aerodynamic part of the project was realized.

The morphing wing system in the wind tunnel runs is shown in Fig. 7.

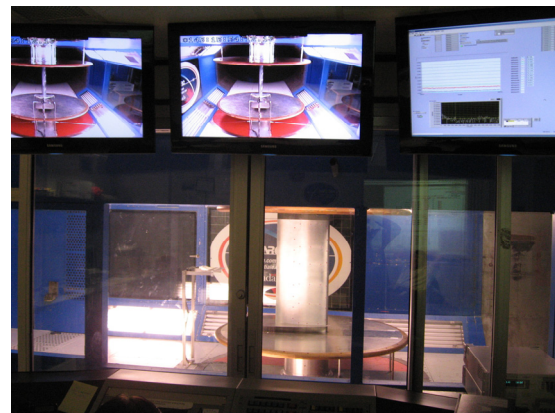


Figure 7: Wind tunnel morphing wing model.

The transition detection was made real time using the pressure data obtained from the 32 Kulite and optical pressure sensors. The pressure data acquisition was performed using the IAR-NRC analog data acquisition system which was connected to the 32 sensors. The sampling rate of each channel was 15 kS/s, which allowed a boundary layer pressure fluctuations FFT spectral decomposition up to 7.5 kHz for all channels. The signal was processed by use of Simulink and visualized in real time on the computer screen in dedicated windows.

The pressure signals were analyzed through Fast Fourier Transforms (FFT) decomposition in order to detect the magnitude of the noise in the surface air flow. Subsequently, the data is filtered by means of high-pass filters and processed by calculating the Root Mean Square (RMS) of the signal in order to obtain a plot diagram of the noise in the air flow. This signal processing is necessary to separate the inherent electronically induced noise, by the Tollmien-Schlichting that are responsible for triggering transition from laminar flow to turbulent flow. The measurements showed that in processed data the transition appeared at frequencies between 3kHz - 5kHz and the magnitude of pressure variations in the laminar flow boundary layer are of the order $5e-4$ Pa ($7.25e-8$ psi). The transition between laminar flow and turbulent flow was shown by an increase of the pressure variations, reflected also by a strong variation of the pressure signal RMS.

For the wind tunnel test the preloaded forces of the gas springs were reconsidered to the 1500 N because of the presence of the aerodynamic forces on the flexible skin of the wing. In Fig. 8 are presented the control results for test run $\alpha=2^\circ$, $Mach=0.225$ ($dY_1=5.56$ mm, $dY_2=7.91$ mm).

The experimental results show a decrease of the SMA wires work temperatures vis-à-vis of numerically simulated and bench tested cases. An explanation can be the appearing of the aerodynamic forces with particular values for each flight condition. The decrease of these temperatures is a beneficial one taking into account the negative impact of a strong thermal field on the other component of the system, especially on the flexible skin and on the pressure sensors. Also, from the experimental results a high frequency noise influencing the LVDT sensors and the thermocouples instrumentation amplifiers can be observed. The noise sources are the wind tunnel vibrations and instrumentation electrical fields. With this noise, the amplitude of the actuation error (difference between the realized deflections and

desired deflections) is under 0.07 mm, but this don't affecting the transition, which is stable on a sensor with a high RMS spike; Fig. 9 presents the results obtained on the transition monitoring for the run test in Fig. 8.

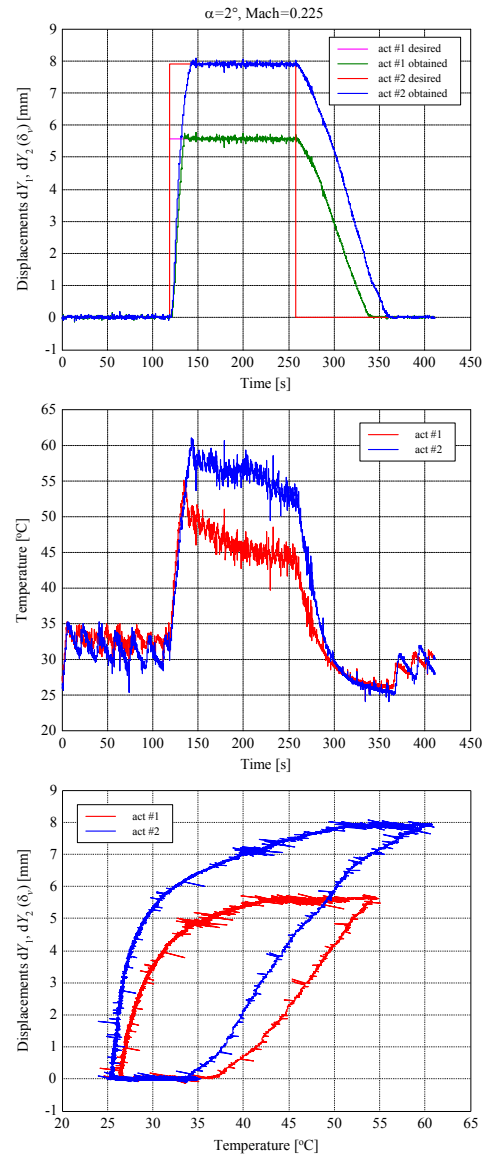


Figure 8: Wind tunnel test for $\alpha=2^\circ$, $M=0.25$ flight condition.

The actuation line control obtained precision can have some influence in the transition point position detection only if the density of the chord disposed pressure sensors becomes bigger; from the experimental data evaluation one concluded that, even the value of the error is 1 mm around the optimized values, the transition point position on the airfoil surface is not significantly changed.

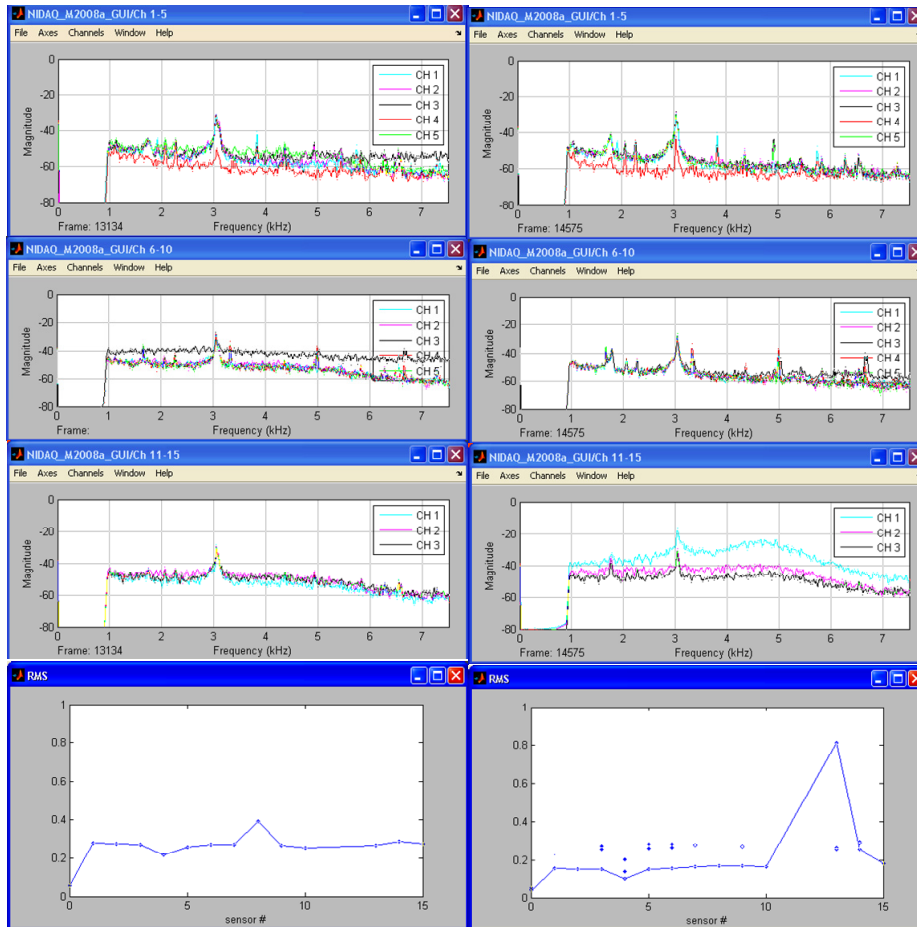


Figure 9: Results obtained on the transition monitoring for the run test in Figure 8.

In Fig. 9 are presented the outputs of the Kulite pressure sensors in leading edge – trailing edge sense of positioning (3 sensors are broken and was not considered in the monitoring phase) and the real time pressure signals RMS for each of these sensors. The left hand column presents the results for reference airfoil, and the right hand column the results for optimized airfoil. The spike of the RMS suggests that we have turbulence on the sensor no. 13, near the trailing edge.

So, the results obtained for the actuators control are very good, the controller fully satisfying the requirements imposed for the project purpose achievement.

The future work on the project supposes the development of the closed loop control, based on the pressure information received from the sensors and on the transition point position estimation. Evidently, the closed loop control will include, as an internal loop, the actuation lines here presented controller.

5 CONCLUSIONS

The paper represents the second part of a study related to the development of an actuators control system for a morphing wing application, and describes the experimental validation of the control designed in the first part. The control validation was made in two experimental ways: bench test and wind tunnel test.

In the bench test phase, the 35 optimized airfoil cases were converted into SMA actuator #1 and #2 vertical displacements which were used as inputs reference for the controller. The characteristics in Fig. 6 ($\alpha=1^\circ$, Mach=0.3 flight condition) show that the controllers, in the two actuation lines, work even in zero values of the desired signal because of the gas springs pretension. Also, small oscillations of the obtained deflection are observed around the desired values of the deflections. The amplitude of the oscillations in this phase is due to the LVDT

potentiometers mechanical link and to the inertia of the SMA wires, being smallest than 0.05 mm. The heating phase is approximately 9 times more rapidly than the cooling phase; heating time equals 8 s while the cooling time equals 70 s.

For the final experimental validation test (wind tunnel test), with real aerodynamic forces load, the 1500 N preloaded forces of the gas springs was reconsidered. From Fig. 8 ($\alpha=2^\circ$, Mach=0.225) a decrease of the SMA wires work temperatures vis-à-vis of numerically simulated and bench tested cases is observed. The decrease of these temperatures is a beneficial one taking into account the negative impact of a strong thermal field on the other component of the system, especially on the flexible skin and on the pressure sensors. Also, a high frequency noise influencing the LVDT sensors and the thermocouples instrumentation amplifiers can be observed. The noise sources are the wind tunnel vibrations and instrumentation electrical fields. With this noise, the amplitude of the actuation error (difference between the realized deflections and desired deflections) is under 0.07 mm, but this doesn't affecting the transition, which is stable on a sensor with a high RMS spike like in Fig. 9.

So, the results obtained for the actuators control are very good, the controller fully satisfying the requirements imposed for the project purpose achievement.

The designed controller is used for the open loop development stage of a morphing wing project, but the closed loop of the morphing wing system, based on the pressure information received from the sensors and on the transition point position estimation, will include, as an internal loop, the actuation lines here presented controller.

ACKNOWLEDGEMENTS

We would like to thank the Consortium of Research in the Aerospace Industry in Quebec (CRIAQ), Thales Avionics, Bombardier Aerospace, and the National Sciences and Engineering Research Council (NSERC) for the support that made this research possible. We would also like to thank George Henri Simon for initiating the CRIAQ 7.1 project and Philippe Molaret from Thales Avionics and Eric Laurendeau from Bombardier Aeronautics for their collaboration on this work.

REFERENCES

- Austerlitz, H., 2003, *Data acquisition techniques using PCs*, Elsevier, USA
- Chang, P., Shah, A., Singhee, M., 2009, *Parameterization of the Geometry of a Blended Wing Body Morphing Wing*, Project report, Georgia Institute of Technology, April 2009, Atlanta, Georgia, USA
- Gonzalez, L., 2005, *Morphing Wing Using Shape Memory Alloy: a concept proposal*, Final research paper, Texas A&M University, College Station, Texas, USA
- Hinshaw, T. L., 2009, *Analysis and Design of a Morphing Wing Tip using Multicellular Flexible Matrix Composite Adaptive Skins*, Master of Science Thesis, Virginia Polytechnic Institute and State University, Virginia, USA
- Kirianaki, N. V., Yurish, S. Y., Shpak, N. O., Deynega, V.P., 2002, *Data Acquisition and Signal Processing for Smart Sensors*. John Wiley & Sons
- Majji, M., Rediniotis, O. K., Junkins, J. L., 2007, *Design of a Morphing Wing: Modeling and Experiments*, AIAA Atmospheric Flight Mechanics Conference and Exhibit, Hilton Head, South Carolina, USA
- Namgoong, H., Crossley, W. A., Lyrintzis, A. S., 2006, *Aerodynamic Optimization of a Morphing Airfoil Using Energy as an Objective*, 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, USA
- Park, J., Mackay, S., 2003, *Practical data acquisition for instrumentation and control systems*, Elsevier, UK
- Ruotsalainen, P., et. al., 2009, *Shape Control of a FRP Airfoil Structure Using SMA-Actuators and Optical Fiber Sensors*. Journal of Solid State Phenomena, Volume 144, pp. 196-201
- Smith, K., Butt, J., Spakovsky, M. R., Moorhouse, D., 2007, *A Study of the Benefits of Using Morphing Wing Technology in Fighter Aircraft Systems*, 39th AIAA Thermophysics Conference, Miami, Florida, USA

PREDICTION OF TEMPERATURE INSIDE A REFRIGERATED CONTAINER IN THE PRESENCE OF PERISHABLE GOODS

Javier Palafox-Albarrán, Reiner Jedermann and Walter Lang
Institute of Microsensors Actuators and Systems (IMSAS), University of Bremen
Otto Hahn Allee NW1, D-28359 Bremen, Germany
{jpalafax, rjedermann, wlang}@imsas.uni-bremen.de

Keywords: System Identification, Temperature, Organic Heat, Feedback-hammerstein.

Abstract: This paper presents an alternative method to predict the temperature profile in a spatial point of the interior of a refrigerated container with the aim of improving the logistics of perishable goods. A SISO gray-box model in which the organic heat is represented by a non-linear feedback system and the cooling process represented by a linear system is proposed. Parameter adaptation and prediction algorithms for the model are modified to reduce the matrix dimensions, implemented in Matlab and applied to experimental data for validation. Apart from being highly accurate, the predictions comply with the desired figures of merit for the implementation in wireless sensor nodes, such as high robustness against quantization and environmental noise. Simulation results concludes that if the cargo emits organic heat, the proposed model is faster and more accurate than the linear models.

1 INTRODUCTION

Research has been done in the past to estimate the temperature profile inside refrigerated containers. Several options have been investigated: mathematical approaches as presented in (Shaik, 2007), K- ϵ models as proposed in (Rouaud, 2002), and several numerical models as reviewed in (Smale, 2006). With the exception of (Moureh, 2004), in which the effect of the pallets is considered; usually the focus is put on the cold air flow as the main factor governing the temperature pattern inside a container and the effects due the cargo presence is sub estimated.

To take into account the effect of the cargo in the temperature, in (Babazadeh, 2008) it is proposed the use of wireless sensor nodes (WSN) to measure the ambient parameters in the surroundings of a spatial point of interest and the use of system identification to estimate the parameters of a linear Multi-Input Single-Output (MISO) system. It concluded that in order to have a good estimation, it is necessary to have a high number of training samples and many inputs to the system.

In this paper an alternative Single-Input Single-Output (SISO) grey-box model is presented to predict the temperature inside the container under the presence of perishable goods with the aim of reducing the complexity and preserving the accuracy. The proposed model provides a meaningful description of the factors involved in the physical system including the effect of transporting living goods such as fruits and vegetables. The starting point is based on the physical relations; subsequently, a tuning parameter for the specific case of bananas is found by simulations.

2 MODEL OF THE SYSTEM

The factors affecting the temperature distribution inside a refrigerated container are illustrated in Figure 1. The cold air flows from bottom to top through the gratings in the floor and through the spaces between the pallets, and eventually it is drawn off the channel between the pallets and the container ceiling.

A naive representation of the container can be done by a SISO linear dynamic system in which the input is the air supply and the output is the spatial

point of interest. However, in reality it is only a simple model of the main contributor to the temperature pattern, the air flow. Several other factors affect the speed of the cooling down.

To improve the accuracy of the model, other contributors are considered as well: first is the heat, produced by respiration of living goods such as fruits and vegetables; second is the thermal loss, affecting the correct cooling of the goods; finally, unpredictable temperature variations due to highly changing external climatic conditions during transportation.

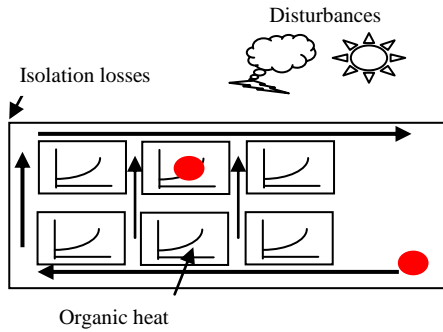


Figure 1: Factors affecting the temperature inside a refrigerated container.

The linear SISO black-box model representing the air flow is represented mathematically by a linear dynamic system H , which in the discrete domain is given by the Equation 1.

$$H(q^{-1}) = \frac{q^{-1}B(q^{-1})}{A(q^{-1})} \quad (1)$$

Where n_a and n_b are the orders of the system polynomials, $b_1 \dots b_{n_b}, a_1 \dots a_{n_a}$ are the polynomial coefficients, and q is the delay operator in discrete domain.

An attenuator, α , models the isolation losses of the air supply temperature and is modeled to affect the input of the dynamic system. The external climatic conditions are unknown in advance, therefore considered a statistical process. The output of the Moving Average (MA) process, which is in fact white noise (WN) filtered by the filter C represented in Equation 2 added to the output of the dynamic system, models them.

$$C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_{n_c} q^{-n_c} \quad (2)$$

To model the organic heat, it is necessary to use experimental data. Figure 2 (Mercantila, 1989) shows a family of curves for organic heat in the case of bananas. A proportional relationship between of the organic heat and the ripening state is observed.

Equation 3 represents the organic heat relation with respect to the temperature. P_{fruit} is the heat production in Watts, γ is a constant which is fixed for a certain type of fruit and ripening-state in $1/^\circ\text{C}$, T is the fruit temperature in $^\circ\text{C}$, and β is a scaling factor which depends of the amount of food and is given in kilograms.

$$P_{fruit} = \beta e^{\gamma T} \quad (3)$$

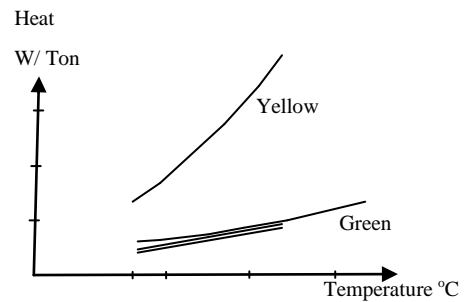


Figure 2: Heat Production of bananas.

Finally, the block diagram to represent the input-output relations of all the factors is built. It is shown in Figure 3. The air flow dynamics are represented as a feed-forward block as it is the most important contributor. The isolation losses affect the correct cooling of the goods before the dynamic system and the noise effect has an additive effect on the output.

The contribution of the organic heat depends on the cooling temperature inside the container. Simultaneously, it has a small additive effect in the input of the linear dynamic system as the air flows through the pallets and is slightly warmed. It is represented by a static exponential feedback. The resulting block diagram, in which a linear dynamic system has a non-linear feedback corresponds to a Feedback-Hammerstein (FH) configuration (Guo, 2004).

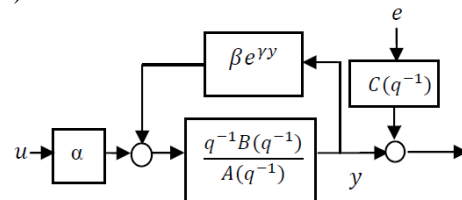


Figure 3: Model of the system.

3 PARAMETER ADAPTATION ALGORITHM

In (Guo, 2004) a Parameter Adaptation Algorithm (PAA) was developed to identify the parameter-set of a FH system. It uses an intermediate variable $\tilde{y}(t)$ and converts the non-linear system into a pseudo-linear one. Its principal advantage is that the conventional recursive matrix-based linear system identification algorithms as those presented in (Landau, 2005) can be applied to estimate the parameter matrix Θ . The recursive form of those algorithm is given by Equation 4. Where $\varepsilon(t)$ is the prediction error as described in Equation 5, $P(t+1)$ is an adaptation matrix to perform the minimization of ε using Recursive Least Squares method, and $\varphi(t)$ is the observation matrix that contains the input and the output data. $\lambda(t+1)$ in Equation 6 is the so called Forgetting Factor (FF).

$$\Theta(t+1) = \Theta(t) + (P(t+1)\varphi(t))^T \varepsilon(t) \quad (4)$$

$$\varepsilon(t) = y(t) - \Theta(t)^T \varphi(t-1) \quad (5)$$

$$P(t+1) = \frac{P(t) - P(t)\varphi\varphi^T \left(\frac{P(t)}{\varphi^T P \varphi + \lambda(t+1)} \right)}{\lambda(t+1)} \quad (6)$$

$$\lambda(t+1) = \lambda_o * \lambda(t) + 1 - \lambda_o \quad (7)$$

Guo considers the non-linearity as a polynomial of order l as shown in Equation 8; however, the dimensions of the matrices in the algorithm would be significantly too large to be applied in platforms where power consumption is an important figure of merit.

$$\eta(y(t)) = \sum_{k=0}^l \mu_k y^k(t) \quad (8)$$

To reduce the dimensions of the matrices, was proposed the use of the exponential Equation in Equation 3 instead. γ is to be determined and it remains constant, while β is a parameter to be identified as it depends on the amount of fruit being transported. The linear term of the Equation 8 needs to be extracted to be included in the polynomial $A^*(q^{-1})$ of the equivalent SISO pseudo-linear system. Expanding it into a Taylor series and rearranging, the summation of the non-linear coefficients of the exponential function can be calculated using Equation 9. The non-linear coefficients and an offset are on the left hand of the equation.

$$\sum_{k=2}^{\infty} \frac{(\gamma y(t))^k}{k!} + 1 = e^{\gamma y(t)} - \gamma y(t) \quad (9)$$

The equivalent pseudo-linear system for an exponential non-linearity is shown in Equation 10.

$$A^*(q^{-1})y(t) = b_1 \alpha u(t) + b_1 \beta e^{\gamma y(t)} - b_1 \beta y(t) + \frac{B^*(q^{-1})}{b_1} \tilde{y}(t) + C(q^{-1})e(t) \quad (10)$$

The resulting coefficients of the polynomials $A^*(q^{-1})$ and $B^*(q^{-1})$ are given by Equation 11 and 12.

$$a_k^* = a_k - (\beta \gamma) b_k \quad (11)$$

$$B^*(q^{-1}) = b_2 q^{-2} + \dots + b_{n_b} q^{-n_b} \quad (12)$$

And the intermediate variable is shown by Equation 13.

$$\tilde{y}(t) = b_1 [a u(t) + \beta (e^{\gamma y(t)} - \gamma y(t))] \quad (13)$$

The choice of the forgetting factor in the algorithm is often critical. In theory, it must be one that converges. On the other hand, if it is less than one the algorithm becomes more sensitive and the estimated parameter changes quickly making the convergence faster. A more complex solution is to allow it to vary with time, lower than one at the beginning but tending to one.

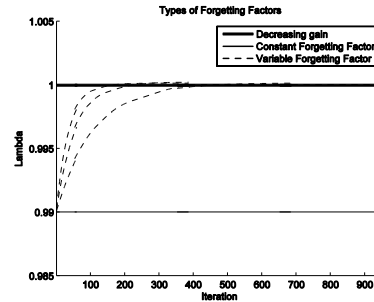


Figure 4: Types of forgetting factors.

Figure 4 illustrates three different types of FF. The first case is obtained by making λ_o , and $\lambda(t)$ in Equation 7 equal to one. It is called Decreasing Gain (DG). In the second case, the Constant Forgetting Factor (CFF) $\lambda(t)$ is set to a value smaller than one and λ_o set to one. Finally, the Variable Forgetting Factor (VFF) uses a value of λ_o smaller than one and recalculates $\lambda(t)$ for each iteration.

Table 1: Elements of the elements in the algorithm matrices.

Symbol	Arrangement of the elements into the matrices
$\varphi(t)$	$[-y(t) \cdots -y(t - n_a + 1), u(t - 1), (e^{\gamma y(t)} - \gamma y(t)), \tilde{y}(t - 1) \cdots \tilde{y}(t - n_b), \varepsilon_n(t) \cdots \varepsilon_n(t - n_c + 1)]$
$\Theta^T(t)$	$[a_1^* \cdots a_{n_a}^*, b_1 \alpha, \beta b_1, b_2/b_1 \cdots b_{n_b}/b_1, c_1 \cdots c_{n_c}]$
$\varphi_{pred}(t)$	$[-y_{pred}(t) \cdots -y_{pred}(t - n_a + 1), u(m), (e^{\gamma y_{pred}(t)} - \gamma y_{pred}(t)), \tilde{y}_{pred}(t - 1) \cdots \tilde{y}_{pred}(t - n_b)]$
$\Theta^T(m)$	$[a_1^* \cdots a_{n_a}^*, b_1 \alpha, \beta b_1, b_2/b_1 \cdots b_{n_b}/b_1]$

4 PREDICTION ALGORITHM

The predictions are made using the estimated parameters in the model. Figure 5 shows experimental data sets from a container transporting bananas. It can be observed how the air supply is kept constant after some days. For the prediction algorithm, $u(t)$ is set to the value of the last sampled input temperature of the parameter adaptation process. Similarly, the initial predicted output value is set to the last acquired value of the output. Equation 14 to 17 describes the prediction algorithm. m is the number of iterations used for the PAA.

$$u_{pred}(t) = u(m) \quad (14)$$

$$y_{pred}(m) = y(m) \quad (15)$$

$$y_{pred}(t) = \Theta^T(m) \varphi_{pred}(t-1) \quad (16)$$

$$\tilde{y}_{pred}(t) = b_1 [\alpha u(m) + \beta (e^{\gamma y_{pred}(t)} - \gamma y_{pred}(t))] \quad (17)$$

5 DETERMINATION OF γ

In considering a linear system, an exponential discrete time decaying system like the one presented in Figure 5 can be described as of the order of one with its unique pole on the real positive axis. The closer the pole to one the higher the delay of the system.

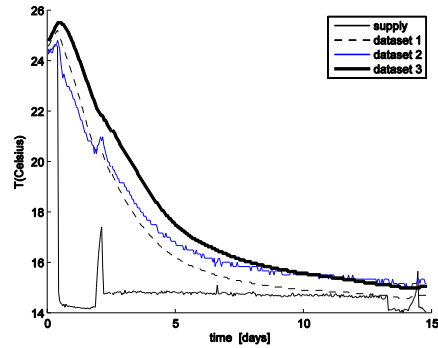
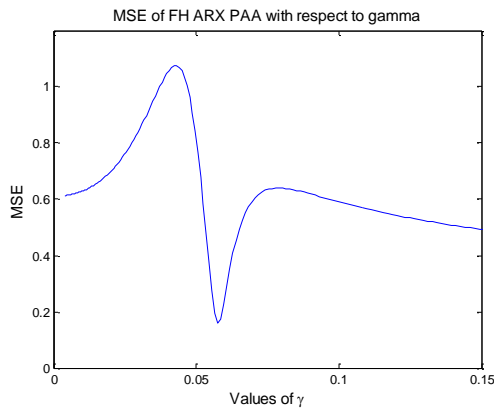


Figure 5: Banana data sets.

To find a trustworthy γ parameter that characterizes the respiration heat of bananas. The presented Feedback-Hammerstein model of linear order one and the FH parameter adaptation and prediction algorithms are run using given experimental data sets. The Mean Squared Error (MSE) of the prediction over n samples, equivalent to fifteen days, is stored for several values of γ and fixed number of training days. If the stored values of the MSE are plotted, the local minimums are determined by the observation of the MSE vs. γ curves. In Figure 6, it can be seen that in the above mentioned plot for five days of training and for the data set 1, a local minimum exists for a value γ of 0.0587.

$$MSE = \frac{1}{n} \sum_{t=m}^n (y_{real}(t) - y_{pred}(t))^2 \quad (18)$$


 Figure 6: Prediction accuracy vs. γ .

6 RESULTS

For validation of the model and algorithms several figures of merit are considered. The accuracy and the speed of convergence are of paramount importance; however, quantization and noise robustness are also highly desirable for implementation in a WSN. Only the linear orders of one and two are considered to avoid computation of complex conjugate poles that would characterize oscillations.

To observe the speed of convergence and the accuracy of the predictions with respect to the number of training days, parameter estimation and a prediction in Matrix form are done (See Table 1) for a fixed number of training days. Subsequently, *MSE vs. Training days* graphs are plotted. Assuming a quantization level of 0.2°C , a Matlab script was written to assign the nearest value of the quantization grid to the input and the output datasets. The results of the predictions using the quantized datasets are overlapped with the results of non-quantized.

Similarly, to determine the noise robustness, MSE versus the signal to noise ratio (SNR) is plotted. Several noise levels of white noise were added to the output of the data set 1, and the resulting signals were applied to PAA and prediction algorithms with fixed number of training days.

$$\text{SNR(dB)}=10\log\left(\frac{P_{\text{signal}}}{P_{\text{noise}}}\right) \quad (19)$$

Simulations were done for two types of data sets. First, the experimental data of bananas were used to include the presence of organic heat. Secondly, the

data sets corresponding to a cheese experiment, which does not present organic heat, were considered. A summary of all simulation results is presented on Table 2.

6.1 FH vs. Linear Models in the Presence of Organic Heat

From the simulations it is observed in Figures 7 and 8 that if linear methods are applied to the banana datasets, the accuracy of the results for different sensor positions of are not sufficient. Quantization robustness is improved with the linear order of one and the speed of convergence is better using CFF. In the best of cases acceptable prediction accuracy can only be achieved after more than five days of training.

It is also observed in Figures 8 and 9 that FH identification algorithms are the best to achieve fast convergence speeds. In the best cases, less than 3 days of training is sufficient to achieve good predictions. However, the plots are made for the data from three days onwards to avoid the visualization of the effects in MSE due to the set point variations in the reefer supply temperature. Linear system orders of one are in all cases better than order of two, both in the speed of convergence and the quantization robustness. Decreasing Gain must be optimal to preserve the accuracy and the quantization robustness.

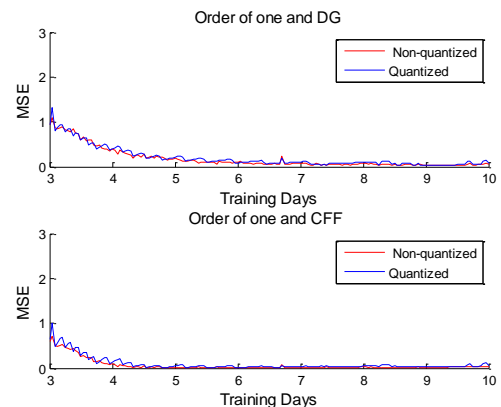


Figure 7: ARX of order one in the presence of organic heat.

Concerning the noise models, results of the simulation of Feedback-Hammerstein with MA process are worse than when modeled as white noise (WN). It affects the accuracy and the quantization robustness

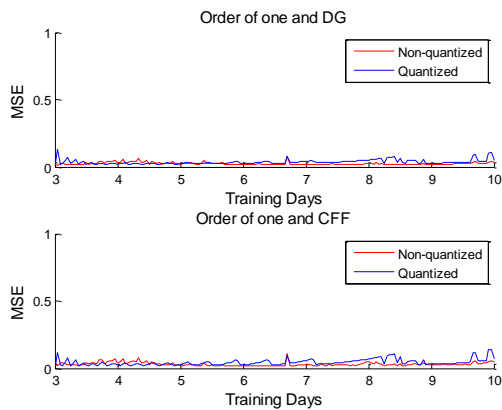


Figure 8: FH of order one in the presence of organic heat.

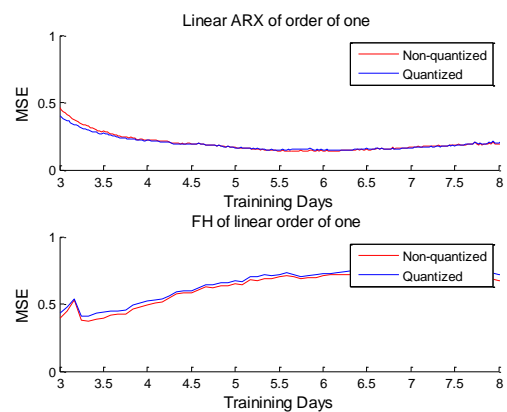


Figure 10: Comparison of FH and linear methods in the absence of organic heat.

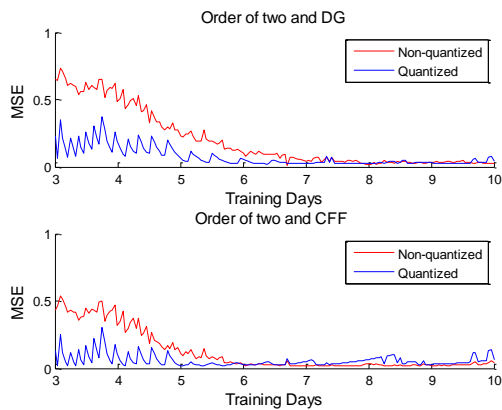


Figure 9: FH of order two in the presence of organic heat.

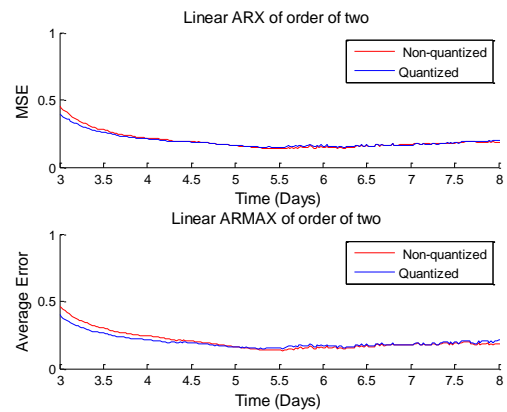


Figure 11: Comparison of linear methods with MA and WN models.

6.2 FH vs. Linear Models in the Absence of Organic Heat

In the case of cheese data set, the linear methods accuracy results are better than that of the Feedback-Hammerstein as can be observed in Figure 10. Modeling noise as white gives better quantization robustness than modeling it as MA process.

The use of forgetting factors does not have a big impact in the results of ARX predictions; however, Constant Forgetting Factor is slightly better for ARMAX predictions. Linear orders do not affect the simulated predictions, but an order of two is selected because it can model more accurately if the behavior of the system is not purely decaying.

6.3 Noise Robustness

The noise was added to validate FH and linear models; also for both of them the accuracy is compared with and without the MA model. Maximum Signal-to-Noise Ratio to obtain a good prediction is observed to be around 43 dB for all of them with the exception of ARX which has a maximum value of 47 Decibels as shown in Table 2.

6.4 Prediction Improvement

The described approach was originally developed based on an experiment in 2008 with records for 3 sensors (data set A). Two new data sets with 16 sensors each, which were recorded in 2009 (Jedermann, 2010) in two separate containers (data set B and C), were used to cross validate the approach.

Table 2: Summary of simulation results.

	Accuracy		Number of matrix elements	Convergence speed	Quantization Robustness	Critical SNR	Estimation for linear dataset
	Best Forgetting Factor	Best Linear order					
ARX	CFF	2	3	Bad	Good	47dB	Good
ARMAX	CFF	2	$3 + n_c$	Bad	Bad	43 dB	
FH and WN model	DG	1	3	Good	Good	43 dB	Bad
FH and MA model	DG	1	$3 + n_c$	Good	Bad	43 dB	

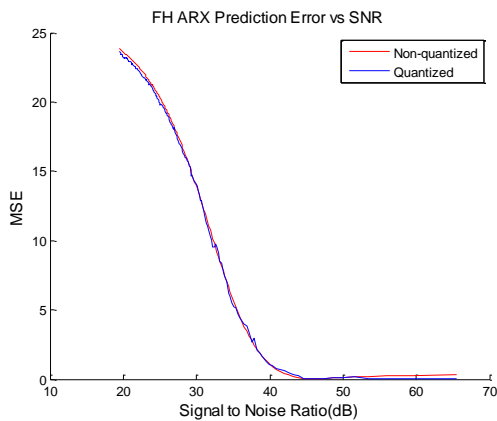


Figure 12: Noise Robustness for FH method

FH algorithm of linear order of one was applied to all data sets; neither quantization nor forgetting factor is used. For the initial parameter settings, the pole and zero of the feed-forward linear system was set to 0.9 and 0.0; β was set to 2.

The previously obtained value of γ equal to 0.0587 is used to predict the temperature inside the containers for many spatial positions. The results are compared to the predictions for the datasets shown in Figure 5 and resumed in Table 3. A good average is observed for the three containers; however, in some positions the predictions are not as accurate as is observed in the Maximum column.

 Table 3: MSE prediction results for a unique value of γ .

Container/Result	Maximum	Minimum	Average
Data set A	0.1893	0.0173	0.0778
Data set B	1.4558	0.0550	0.4130
Data set C	0.8888	0.0101	0.2798

A second approach is to select γ according to the position of the pallets inside the container. The method to find γ , described previously, is applied to all the new container datasets.

It is observed that an improvement in the accuracy of the predictions can be made if two different values of γ are selected: one for pallets close to the door-end, and one for pallets close to the reefer supply. In Table 4 it is resumed the prediction results if values of 0.0525 and 0.055 are set respectively.

 Table 4: MSE prediction results for values of γ according to the position inside the container.

Container/Result	Maximum	Minimum	Average
Data set A	0.1893	0.0173	0.0778
Data set B	0.4767	0.0279	0.0946
Data set C	0.5747	0.0201	0.1743

7 CONCLUSIONS

A model to represent the factors affecting the temperature inside a refrigerated container transporting perishable goods was proposed. It models the effect of organic heat using a static non-linear feedback system, the refrigeration by a linear dynamic feed-forward system, and the disturbances by stochastic processes. This complex model can provide an accurate description of the factors involved in the physical system.

The selected identification method was adapted specifically to reduce the dimensions of the matrices. The non-linear exponential function is used instead of a polynomial to preserve the simplicity of the parameter adaptation and the prediction algorithms. The disadvantage of the

simplification is that depending on the kind of fruits to be transported, it is required to tune the algorithm by a correct selection of γ which has to be known in advance. An improvement can be observed in the accuracy of the predictions if γ is set according to the position of the pallets inside the container.

From the simulation results it is concluded that the FH identification algorithm is efficient when the cargo emits organic heat. The method of FH of order 1 is optimal to achieve all figures of merit. It makes accurate predictions only after three days of training and maintains low dimensions of matrices.

However, if the linear method is applied to the banana datasets, a comparable accuracy can only be achieved after more than five days of training. Also, it is concluded that when the goods to transport are free of organic heat, such as in the case of cheese, it is preferable to use a linear system instead.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to Prof. Rainer Laur, Dirk Hentschel, Mehrdad Babazadeh, and Chanaka Lloyd for all their help.

This research was supported by the German Research Foundation (DFG) as part of the Collaborative Research Centre 637 "Autonomous Cooperating Logistic Processes". Further project information can be found at www.intelligentcontainer.com.

REFERENCES

Babazadeh, M., Kreowski, H.-J., Lang, W., 2008. Selective Predictors of Environmental Parameters in Wireless Sensor Networks. In International Journal of Mathematical Models and Methods in Applied Sciences Vol.2, pages 355-363.

Guo, F., 2004. A New Identification Method for Wiener and Hammerstein System. Ph.D. Thesis, Institut für Angewandte Informatik Forschungszentrum Karlsruhe, Karlsruhe.

Jedermann, R., Becker, M., Görg, C., Lang, W., 2010. Field Test of the Intelligent Container, In European Conference on Wireless Sensor Networks EWSN2010, 16-19 February 2010 in Coimbra, Portugal.

Landau, I.D., Zito G., 2006. Digital Control Systems: Design, identification and implementation. Springer. London, 1st edition.

Mercantila. 1989. Guide to food transport: fruit and vegetables. Mercantila Publ. Copenhagen. Moureh J., Flick, D., 2004. Airflow pattern and temperature distribution in a typical refrigerated truck configuration loaded with pallet. In International Journal of Refrigeration, V. 27, Issue 5, pages 464-474.

Palafox J., 2009. Prediction of temperature in the transport of perishable goods based in On-line System Identification. M.Sc. Thesis, University of Bremen. Bremen.

Rouaud, O., Havet, M., 2002. Computation of the airflow in a pilot scale clean room using K- ϵ turbulence models, In International Journal of Refrigeration, Vol. 25, Issue 3, pages 351-361.

Shaikh, N. I., Prabhu, V., 2007. Mathematical modeling and simulation of cryogenic tunnel freezers. In Journal of Food Engineering, Vol. 80, Issue2, pages 701-710.

Smale, N. J., Moureh, J., Cortella, G., 2007. A review of numerical models of airflow in refrigerated food applications. In International Journal of Refrigeration, Vol. 29, Issue 6, pages 911-930.

LIST OF ABBREVIATIONS

ARMAX	Auto Regressive Moving Average with External input.
ARX	Auto Regressive with External input.
CFF	Constant Forgetting Factor
DG	Decreasing Gain
FF	Forgetting Factor
FH	Feedback Hammerstein
MA	Moving Average
MSE	Mean Squared Error
PAA	Parameter Adaptation
WN	Algorithm White Noise
WSN	Wireless Sensor Node

DYNAMIC CONTROL OF MOBILE AD-HOC NETWORKS

Network Protocol Parameter Adaptation using Organic Network Control

Sven Tomforde, Björn Hurling and Jörg Hähner

Institute of Systems Engineering, Leibniz Universität Hannover, Appelstr. 4, 30167 Hannover, Germany

{tomforde, hurling, haehner}@sra.uni-hannover.de

Keywords: Organic computing, Self-optimising data communication protocols, Framework for protocol control, Mobile ad-hoc networks.

Abstract: Data communication protocols show an increasing complexity in terms of variable configurations – especially if their target execution area is highly dynamic. One domain that represents these characteristics are mobile ad-hoc networks (MANets). Since nodes are moving, the situation surrounding a particular node is steadily changing. This provides the opportunity to significantly increase the system’s performance by continuously adapting the protocol. This paper demonstrates the benefit of such an adaptation using the Organic Network Control (ONC) system. Based upon a brief overview of ONC, the adjustment of the framework to enable the control of MANets is described, followed by a simulation-based evaluation using an exemplary broadcast protocol.

1 INTRODUCTION

As the number of interconnected devices and the corresponding transfer load of data communication networks is steadily increasing, networks reach their limits. This leads to the insight that the currently used techniques (e. g. protocols and infrastructure) will not be able to cope with the demand in the near future (Handley, 2006). Based upon this assumption, researchers develop new concepts (e. g. for the Internet (Siekkinen et al., 2007)). To cope with the problem, two approaches are possible: Develop new protocols (with higher complexity in terms of more parameters) and increase the quality of service or dynamically adapt existing protocols to changing environments, which is done by the Organic Network Control (ONC) system (introduced in (Tomforde et al., 2009a; Tomforde et al., 2009b)).

The ONC system is based on the principles of Organic Computing (Schmeck, 2005) which is a recent research area focusing self-organisation to deal with complex problems. Autonomous entities are acting without strict central control and achieve global goals although their decisions are mainly based on local knowledge. The authors assume that due to the complexity of the particular tasks not all situations can be foreseen during the development process of the system. Therefore, the system must be adaptive and equipped with learning capabilities, which leads

to the ability to learn new actions and strategies for previously unknown situations. The self-control of network entities is also part of the focus of Autonomic Computing (Kephart and Chess, 2003).

A demanding challenge for the ONC system is the control and adaptation of mobile ad-hoc network (MANet) protocols as they are processed in highly dynamic environments. The possible movement of nodes leads to a continuous change of the situation: Neighbours are getting out of reach or joining the sending distance. This does not only lead to complex problems of how to configure the protocol, it also offers high potential for an improvement of the system performance. Within this paper, we explain how the ONC system is applied to a MANet broadcast protocol and how the overall performance of the MANet system can be increased using ONC.

This paper demonstrates the application of the ONC system to MANet-based broadcast protocols by dynamically adapting network protocol parameters (e. g. values for timeouts, maximum number of re-transmissions, number of open connections, etc.). Section 2 describes the related work and gives a summary of approaches to adapting network protocols dynamically to changing environments, followed by an overview of the ONC system in Section 3. In Section 4, we explain how the ONC system can be applied to a new protocol and what is actually done to enable MANet protocol control using ONC. Af-

terwards, Section 5 demonstrates the benefit of using ONC by explaining and analysing the achieved results. Finally, Section 6 summarises the presented system and names further research to be done.

2 RELATED WORK

Autonomic adaptation of networks is part of the focus of Autonomic Computing (Kephart and Chess, 2003). Researchers have considered the problem to adapt network protocols for many years: from off-line optimisation when presenting a new protocol to adapting protocol configurations during runtime. For the off-line optimisation, several examples can be found in literature, but none aims at providing a generalised approach for more than one specific protocol (see, e.g., (Montana and Redi, 2005; Sözer et al., 2000; Turgut et al., 2002)). Due to time and computational restrictions, on-line adaptation is a more complex task compared to the off-line part. Besides the ONC approach, different directions of research are known to cope with the problem: adaptive protocols, composition of protocol stacks, or centralised solutions to adapt protocol configurations.

The most obvious way of dealing with the problem is to develop adaptive protocols which are able to handle dynamic environments. One example has been presented in (Whiteson and Stone, 2004). They introduced an on-line learning mechanism to increase the performance of a routing protocol. Based on the Q-routing techniques presented in (Boyan and Littman, 1994), they learn the best routes by receiving immediate answers of the next hop. Another example has been introduced in (Huang et al., 2009). They present an adaptive medium access control (MAC) protocol framework. Since the radio node density and service requirements can vary widely over time, they defined the need of an adaptation to changing environments and needs. Their protocol prototype can switch between CSMA and TDMA within a radio platform scenario. Both approaches rely on the existence of a protocol extension covering the learning/adaptation information. In contrast to the ONC system, they are system-specific solutions and cannot be applied to other protocols.

Since developing new protocols for all possible adaptation and learning processes is not feasible, a research field called *protocol stack composition* emerged covering the upcoming tasks by exchanging protocols and stacks dynamically (Rosa et al., 2007). In contrast to the ONC system which keeps the existing and currently used techniques and optimises their behaviour, this field of research re-combines the pro-

ocols. Although the target deviates from the ONC approach (the protocol stack exchange has impact on all involved systems and can hardly be done locally), the approach has some similarities. The most important representatives are *Appia* (Miranda et al., 2001), *Cactus* (Hiltunen et al., 2000), *Ensemble* (van Renesse et al., 1998), and *Horus* (van Renesse et al., 1996). Additionally, the recent work done by *Mena et al.* (Mena et al., 2003) has to be named. Besides the locality aspect, some characteristics of the approach separate it from the requirements of the ONC framework: the protocols and their configurations have to be known in advance and further extensions with new behavioural repertoire are not possible.

The approach presented in (Schöler and Müller-Schloer, 2004) is also dealing with a kind of protocol composition, but is already a bit more focused on the techniques used within the ONC system. The authors describe their adaptive monitoring architecture for protocol stack configuration and demonstrate the integration in the Observer/Controller pattern of Organic Computing. The learning part is covered by a Fuzzy Learning Classifier System (Casillas et al., 2004). Due to the usage of the same architectural pattern (Observer/Controller), the approach has some similarities with the ONC system. Unlike the ONC framework, the approach is built again without offering the opportunity of handling different protocols and extending the set of solutions on demand.

On-line adaptation of protocols itself has been focused by researchers before. (Sudame and Badrinath, 2001) presents a first TCP- and UDP-based study and defined the need of dynamic adaptation, but detailed examination and a demonstration of the re-usability for other protocols is still missing. Currently, there exist only two approaches covering a similar target as ONC: the systems introduced by *Ye et al.* and by *Georganopoulos and Lewis*. The former one (Ye and Kalyanaraman, 2001) introduces an adaptive random search algorithm which tries to combine the stochastic advantages of pure random search algorithms with threshold-based knowledge. Their approach is based on the initial system as presented in (Ye et al., 2001). In contrast to our approach, *Ye et al.* propose a centralised system that tackles the optimisation task for each node. To allow for such a division of work between a central server and the particular network nodes, problems like e.g. bandwidth usage, single point of failure, or local knowledge accessible from server-side have to be covered.

The second system has been presented in 2007 (Georganopoulos and Lewis, 2007) and introduces a dynamic optimisation framework for the reconfiguration of network protocols at all layers of the protocol

stack. In order to optimise the performance of the system depending on given goals, different entities can be adjusted (applications, protocols, etc.) or replaced. Again, the system relies mainly on a centralised element being responsible for the optimisation tasks. The focus of the initial paper has been set on cross-layer optimisation for the protocol stack, but less on considering environmental conditions. Hence, the authors demonstrated the performance of the solution by applying it to two different layers of the protocol stack: the link and the network layer. A detailed proof of the approach and insights on the currently vague blackbox *dynamic optimisation engine* are still missing, consequently a suitability of the approach cannot be estimated – although some criteria (centralised element, low re-usability of existing protocols, etc.) are contradicted for the ONC requirements.

3 THE ORGANIC NETWORK CONTROL SYSTEM

The Organic Network Control (ONC) system has been introduced in (Tomforde et al., 2009a; Tomforde et al., 2009b). The system’s architecture is founded on the generic Observer/Controller approach as presented by *Richter et al.* in (Richter et al., 2006) and is organised using three consecutive layers, see Figure 1.

Layer 0 encapsulates an existing network protocol instance, e.g. a broadcast algorithm for mobile ad-hoc networks (MANets) or a Peer-to-Peer (P2P) protocol. In terms of Organic Computing, this controlled network protocol instance is the “System under Observation and Control” (SuOC). The ONC system aims at providing a basic solution to control existing protocols dynamically without the need of knowing internals of the particular protocol or interfering with the protocol logic. However, it is required that the parameters of the protocol can be altered by the ONC system. Additionally, the current status of the protocol instance and the environment it acts in have to be observable and accessible locally. For MANets, one of the most important factors describing the current status of the protocol instance’s environment is the neighbourhood of other nodes. Besides this observable environment, a performance measure (also called fitness or evaluation function) quantifying good and bad performance has to be provided in order to evaluate the current performance of the protocol.

Layer 1 of the ONC architecture aims at adapting the SuOC dynamically to changes in the environment. It therefore consists of two basic components: an Observer and a Controller containing a machine learning

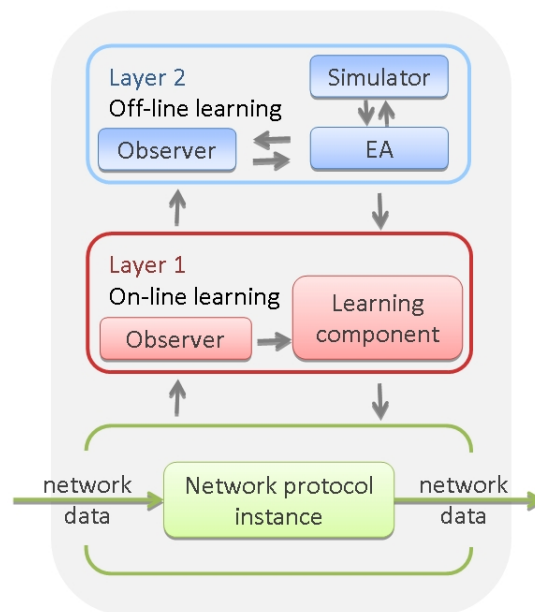


Figure 1: Architecture of the ONC system.

component (which is currently realised as a Learning Classifier System (LCS)). The Observer is responsible for locally collecting status information about the network protocol instance and its settings. Additionally, it aggregates the observed figures and augments them with optional further knowledge (e.g. prediction values, historic knowledge, etc.) and builds a vector describing the current situation at the node. This situation vector then serves as input to the Controller which has to fulfill two tasks: evaluate the system’s performance within the last evaluation cycle and decide about the next action to be taken. The main component of the Controller is a LCS which is realised as an adapted variant of Wilson’s XCS (Wilson, 1995). The LCS is responsible for choosing the next action – based on the situation vector. The result of the selection process is an action (a configuration of parameter sets for the controlled protocol) and a prediction value of how the system will perform after applying the action. To evaluate the system’s performance, the Controller compares the system’s performance measure (the fitness function) with the last prediction and calculates the reward for the LCS to enable the automated learning. In case the LCS does not contain a matching parameter set, new classifiers need to be created. In contrast to the original LCS algorithm, however, the ONC architecture does not allow new classifiers (pairs of situation/conditions and parameters/actions) to be created randomly by Genetic Algorithms. Instead, control is transferred to Layer 2 of the ONC architecture.

Layer 2 of the ONC system is again designed us-

ing the Observer/Controller pattern: The Observer monitors the Layer 1 component and realises the need of a new classifier, it therefore receives the situation vector. The Controller part contains two basic components: a simulator and an Evolutionary Algorithm (EA). The Controller creates an appropriate simulation scenario from the situation vector and triggers the EA to repeatedly evolve a number of parameter sets for the network protocol. These parameter sets are evaluated in the simulator. This bears the advantage that newly created parameter sets are not directly used in the live system, as this can cause the system to perform badly or even malfunction. Only those parameter sets that qualify in the simulator of Layer 2 are passed back to Layer 1 and may then be applied in the real world. Therefore, Layer 2 allows for a kind of *sandbox*-learning without the risk of applying arbitrary parameter sets to the live system.

The ONC approach as described before provides a black-box solution to control different types of network protocols. In order to integrate a new protocol into the ONC architecture and consequently enable ONC controlling the protocol, an engineer has to fulfill three major tasks: Specify the performance metric, describe the situation (what are the dynamic factors defining the need of an adaptation, e.g. available neighbours and their positions in MANets) accompanied by a distance function between two situations, and provide a simulation model to enable the simulation-based optimisation process of Layer 2. Within the following Section, we describe how the ONC system is applied to MANet protocols.

4 DYNAMIC CONTROL OF MOBILE AD-HOC NETWORKS

This Section describes how ONC is adapted to allow for the control of MANet protocols - based on the tasks named within the last Section. In order to keep the same organisation as before, this Section again distinguishes between the three layers and describes what has to be done on each layer.

The focus of Layer 0 is to integrate a new protocol into the framework. Therefore, the engineer has to describe its observation and control process leading to the need of two interfaces: one for accessing the protocol parameters and one for collecting information about the local status of the system. The former interface enables the framework to adapt the behaviour of the protocol which means the parameter settings can be adapted at runtime. In the latter interface, the engineer has to define what is relevant and influences the protocol's performance - we call this the *situa-*

tion the system is in. In a MANet environment, the most important factor influencing the protocol's performance is the distribution of other nodes within its sending and sensing range. Therefore, a sector-based approach as depicted in Figure 2 has been developed. The radius of the outer circle is equal to the sensing distance (*sensDist*) of the node, as this is the most remote point where messages of this node can interfere with other ones. Typically, the transmission range for Wireless-LAN based MANets is about 250 meter (half of the sensing distance). The radii of the inner circles have been chosen empirically.

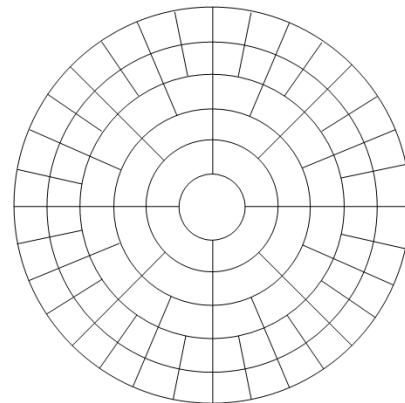


Figure 2: Environment representation.

As nodes within the first circle are really close (50m), their exact position does not matter. The second circle (125m) has been partitioned into 4 sectors, the third circle (200m) into 8 sectors, and the fourth circle (250m, maximum transmission range) into 16 sectors. The next two circles (375m and 500m) are representing the area within sensing range - with both circles divided into 32 sectors each. We assume that a node is able to determine the current positions of its neighbours in sensing range relative to its own position (e.g. based on GPS, see (Pahlavan and Krishnamurthy, 2001)). Additionally, the node's direction of movement is needed since it has high influence on the best parameter set (e.g. moving towards/away from a set of nodes influences the delay). Due to the sector-based approach, situations are generalised which is necessary to avoid evolving a rule for each situation.

The Layer 1 component is responsible for the adaptation process and for increasing the system's performance by learning. Again, two aspects have to be considered: a learning feedback and a measurement to compare different situations. To enable the learning feedback, a fitness or evaluation function is needed. Several fitness functions have been proposed for MANet protocols. Since our current focus is set on MANet-based broadcast algorithms, the standard functions are *Packet Delivery Ratio* and *Packet La-*

tency – both cannot be measured locally at each node. Based upon the locally available information, the target is to reduce the number of forwarded broadcasts and assure the delivery of the broadcast to each node at the same time. Therefore, we introduce the following formula:

$$Fit(x) = \frac{\#RecMess}{\#FwMess}$$

The variable x stands for the currently observed network protocol instance. Since a new parameter set has to be applied for a minimum duration to show its performance, we use *evaluation cycles* defining discrete time slots. The duration of the cycle depends on how dynamic an environment is: The faster it changes, the shorter is the cycle to be chosen. For the last evaluation cycle, the function takes into account the sum of all messages being forwarded by all of the neighbours and the node to be evaluated within the last evaluation cycle ($\#FwMess$), and the sum of all messages being received by them ($\#RecMess$).

The second aspect on Layer 1 is the comparison of situations: we need to quantify the distance of two situations. The target is that more similar situations will receive a low distance value and those having low similarity will receive a high distance value. Based upon the sector-based situation description as introduced before, a measure for the similarity of two entities (A, B) can be defined. To be able to determine the distance, the possible influence of rotation and reflection are deducted initially. Afterwards, the formula for the distance (δ) can be defined with $r \in RADII$ and $s \in SECTORS$ as follows:

$$\delta(A, B) = \sum_r \sum_s (A_{r,s} - B_{r,s})^2 / r.distance$$

The function *r.distance* defines the radius size as introduced before (50m, 125m, ...). $A_{r,s}$ gives the number of neighbours within the sector s of radius r for the situation description A . This means that the importance of a node's neighbour decreases if it is situated within an outer radius.

Finally, Layer 2 has to be able to build adequate simulation scenarios out of the information obtained by Layer 1. In ONC, we use the standard network simulation tool *NS-2* (Fall, 1999), but this can easily be exchanged by other solutions. The network simulator *NS-2* has a large set of integrated or available standard protocols, but for recently developed or proprietary protocols a simulation model probably does not exist. The engineer has to provide a realistic model (as it is also used during the protocol development) which can be adapted to the observed situation

by generating an appropriate scenario. The adaptation of the scenario is done using the configuration interface by considering the observed situation of the node, e.g., a randomised instance of the sector-model is created defining the distribution of the neighbouring nodes and the movement direction of the node is transmitted to *NS-2* using the same coordinate system as for the observed system. After finishing the previously described tasks, ONC is able to control MANet-based protocols. The benefit of the dynamic control is demonstrated in the next Section.

5 EVALUATION

Based upon the previously described adaptations, ONC is able to control MANets. Within this Section, the results of the evaluation are presented. Since the adaptation of protocols is organised locally - but has high influence on the network-level, both aspects are taken into consideration. Therefore, we start with the experimental setup, followed by a short introduction of the analysed protocol, and conclude with the achieved results for the local and the network-wide view.

5.1 Experimental Setup

The ONC framework is implemented in JAVA. The moving agents communicating via the MANet protocol are simulated using the Multi-Agent Simulation Toolkit *MASON* (Luke et al., 2004), with each agent's protocol instance representing a SuOC of the architecture as depicted in Figure 1. The respective Layer 1 Controller is an adapted Learning Classifier System as described in (Tomforde et al., 2009b). At Layer 2, the standard network simulation tool *NS-2* (Fall, 1999) is used to evolve new parameter sets in combination with a standard Genetic Algorithm (population size: 15, new children per iteration: 7, mutation rate: 0.2 per child, all children via crossover with fitness-based selection of parents). We use two different simulation tools in order to avoid having exactly the same conditions while optimising rules, since a complete copy of the current situation observed in the real environment within the simulator is not realistic. 100 agents have been created and applied to the simulated area, which has dimensions of 1000 x 1000 cells (corresponds to 1000 x 1000 meters). The agents move according to a random-waypoint-model. The Physical/Mac layer is an IEEE 802.11 in ad-hoc mode at 2 Mbps.

To demonstrate the performance of ONC controlling MANets, we choose the *Reliable Broadcast Protocol* (R-BCast) as introduced in (Kunz, 2003), since

this protocol is representative for the research field of reliable broadcast protocols in MANets. In order to achieve reliability and increase the packet delivery ratio compared to other protocols, additional effort is made by equipping the nodes with extra buffers. These round-robin based buffers are used to store the last p unique packets the particular nodes received. In contrast to other protocols, the R-BCast protocol has significantly more variable parameters and consequently the task to control the protocol is more complex, but it also offers a higher potential benefit due to a dynamic adaptation. The parameters being subject to ONC control actions are: **Delay** (Maximum deceleration time between receiving and forwarding of a message), **Allowed Hello-loss** (Maximum number of Hello-messages, which may be lost until a node is assumed to be out of transmission range), **HelloInterval** (Interval between two Hello-messages), δ **HelloInterval** (Randomises Hello-Interval), **Packet count** (Number of the last x stored NACK messages), and **Minimum difference** (Minimum difference between NACK messages). Details on the parameters and the protocol can be found in (Kunz, 2003).

5.2 Experimental Results

In order to analyse the performance of the ONC system, the simulation is repeated for two cases under the same restrictions and using the same seeds for the randomised values: a) all nodes are uncontrolled (no ONC system) and use the manually optimised standard configuration of the protocol, and b) all nodes have an own instance of the ONC system to control their protocol configuration. All values presented in the remainder of this Section are averaged values received from three runs, where each run has a duration of 10,000 simulated seconds. During one run of the scenario, 17,400 BCast-messages have been simulated. The learning component has been trained using 10 complete runs with different seeds – leading to completely different movements of the nodes and along with these to different situations.

The performance measurement relies on the fitness function as described for the local feedback mechanism of the learning component (Section 4). Since the evaluation takes both views into account (local and network-wide view), the x in $Fit(x)$ refers to different systems: a) In the local view x stands for the local network protocol instance of the node and b) on network-level obtained for reference, x represents the set of protocol instances within the network and gives an averaged value for all instances.

Figure 3 plots the system's performance considering only one node. The X-axis describes the sim-

ulation time (in simulated seconds) and the Y-axis the measured fitness value. In principle, all simulated nodes show a comparable behaviour; this specific node has been explicitly chosen to demonstrate the typical differences between an ONC-controlled and an uncontrolled node. During the simulation, the node gets separated from the rest of the network (no other nodes within *sending* distance) between simulation seconds 7,350 and 7,700. Within this interval, the fitness is 0 for both cases. But especially these situations demonstrate the benefit of ONC control: The delays have been lengthened so that the node receives more *old* messages when it arrives back in sending distance of another node resulting in a quicker recovery of the ONC-controlled system.

Another observation that can be made considering Figure 3 is the impact of the learning module. To be able to learn, it has to be allowed to try different rules and not to use always the best matching one. E.g. at simulation second 1,800, the learning component tried a rule that results in a performance slightly worse than the standard protocol configuration. These small drawbacks have to be taken into account to achieve an improvement for the system. Averaged over the complete simulation time (10,000 simulated seconds), the performance of the protocol instance has been enhanced in terms of the fitness function from 0.8270 (all nodes perform the standard protocol without any adaptation) to 0.8991 which is an increase of 8.71%.

Figure 4 depicts the averaged performance of the network protocol instances on network-level. The averaging leads to the effect that separation of single nodes influences the performance only slightly. Nevertheless, Figure 4 shows two drops (simulation seconds 3,850 to 4,200 and 7,450 to 7,550). The first drop can be explained by a split of the set of nodes – about 30 nodes are not within sending distance of the rest. Here, two different networks have been established. Within the second drop, again, a larger group (18 nodes) has been separated from the rest of the network. Despite these separation effects, the performance of the system has been increased. When all nodes perform just the standard protocol configuration without any adaptation, the resulting averaged fitness is 0.8760. The same simulation with additional ONC control for all nodes leads to an averaged fitness value of 0.9456 which is an increase of 7.94%. Both aspects of the fitness function are responsible for the increase: The number of forwarded messages has been decreased slightly, whereas the number of received messages has been increased more significantly.

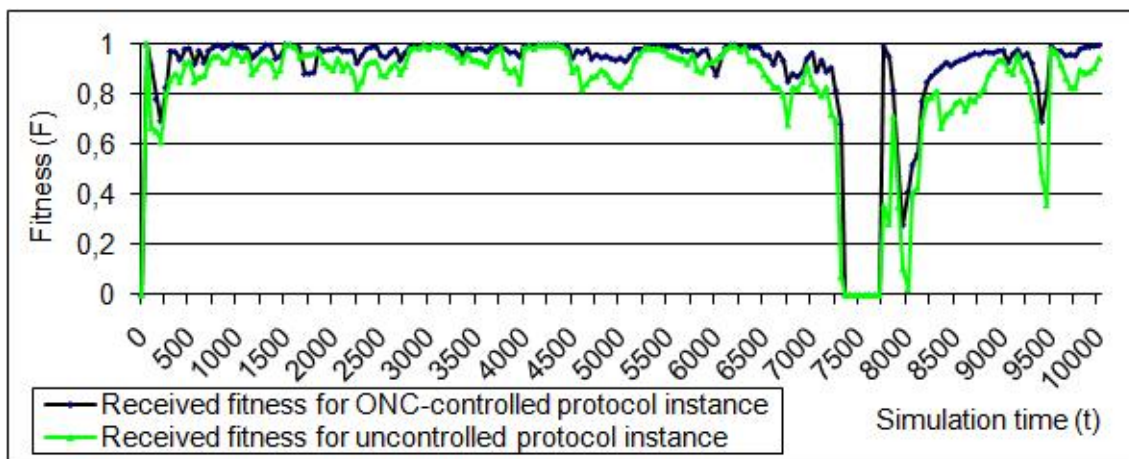


Figure 3: Comparison of simulation results for one node: uncontrolled and ONC-controlled.

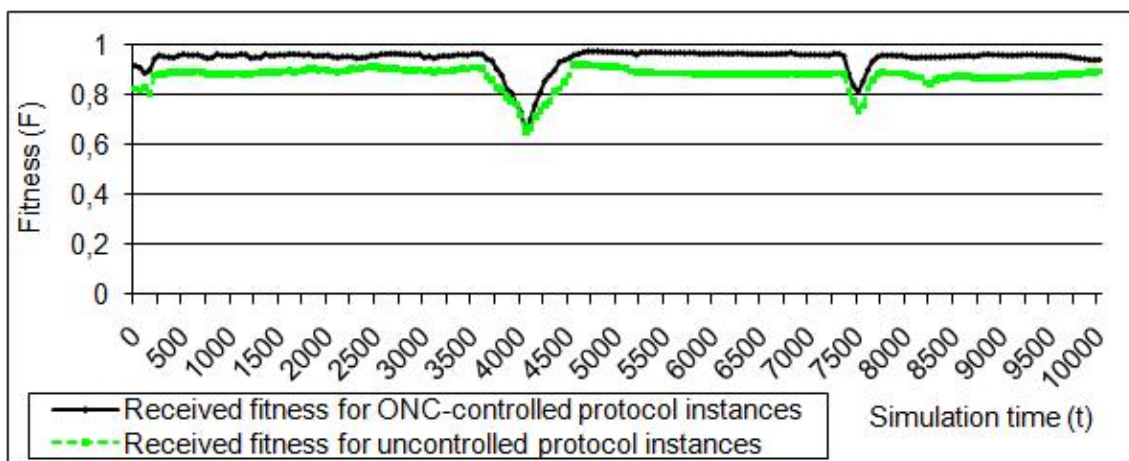


Figure 4: Network-wide comparison of simulation results: uncontrolled and ONC-controlled.

6 CONCLUSIONS

This paper described how the Organic Network Control (ONC) system is applied to mobile ad-hoc networks (MANets). Therefore, a brief overview of the system has been given, followed by a description of the adjustments needed to apply ONC to this type of protocols. Afterwards, the benefit of ONC control has been demonstrated using an exemplary MANet-broadcast protocol.

Current and future work of the system focuses on three main aspects: Increase the performance of the control mechanism by improving its behaviour, decrease the effort spent on evolving new rules by using a simple collaboration mechanism, and demonstrate the generic character of the approach by applying it to other types of protocols that are performed in highly dynamic environments (e.g. sensor networks).

REFERENCES

- Boyan, J. A. and Littman, M. L. (1994). Packet routing in dynamically changing networks: A reinforcement learning approach. In *Advances in Neural Information Processing Systems 6*, pages 671–678. Morgan Kaufmann.
- Casillas, J., Carse, B., and Bull, L. (2004). Fuzzy XCS: An Accuracy-Based Fuzzy Classifier System. In *Proceedings of the XII Congreso Espanol sobre Tecnologia y Logica Fuzzy (ESTYLF 2004)*.
- Fall, K. (1999). Network Emulation in the Vint/NS Simulator. In *Proc. of the 4th IEEE Symposium on Computers and Communications (ISCC'99)*, page 244.
- Georganopoulos, N. and Lewis, T. (2007). A framework for dynamic link and network layer protocol optimisation. *Mobile and Wireless Communications Summit, 2007. 16th IST*, pages 1–5.
- Handley, M. (2006). Why the internet only just works. *BT Technology Journal*, 24(3):119–129.

- Hiltunen, M. A., Schlichting, R. D., Ugarte, C. A., and Wong, G. T. (2000). Survivability through customization and adaptability: the cactus approach. In *Proc. of DARPA Information Survivability Conference and Exposition (DISCEX'00)*, volume 1, pages 294 – 307.
- Huang, K.-C., Jing, X., and Raychaudhuri, D. (2009). Mac protocol adaptation in cognitive radio networks: An experimental study. *Int. Conf. on Computer Communications and Networks*, 0:1–6.
- Kephart, J. O. and Chess, D. M. (2003). The Vision of Autonomic Computing. *IEEE Computer*, 36(1):41–50.
- Kunz, T. (2003). *Reliable Multicasting in MANETs*. PhD thesis, Carleton University.
- Luke, S., Cioffi-Revilla, C., Panait, L., and Sullivan, K. (2004). MASON: A New Multi-Agent Simulation Toolkit. In *Proc. of the 2004 Swarmfest Workshop*.
- Mena, S., Schiper, A., and Wojciechowski, P. (2003). A step towards a new generation of group communication systems. In *Proceedings of the International Conference on Middleware*, pages 414–432. Springer-Verlag, New York.
- Miranda, H., Pinto, A., and Rodrigues, L. (2001). Appia: A flexible protocol kernel supporting multiple coordinated channels. In *Proceedings of the 21st Int. Conf. on Distributed Computing Systems (ICDCS'01)*.
- Montana, D. and Redi, J. (2005). Optimizing parameters of a mobile ad hoc network protocol with a genetic algorithm. In *Proc. of the Conf. on Genetic and Evolutionary Computation (GECCO'05)*, pages 1993–1998.
- Pahlavan, K. and Krishnamurthy, P. (2001). *Principles of Wireless Networks: A Unified Approach*. Prentice Hall PTR, Upper Saddle River, NJ, USA.
- Richter, U., Mnif, M., Branke, J., Müller-Schloer, C., and Schmeck, H. (2006). Towards a generic observer/controller architecture for Organic Computing. In *GI Jahrestagung (1)*, pages 112–119.
- Rosa, L., Rodrigues, L., and Lopes, A. (2007). Appia to r-appia: Refactoring a protocol composition framework for dynamic reconfiguration. Technical report, Department of Informatics, University of Lisbon.
- Schmeck, H. (2005). Organic Computing – A new vision for distributed embedded systems. In *Proc. of the 8th Int. Symp. on Object-Oriented Real-Time Distributed Computing (ISORC'05)*, pages 201–203.
- Schöler, T. and Müller-Schloer, C. (2004). Design, implementation and validation of a generic and reconfigurable protocol stack framework for mobile terminals. In *Proc. of the 24th Int. Conf. on Distributed Computing Systems (ICDCSW'04)*, pages 362–367.
- Siekkinen, M., Goebel, V., Plagemann, T., Skevik, K.-A., Banfield, M., and Brusica, I. (2007). Beyond the Future Internet—Requirements of Autonomic Networking Architectures to Address Long Term Future Networking Challenges. *Future Trends of Distributed Computing Systems, IEEE International Workshop*, pages 89–98.
- Sözer, E. M., Stojanovic, M., and Proakis, J. G. (2000). Initialization and routing optimization for ad-hoc underwater acoustic networks. In *Proc. of Opnetwork'00*.
- Sudame, P. and Badrinath, B. R. (2001). On providing support for protocol adaptation in mobile wireless networks. *Mob. Netw. Appl.*, 6(1):43–55.
- Tomforde, S., Cakar, E., and Hähner, J. (2009a). Dynamic Control of Network Protocols - A new vision for future self-organised networks. In *Proc. of the 6th Int. Conf. on Informatics in Control, Automation, and Robotics (ICINCO'09)*, pages 285 – 290.
- Tomforde, S., Steffen, M., Hähner, J., and Müller-Schloer, C. (2009b). Towards an Organic Network Control System. In *Proceedings of the 6th International Conference on Autonomic and Trusted Computing (ATC'09)*, pages 2 – 16. Springer Verlag.
- Turgut, D., Daz, S., Elmasri, R., and Turgut, B. (2002). Optimizing clustering algorithm in mobile ad hoc networks using genetic algorithmic approach. In *Proc. of the IEEE Global Telecommunications Conference (GLOBECOM '02)*, pages 62 – 66.
- van Renesse, R., Birman, K., Hayden, M., Vaysburd, A., and Karr, D. (1998). Building adaptive systems using ensemble. *Softw. Pract. Exper.*, 28(9):963–979.
- van Renesse, R., Birman, K. P., and Maffei, S. (1996). Horus: a flexible group communication system. *Communications of the ACM*, 39(4):76–83.
- Whiteson, S. and Stone, P. (2004). Towards autonomic computing: adaptive network routing and scheduling. In *Proceedings of the International Conference on Autonomic Computing (ICAC'04)*, pages 286–287.
- Wilson, S. W. (1995). Classifier fitness based on accuracy. *Evolutionary Computation*, 3(2):149–175.
- Ye, T., Harrison, D., Mo, B., Sikdar, B., Kaur, H. T., Kalyanaraman, S., Szymanski, B., and Vastola, K. (2001). Network Management and Control Using Collaborative On-line Simulation. In *Proceedings of IEEE ICC*, Helsinki, Finland. IEEE.
- Ye, T. and Kalyanaraman, S. (2001). An adaptive random search algorithm for optimizing network protocol parameters. Technical report, Rensselaer Polytechnic Inst.

A PRACTICAL METHOD FOR SELF-ADAPTING GAUSSIAN EXPECTATION MAXIMIZATION

Nicola Greggio ^{⊙,‡}, Alexandre Bernardino [‡] and José Santos-Victor [‡]

[⊙] ARTS Lab - Scuola Superiore S. Anna, Polo S. Anna Valdera, Viale R. Piaggio, 34, 56025 Pontedera, Italy

[‡] Instituto de Sistemas e Robótica, Instituto Superior Técnico, 1049-001 Lisboa, Portugal
{ngreggio, alex, javs}@isr.ist.utl.pt

Keywords: Unsupervised learning, Self-adapting gaussian mixture, Expectation maximization, Machine learning, Clustering.

Abstract: Split-and-merge techniques have been demonstrated to be effective in overtaking the convergence problems in classical EM. In this paper we follow a split-and-merge approach and we propose a new EM algorithm that makes use of a on-line variable number of mixture Gaussians components. We introduce a measure of the similarities to decide when to merge components. A set of adaptive thresholds keeps the number of mixture components close to optimal values. For sake of computational burden, our algorithm starts with a low initial number of Gaussians, adjusting it in runtime, if necessary. We show the effectivity of the method in a series of simulated experiments. Additionally, we illustrate the convergence rates of of the proposed algorithms with respect to the classical EM.

1 INTRODUCTION

UNSUPERVISED clustering classifies different data into classes based on redundancy contained within the data sample. The classes, also called *clusters*, are detected automatically, i.e. without any input by an external agent. This method finds applications in many fields, such as in image processing (e.g. segmentation of different object from the background), sound analysis (e.g. word perceptions), segmentation of multivariate medical images, and many others. The techniques for unsupervised learning range from Kohonen maps, Growing Neural gas, k-means, to Independent component analysis, Adaptive resonance theory, etc. Particularly interesting is the Expectation Maximization algorithm applied to Gaussian mixtures which allows to model complex probability distribution functions. Fitting a mixture model to the distribution of the data is equivalent, in some applications, to the identification of the clusters with the mixture components (McLachlan and Peel, 2000).

Expectation-Maximization (EM) algorithm is the standard approach for learning the parameters of the mixture model (Dempster et al., 1977). It is demonstrated that it always converges to a local optimum. However, it also presents some drawbacks. For instance, EM requires an *a-priori* selection of model order, namely, the number of components (M) to be

incorporated into the model, and its results depend on initialization. The more Gaussians there are within the mixture, the higher will be the total log-likelihood, and more precise the estimation. Unfortunately, increasing the number of Gaussians will lead to overfitting the data and it increases the computational burden. Therefore, finding the best compromise between precision, generalization and speed is a must. A common approach to address this compromise is trying different configurations before determining the optimal solution, e.g. by applying the algorithm for a different number of components, and selecting the best model according to appropriate criteria.

1.1 Related Work

Different approaches can be used to select the best number of components. These can be divided into two main classes: *off-line* and *on-line* techniques.

The first ones evaluate the best model by executing independent runs of the EM algorithm for many different initializations, and evaluating each estimate with criteria that penalize complex models (e.g. the Akaike Information Criterion (AIC) (Sakimoto et al., 1986) and the Rissanen Minimum Description Length (MDL) (Rissanen, 1989)). These, in order to be effective, have to be evaluated for every possible number of models under comparison. Therefore, it is clear that,

for having an sufficiently exhaustive search the complexity goes with the number of tested models, and the model parameters.

The second ones start with a fix set of models and sequentially adjust their configuration (including the number of components) based on different evaluation criteria. Figueiredo and Jain proposed a method that starts with a high number of mixture parameters, merging them step by step until convergence (Figueiredo and Jain, 2002). This method can be applied to any parametric mixture where the EM algorithm can be used. Pernkopf and Bouchaffra proposed a Genetic-Based EM Algorithm capable of learning Gaussian mixture models (Pernkopf and Bouchaffra, 2005). They first selected the number of components by means of the minimum description length (MDL) criterion. A combination of genetic algorithms with the EM has been explored.

Ueda *et Al.* proposed a split-and-merge EM algorithm to alleviate the problem of local convergence of the EM method (Ueda et al., 2000). Subsequently, Zhang *et Al.* introduced another split-and-merge technique (Zhang et al., 2003). The split-and-merge equations show that the merge operation is a well-posed problem, whereas the split operation is ill-posed. Two methods for solving this problem are developed through singular value decomposition and Cholesky decomposition and then a new modified EM algorithm is constructed. They demonstrated the validity of split-and-merge approach in model selection, in terms of convergence properties. Moreover, the merge an split criterion is efficient in reducing number of model hypothesis, and it is often more efficient than exhaustive, random or genetic algorithm approaches.

1.2 Our Contribution

In this paper, we propose an algorithm for computing the number of components as well as the parameters of the mixture model. Similarly to other split-and-merge methods, our technique uses a local parameter search, that reuses the information acquired on previous steps, being suitable to problems with slowly changing distributions or to adapt the parameters when new samples are added or removed. The algorithm starts with a fixed number of Gaussians, and automatically decides whether increasing or reducing it. The key feature of our technique is the decision of when add or merge a Gaussian. In order to accomplish this at best we introduce a new concept, the dissimilarity index between two Gaussian distributions. Moreover, in order to evade local optimal solutions we make use of self-adaptative thresholds for deciding when Gaussians are split or merged. Our

algorithm starts with high thresholds levels, preventing large changes in the number of Gaussian components at the beginning. It also starts with a low initial number of Gaussians, which can be increased during computation, if necessary. The time evolution of the threshold values allows periods of stability in the number of components so that they can freely adapt to the input data. After this period of stability these thresholds become more sensitive, promoting the escape from local optimum solutions by perturbing the system configuration when necessary, until a stopping criterion has reached. This makes our algorithm results less sensitive to initialization. The algorithm is presented for Gaussian mixture models.

1.3 Outline

The paper is organized as follows. In sec. 2 we describe the notation and formulate the classical Expectation Maximization algorithm. In sec. 3 we introduce the proposed algorithm. Specifically, we describe the insertion of a new Gaussian in sec. 3.3, its merging in sec. 3.4, the initializations in sec. 3.2, and the decision thresholds update rules in sec. 3.5. Furthermore, in sec. 4 we describe our experimental set-up for testing the validity of our new technique and the results. Finally, in sec. 5 we conclude and propose directions for future work.

2 EXPECTATION MAXIMIZATION ALGORITHM

2.1 EM Algorithm: The Original Formulation

A common usage of the EM algorithm is to identify the "incomplete, or unobserved data" $\bar{y} = (\bar{y}^1, \bar{y}^2, \dots, \bar{y}^k)$ given the couple (\bar{x}, \bar{y}) - with \bar{x} defined as $\bar{x} = \{\bar{x}^1, \bar{x}^2, \dots, \bar{x}^k\}$, also called "complete data", which has a probability density (or joint distribution) $p(\bar{x}, \bar{y} | \bar{\vartheta}) = p_{\bar{\vartheta}}(\bar{x}, \bar{y})$ depending on the parameter $\bar{\vartheta}$. We define $E'(\cdot)$ the expected value of a random variable, computed with respect to the density $p_{\bar{\vartheta}}(\bar{x}, \bar{y})$.

We define $Q(\bar{\vartheta}^{(n)}, \bar{\vartheta}^{(n-1)}) = E'L(\bar{\vartheta})$, with $L(\bar{\vartheta})$ being the log-likelihood of the observed data:

$$L(\bar{\vartheta}) = \log p_{\bar{\vartheta}}(\bar{x}, \bar{y}) \quad (1)$$

The EM procedure repeats the two following steps until convergence, iteratively:

- E-step: It computes the expectation of the joint probability density:

$$Q(\bar{\vartheta}^{(n)}, \bar{\vartheta}^{(n-1)}) = E'[\log p(\bar{x}, \bar{y} | \bar{\vartheta}^{(n-1)})] \quad (2)$$

- M-step: It evaluates the new parameters that maximize Q :

$$\bar{\vartheta}^{(n+1)} = \arg \max_{\bar{\vartheta}} Q(\bar{\vartheta}^n, \bar{\vartheta}^{(n-1)}) \quad (3)$$

The convergence to a local maxima is guaranteed. However, the obtained parameter estimates, and therefore, the accuracy of the method greatly depend on the initial parameters $\hat{\vartheta}^0$.

2.2 EM Algorithm: Application to a Gaussians Mixture

When applied to a Gaussian mixture density we assume the following model:

$$p(\bar{x}) = \sum_{i=1}^{nc} w_i \cdot p_i(\bar{x}) \quad (4)$$

$$p_c(\bar{x}) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_c|^{\frac{1}{2}}} e^{-\frac{1}{2}(\bar{x} - \bar{\mu}_c)^T |\Sigma_c|^{-1} (\bar{x} - \bar{\mu}_c)}$$

where $p_c(X)$ is the component prior distribution for the class c , and with d , $\bar{\mu}_c$ and Σ_c being the input dimension, the mean and covariance matrix of the Gaussians component c , and nc the total number of components, respectively.

Consider that we have nc classes C_{nc} , with $p(\bar{x}|C_c)$ and $P(C_c) = w_c$ being the density and the *a-priori* probability of the data of the class C_c , respectively. Then the *E* and *M* steps become, respectively:

E-step:

$$P(y_c^k = 1 | x^k) = P(C_c | x^k) = E'(y_c | x^k) \quad (5)$$

$$= \frac{p(x^k | C_c) \cdot P(C_c)}{p(x^k)} = \frac{w_c \cdot p_c(x^k)}{\sum_{c=1}^{nc} w_c \cdot p_c(x^k)}$$

For simplicity of notation we refer to $E'(y_c | x^k)$ as π_c^k . This is probability that the x^k belongs to the class C_c .

M-step:

$$\bar{\mu}_c^{(n+1)} = \frac{\sum_{i=1}^k \pi_c^k \bar{x}^i}{\sum_{i=1}^k \pi_c^k} \quad (6)$$

$$\Sigma_c^{(n+1)} = \frac{\sum_{i=1}^k \pi_c^k (\bar{x}^i - \bar{\mu}_c^{(n)}) (\bar{x}^i - \bar{\mu}_c^{(n)})^T}{\sum_{i=1}^k \pi_c^k}$$

Finally, *a-priori* probabilities of the classes, i.e. the probability that the data belongs to the class c , are reestimated as:

$$w_c^{(n+1)} = \frac{1}{K} \sum_{i=1}^k \pi_c^i, \quad \text{with } c = \{1, 2, \dots, nc\} \quad (7)$$

3 SAGEM: SELF-ADAPTING GAUSSIAN EXPECTATION MAXIMIZATION ALGORITHM

The key issue of our algorithm is looking whether one or more Gaussians are not increasing their own likelihood during optimization. In other words, if they are not participating in the optimization process, they will be split into two new Gaussians or merged. We will introduce two new concepts related to the state of a Gaussian component:

- The *Area* of a Gaussian, that measures how much a Gaussian component covers the data space; This actually correspond to the covariance matrix determinant;
- Its age, that measures how long the component's own likelihood does not increase significantly (see sec. 3.1);

Then, the merge and split processes are controlled by the following adaptive thresholds:

- One adaptive threshold L_{TH} for determining a significant increase in likelihood (see sec. 3.5);
- One adaptive threshold A_{TH} for triggering the merge or split process based on the component's own age (see sec. 3.5);
- One adaptive threshold M_{TH} for deciding to merge two Gaussians based on their area and position (see sec. 3.4);
- One adaptive threshold S_{TH} for deciding to split a Gaussian based on its area (see sec. 3.3).

It is worth noticing that even though we consider four thresholds to tune, all of them are adaptive, and only require a coarse initialization. All these thresholds cooperate in keeping the number of components stable for a period after a split or merge operation (see sec. 3.5).

These parameters will be fully detailed within the next sections.

3.1 SAGEM Formulation

Our formulation can be summarized within four steps:

- Initializing the parameters;
- Adding a Gaussian;
- Merging two Gaussians;
- Updating decision thresholds.

Each Gaussian element i of the mixture is represented as follows:

$$\bar{\vartheta}_i = \rho(w_i, \bar{\mu}_i, \Sigma_i, \xi_i, \Lambda_{last(i)}, \Lambda_{curr(i)}, a_i) \quad (8)$$

where w_i is the *a-priori* probabilities of the class, $\bar{\mu}_i$ is its mean, Σ_i is its the covariance matrix, ξ_i its *area*, $\Lambda_{last(i)}$ and $\Lambda_{curr(i)}$ are its last and its current log-likelihood value, and a_i its *age*. Here, we define two new elements, the area and the age of the Gaussian, which will be described later.

During each iteration, the algorithm keeps memory of the previous likelihood. Once the re-estimation of the vector parameter $\bar{\vartheta}$ has been computed in the EM step, our algorithm evaluates the current log-likelihood of each single Gaussian as:

$$\Lambda_{curr(i)}(\bar{\vartheta}) = \sum_{j=0}^{k-1} \log(w_i \cdot p_i(\bar{x}^j)) \quad (9)$$

Then, for each Gaussian the difference between the current and last log-likelihood value is compared with a increment threshold, equal for all the Gaussians, L_{TH} . If the difference is smaller than the threshold L_{TH} , i.e. there is no significant increment, the algorithm increases the *age* a_i of the relative Gaussian. If a_i overcomes the age threshold A_{TH} (i.e. the Gaussian i does not increase its own likelihood for a predetermined number of times significantly), the algorithm decides whether to split this Gaussian or merging it with existing ones.

Then, after a certain number of iterations the algorithm will stop. The whole algorithm pseudocode is shown in Fig. 3.1.

3.2 Split Threshold Initialization

At the beginning, S_{TH} will be automatically initialized to the area of the covariance of all the data set, relative to the total mean. The other decision thresholds will be initialized as follows:

$$\begin{aligned} M_{TH-INIT} &= 0 \\ L_{INIT} &= k_{LTH} \\ Age_{INIT} &= k_{ATH} \end{aligned} \quad (10)$$

with k_{LTH} and k_{ATH} (namely, the minimum amount of likelihood difference between two iterations and the number of iterations required for taking into account the lack of a likelihood consistent variation) relatively low (i.e. both in the order of 10, or 20). Of course, higher values for k_{LTH} and smaller for k_{ATH} give rise to a faster adaptation, however adding instabilities.

3.3 Adding a Gaussian

If the covariance matrix area of the examined Gaussian at each stage overcomes the maximum area threshold S_{TH} , then another Gaussian is added to the mixture.

Algorithm 3.1. Pseudocode.

```

1: - Parameter initialization;
2: while (stopping criterion is not met) do
3:    $\Lambda_i$  evaluation, for  $i = 0, 1, \dots, c$ ;
4:    $L(\bar{\vartheta})$  evaluation (1);
5:   re-estimate priors  $w_i$ , for  $i = 0, 1, \dots, c$ ;
6:   recompute center  $\bar{\mu}_i^{(n+1)}$  and covariances  $\Sigma_i^{(n+1)}$ , for
    $i = 0, 1, \dots, c$  (5);
7:   - Evaluation whether changing the Gaussian
   distribution structure -
8:   for ( $i = 0$  to  $c$ ) do
9:     if ( $a_i > A_{TH}$ ) then
10:      if ( $(\Lambda_{curr(i)} - \Lambda_{last(i)}) < L_{TH}$ ) then
11:         $a_i + = 1$ ;
12:        - General condition for changing satisfied;
        now checking those for each Gaussians -
13:        if ( $\Sigma_i > S_{TH}$ ) then
14:          if ( $c < \text{maxNumGaussians}$ ) then
15:            add Gaussian  $\rightarrow$  split;
16:             $c + = 1$ ;
17:            reset  $S_{TH}$  to its initial value;
18:            reset  $L_{TH}$  to its initial value;
19:            return;
20:          end if
21:        end if
22:      for ( $j = 0$  to  $c$ ) do
23:        eval  $\chi_{i,j}$  (14)
24:        if ( $\chi_{i,j} < M_{TH}$ ) then
25:          merge Gaussian  $\rightarrow$  merge;
26:           $c - = 1$ ;
27:          reset  $M_{TH}$  to its initial value;
28:          reset  $L_{TH}$  to its initial value;
29:          return;
30:        end if
31:      end for
32:       $S_{TH} = S_{TH} \cdot (1 + \alpha \cdot \xi_i)$ ;
33:       $M_{TH} = M_{TH} \cdot (1 + \gamma \cdot \chi_{A,B})$ ;
34:    end if
35:  end if
36: end for
37: end while

```

More precisely, the original Gaussian with parameters $\bar{\vartheta}_{old}$ will be split within other two ones. The new means, A and B , will be:

$$\begin{aligned} \bar{\mu}_A &= \bar{\mu}_{old} + \frac{1}{2}(\Sigma_{i=j})^{\frac{1}{2}}; & \bar{\mu}_B &= \bar{\mu}_{old} - \frac{1}{2}(\Sigma_{i=j})^{\frac{1}{2}} \\ i, j &= \{1, 2, \dots, d\} \end{aligned} \quad (11)$$

where d is the input dimension.

The covariance matrixes will be updated as:

$$\begin{aligned} \Sigma_{A(i,j)} &= \Sigma_{old(i,j)}; & \Sigma_{B(i,j)} &= \Sigma_{old(i,j)} \\ i, j &= \{1, 2, \dots, d\} \end{aligned} \quad (12)$$

The *a-priori* probabilities will be

$$w_A = \frac{1}{2}w_{old}; \quad w_B = \frac{1}{2}w_{old} \quad (13)$$

Finally, their ages, a_A and a_B , will be reset to zero.

3.4 Merging Two Gaussians

For each Gaussian B other than the given one A , the algorithm evaluates a dissimilarity index. If that index is lower than the merge threshold M_{TH} , Gaussian B will be merged with Gaussian A . There are still no exact methods for computing the overlap between two Gaussians in arbitrary dimension. The existing methods use iterative procedures to approximate the degree of overlap between two components (Sun et al., 2007).

Here, we introduce a new dissimilarity index between Gaussian distributions, χ , that uses a closed form expression, evaluated as follows:

$$\begin{aligned} \chi_{i,j} &= \|\rho_{i,j} + \delta_{i,j}\|^2 \\ \rho_{i,j} &= \frac{\sum_{i,j} \|\Sigma_{A_{i,j}} - \mu_{A_i}\| \cdot \|\Sigma_{B_{i,j}} - \mu_{B_i}\|}{\xi_i \cdot \xi_j} \quad (14) \\ &\text{with } i, j = \{1, 2, \dots, d\} \end{aligned}$$

where d is the input dimension, $\chi_{i,j}$ is a new *dissimilarity index* for mixture components, $\delta_{i,j}$ is the Euclidean distance between the two Gaussians i and j . Another solution would have been to choose the Mahalanobis distance (Mahalanobis, 1936) instead of $\chi_{i,j}$. This takes into account both the mean and the covariance of the matrixes. However, the Mahalanobis distance gives rise to zero if the two Gaussians have the same mean, independently whether they cover the same space (or, whether they have the same area ξ), then rendering it not suitable for our purposes. Again, we to avoid singularities, we use the *area* instead of the covariance matrix determinant at the denominator of the correlation index $\rho_{i,j}$. Finally, we defined $\chi_{i,j}$ as the square of the norm because this heightens values of $\|\rho_{i,j} + \delta_{i,j}\| > 1$ while reducing those $\|\rho_{i,j} + \delta_{i,j}\| < 1$.

The new Gaussian (*new*) will have the following mean and covariance:

$$\begin{aligned} \mu_{new(i)} &= \frac{1}{2}(\mu_{iA} + \mu_{iB}); \quad \rho_{new(i,j)} = \frac{1}{2}(\rho_{A_{i,j}} + \rho_{B_{i,j}}) \\ &\text{with } i, j = \{1, 2, \dots, d\} \quad (15) \end{aligned}$$

where d is the input dimension. The *a-priori* probabilities for the remaining Gaussian will be updated as:

$$\begin{aligned} c_{new} &= c_{old} - 1; \quad w_i = w_i + w_{old}/c_{new} \\ &\text{with } i|_{i \neq old} = \{1, 2, \dots, c_{new}\} \quad (16) \end{aligned}$$

Finally, like when adding a Gaussian, the age a_{new} of the resultant Gaussian will be reset to zero.

3.5 Updating Decision Thresholds

The thresholds L_{TH} , S_{TH} , and ξ_{TH} vary with the following rules:

$$\begin{aligned} L_{TH} &= L_{TH} - \lambda \cdot L_{TH} = L_{TH} \cdot (1 - \lambda) \\ S_{TH} &= S_{TH} + \alpha \cdot (\xi - S_{TH}) = S_{TH} \cdot (1 + \alpha \cdot \xi) \\ M_{TH} &= M_{TH} + \gamma \cdot (\chi_{A,B} - M_{TH}) = M_{TH} \cdot (1 + \gamma \cdot \chi_{A,B}) \\ &\text{with } \xi - S_{TH} < 0, \quad \chi_{A,B} - M_{TH} > 0 \quad (17) \end{aligned}$$

with λ , α , and γ chosen arbitrarily low (we used $\lambda = 0.04$, $\alpha = 0.04$, $\gamma = 0.001$). Following this rules L_{TH} will decrease step by step, approaching the current value of the global log-likelihood increment. This is the same for S_{TH} , which will become closer to the *area* of some Gaussian, and for M_{TH} that will increase. This will allow the system to avoid some local optima, by varying its configuration if a stationary situation occurs.

Finally, every time a Gaussian is added or merged, these thresholds will be reset to their initial value.

3.6 Computational Complexity Evaluation

Within this section we will use the following convention: ng is the number of the mixture Gaussian components, k is the number of input vectors, d is the number of input dimension, and it is the number of iterations.

The computational burden of the EM algorithm is, referring to the pseudocode in tab. 3.1 as follows:

- the original EM algorithm (steps 3 to 6) take $O(k \cdot d \cdot ng)$ for 3 and 6, while step 4 and step 5 take $O(1)$ and $O(k \cdot ng)$;
- our algorithm takes $O(ng)$ for evaluating all the Gaussians (step 8 to 36) and another $O(ng)$ in step 23 for evaluating the dissimilarity between each Gaussian (14);
- our merge and split (step 15 and 25) operations require $O(d)$ and $O(d \cdot ng)$, respectively.
- the others take $O(1)$.

Therefore, the original EM algorithm takes $O(k \cdot d \cdot ng)$, while our algorithm adds $O(d \cdot ng^2)$ on the whole, giving rise to $O(k \cdot d \cdot ng) + O(d \cdot ng^2) = O(k \cdot d \cdot ng + d \cdot ng^2) = (ng \cdot d \cdot (k + ng))$. Considering that usually $d \ll k$ and $ng \ll k$ this does not add a considerable burden, while giving an important improvement to the original computation in terms of self-adapting to the data input configuration at best.

4 EXPERIMENTAL VALIDATION

In order to evaluate the performance of our algorithm, we tested it by classifying different input data randomly generated by a known Gaussian mixture, and subsequently saved to a file. This is in order to use the same input data to our algorithm, SAGEM, and to the original EM. Our algorithm starts with a low initial number of Gaussians, while the original EM starts with the exact number of Gaussians, i.e. the mixture configuration we generated the input data points from. Moreover, in order to make a fair comparison, both the algorithms started with the same input Gaussian means (of course, since SAGEM has fewer Gaussian components than EM these are a subset of those of EM).

We refer to the EM with the exact number of Gaussians that generated the input data as the best EM algorithm applicable, i.e. the algorithm that uses the best compromise between number of Gaussian components and final likelihood. Therefore, we will use its results as comparison for our algorithm.

4.1 Experimental Set-up

We made different trials, with mixtures containing 4, 8, 12, 14, and 16 Gaussians. Each of them contains 2000 points in two dimensions. We choose to show the results for 2-dimensional input because they are easier to show than multidimensional ones (for instance, a 2-dimensional Gaussian is represented in 2D as an ellipse). As the ratio (*# Gaussians*)/(*# data points*) increases, it become harder to reach a good solution in the reasonable number of steps. Therefore, we are interested in evaluating how SAGEM behaves when the model complexity gradually increases.

We evaluated the performance of our algorithm compared with the original EM principally based on three main points:

- Final log-likelihood value;
- Final number of Gaussian components;
- Number of iterations needed for reaching stability.

Therefore, we used the following equations for evaluating the error on the final Log-likelihood, and the error on the predicted number of Gaussian components as follows, respectively (the required number of iterations can be extracted from the plots in Fig. 1, directly).

$$\begin{aligned} \text{Log} - \text{Lik}_{err} &= \frac{\text{Log} - \text{Lik}_{(SAGEM)}}{\text{Log} - \text{Lik}_{(EM)}} \\ \text{NumGauss}_{err} &= \text{NumGauss}_{(SAGEM)} - \text{NumGauss}_{(EM)} \end{aligned} \quad (18)$$

In table I the output of the different computations are shown.

The results will be discussed within the next section.

4.2 Discussion

The output of the two algorithms is shown in Fig. 1. The input data (green points) with the generation mixture (blue) and the evaluated one (red) are represented in the same figure, while the resultant log-likelihood (blue for the EM algorithm and red for SAGEM) is shown. It is worth noticing that we also represented the iterations at which a split or a merge operation is performed, as vertical blue and red lines, respectively. Here, it is possible to see how just after a splitting or merging operation the final log-likelihood has some spikes. When a merge operation is performed the algorithm decreases the number of Gaussian components, therefore decreasing the log-likelihood momentarily abruptly. An example is at iteration 92 of the 12-Gaussian plot. Besides, when a new class is added by means of a splitting operation, it may happen that the final log-likelihood starts increasing smoothly due to the new component's contribute (e.g. see the 8-Gaussian plot at iteration 22, or the 14-Gaussian plot at iteration 50), or has a decreasing spike (e.g. see the 8-Gaussian plot at iteration 34 or 53) If the components are reorganized by the EM procedure in order to describe the data better, the log-likelihood will start to increase smoothly.

Finally, Fig. 2.6 shows the 3D histogram representation of a generated Gaussian mixture data and the estimated one. Due to space limitations, we choose to show only the one that gave rise to the worst log-likelihood estimation plot, i.e. the one with 8 Gaussians.

In table I the results of the different computations are shown. The table contains, for both SAGEM and the EM algorithm:

- The starting number of Gaussian components;
- The final (i.e. detected, in the case of the SAGEM approach) number of Gaussian components;
- The error on the final number of components.
- The final reached log-likelihood.
- The error on the final log-likelihood, as absolute value.

Due to the formulation in (18), if $\text{Log} - \text{Lik}_{err} < 1$ than SAGEM reached a final log-likelihood greater than EM (both are negative), and vice versa. Similarly, when $\text{NumGauss}_{err} < 0$ it means that SAGEM

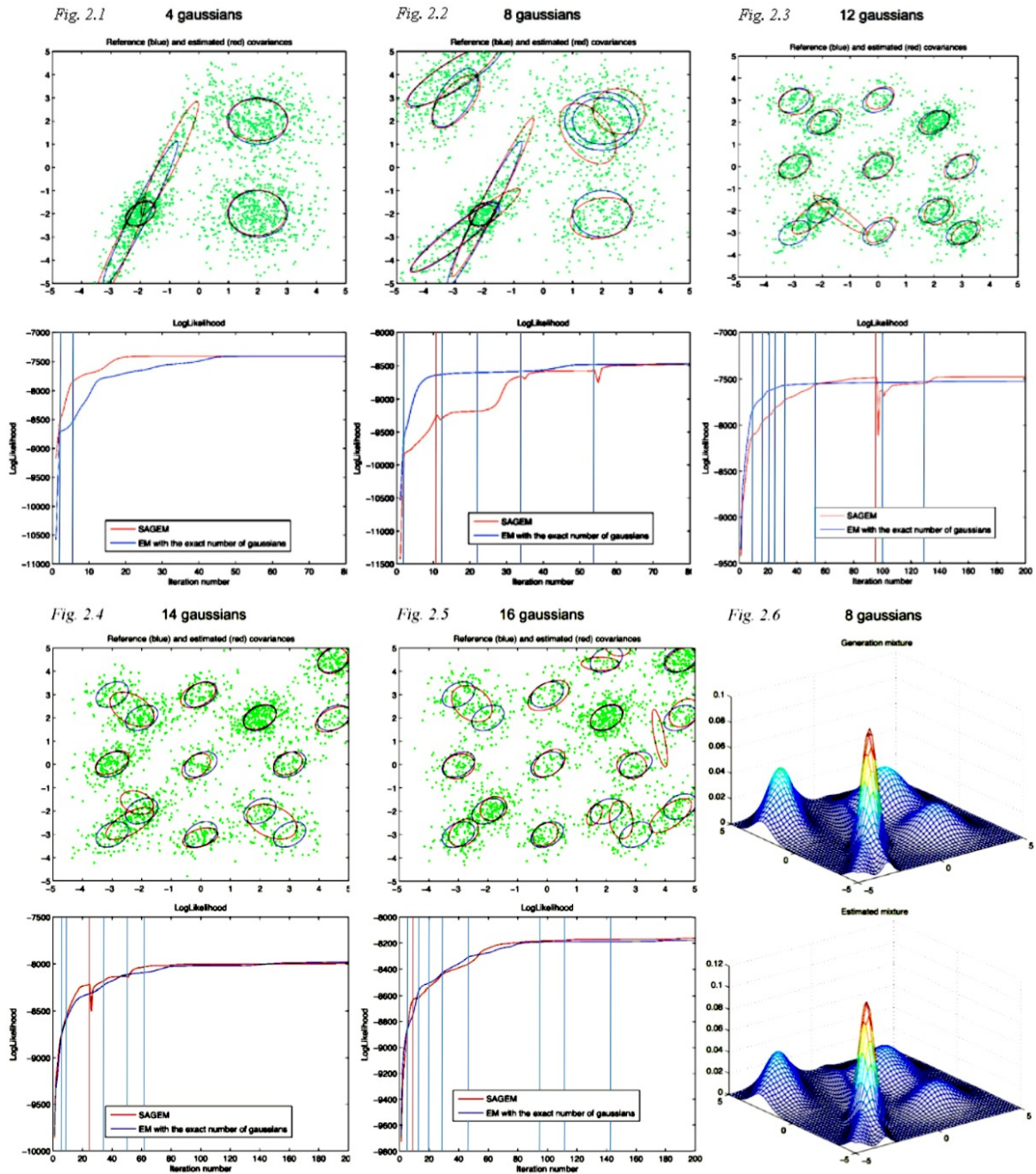


Figure 1: The 2D representation of the final SAGEM Gaussian mixture vs. the real one and the SAGEM and EM log-likelihood output as function of the iterations number, for different input mixtures of data (4, 8, 12, 14, and 16 Gaussian components). Moreover the 8-Gaussians case comparison between the generated and computed mixtures is shown.

used fewer Gaussians than real ones. These two comparisons are shown in plots 2(a), 2(b), and 2(c) respectively.

We can see that the final SAGEM log-likelihood is often better than the one obtained with the original EM with the exact number of Gaussians, except for the case of 8-Gaussians. There are three interesting

points:

- Within the 14-Gaussians case SAGEM reached a higher log-likelihood than EM even with less Gaussians components than required.
- Increasing the number of components, the points representing each class will be fewer. This in-

Table 1: Experimental results.

Number of effective mixture Gaussians	Algorithm	Starting number of Gaussians	Arrival number of Gaussians	Error on the number of Gaussians	Final log-likelihood	Error on the final log-likelihood
4	SAGEM	2	4	0	-7402.82	-2.26
	EM	4	4	/	-7405.08	/
8	SAGEM	4	8	0	-8448.56	14.43
	EM	8	8	/	-8434.13	/
12	SAGEM	6	12	0	-7501.62	-28.36
	EM	12	12	/	-7526.74	/
14	SAGEM	8	12	-2	-7928.95	-53.71
	EM	14	14	/	-7982.66	/
16	SAGEM	10	17	+1	-8161.29	-17.13
	EM	16	16	/	-8178.42	/

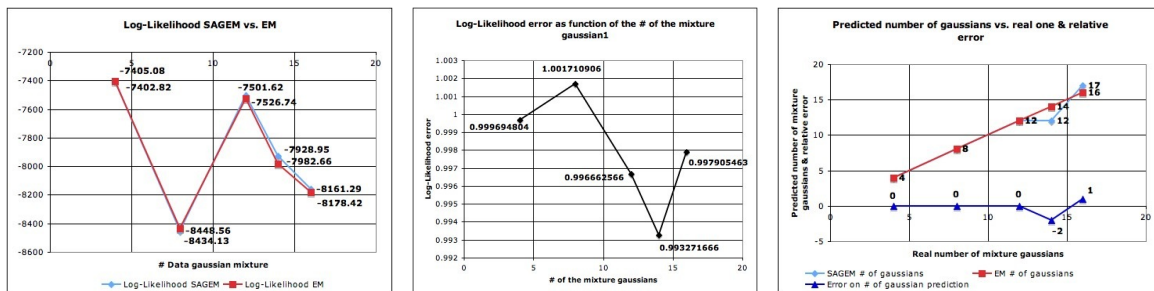


Figure 2: The final log-likelihood of both SAGEM and EM as function of the number of input data Gaussian components (a), the relative error (b), and the final reached number of Gaussians as function of the input ones (c).

creases the difficulty of clustering them into distinct classes. Nevertheless, SAGEM has been able to detect a number of required Gaussians close to the original one (12 vs. 14 and 17 vs. 16), and to have a better final log-likelihood than EM.

- Regarding to the plots of the log-likelihood we can see that both SAGEM and EM reach a similar likelihood within the first 60-80 iterations, even though EM is usually faster in increasing it. Some spikes in the SAGEM plots occur, due to the modification of the number of Gaussians. In fact, when merging two mixture components the log-likelihood momentarily decreases.
- Even though the 8-Gaussian case is the worst one in terms of final likelihood, the histogram of the final Gaussian mixture is very similar to the true one.

5 CONCLUSIONS AND FUTURE WORK

In this paper we proposed a new split-and-merge Expectation Maximization algorithm, called SAGEM. The algorithm is specific for Gaussian mixtures. SAGEM starts with a fixed number of Gaussians, and automatically decides whether increasing or reducing it. We introduced a new concept, a dissimilarity in-

dex between two Gaussian distributions. Moreover, in order to evade local optimal solutions we make use of self-adaptative thresholds for deciding when Gaussians are split or merged. We tested it with different Gaussian mixtures, comparing its results with those of the EM original algorithm set with the model complexity that matches those we generated the data from (ideal case). We showed that our algorithm is capable to evaluate a number of Gaussian components close to the true one, and to provide a final mixture description with a log-likelihood comparable with the original EM, sometimes even better. Finally, its convergence occurs with the same number of iterations than it does for the original EM (60-80 iterations).

5.1 Future Work

At the moment we tested our algorithm with synthetic data. As future work, we will test SAGEM for the purpose of image segmentation on real images captured in order to test it in real robotic applications, where the computational burden must be kept low. More specifically, being part of the RobotCub European Project, we will test it with the iCub robotics platform.

ACKNOWLEDGEMENTS

This work was supported by the European Commission, Project IST-004370 RobotCub and FP7-231640 Handle, and by the Portuguese Government - Fundação para a Ciência e Tecnologia (ISR/IST pluriannual funding) through the PIDDAC program funds and through project BIO-LOOK, PTDC / EEA-ACR / 71032 / 2006.

REFERENCES

- Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood estimation from incomplete data via the em algorithm. *J. Royal Statistic Soc.*, 30(B):1–38.
- Figueiredo, A. and Jain, A. (2002). Unsupervised learning of finite mixture models. *IEEE Trans. Patt. Anal. Mach. Intell.*, 24(3).
- Mahalanobis, P. C. (1936). On the generalized distance in statistics. *Proceedings of the National Institute of Sciences of India*, 2(1):39–45.
- McLachlan, G. and Peel, D. (2000). Finite mixture models. *John Wiley and Sons*.
- Pernkopf, F. and Bouchaffra, D. (2005). Genetic-based em algorithm for learning gaussian mixture models. *IEEE Trans. Patt. Anal. Mach. Intell.*, 27(8):1344–1348.
- Rissanen, J. (1989). Stochastic complexity in statistical inquiry. *Wold Scientific Publishing Co. USA*.
- Sakimoto, Y., Iahiguro, M., and Kitagawa, G. (1986). Akaike information criterion statistics. *KTK Scientific Publisher, Tokio*.
- Sun, H., Sun, M., and Wang, S. (19-22 August 2007). A measurement of overlap rate between gaussian components. *Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong*.
- Ueda, N., Nakano, R., Ghahramani, Y., and Hiton, G. (2000). Smem algorithm for mixture models. *Neural Comput*, 12(10):2109–2128.
- Zhang, Z., Chen, C., Sun, J., and Chan, K. (2003). Em algorithms for gaussian mixtures with split-and-merge operation. *Pattern Recognition*, 36:1973 – 1983.

PATTERNS FOR TEMPORAL REQUIREMENTS ENGINEERING

*A Level Crossing Case Study**

A. Mekki^{1,2}, M. Ghazel^{1,2} and A. Toguyeni^{1,3}

¹ Univ Lille Nord de France, F-59000 Lille, France

² INRETS, ESTAS, F-59666, Villeneuve d'Ascq, France

³ EC LILLE, LAGIS, F-59651, Villeneuve d'Ascq, France

{ahmed.mekki, mohamed.ghazel}@inrets.fr, armand.toguyeni@ec-lille.fr

Keywords: Model Transformation, Time-Constrained System, UML State Machines, Timed Automata, Verification and Validation.

Abstract: This work presents a method for verifying temporal requirements of time-constrained systems. The method predates by establishing a new time constraints (properties) taxonomy. Then, a basis of observation patterns relative to the predefined requirements is developed. Our approach allows the automated verification of temporal requirements, initially expressed in a semi-formal formalism, through model transformation and model-checking. The contributions of the paper are: the definition of a new time constraints (properties) typology as well as a basis of appropriate State Machines (SM) observation patterns. The second contribution consists in developing an algorithm for transforming UML SM with time annotations into Timed Automata (TA). In practice, in order to verify the temporal aspects of a given specification, the observation patterns relative to the investigated properties are instantiated to make appropriate observers. Then using our transformation algorithm, the system specification (denoted in the shape of an UML SM model) with time annotations as well as the obtained observers are translated into TA models. The TA system model is next synchronized with the TA observers. Thereby, the verification process is reduced to a reachability analysis.

1 INTRODUCTION

Given their practical implication on safety and correctness of critical applications (e.g. transportation systems, nuclear plants), specification and verification are one of the most important research topics in critical systems engineering since such kind of systems must achieve a high level of robustness and reliability. In addition, these systems usually involve time-dependent functionality. Consequently, methods for behavior modeling and verifying (especially temporal requirements) are increasingly important. The most used approaches for specifying timed systems are based on **Timed Automata (TA)**. TA are well suited for expressing timed behavior and for modeling real-time components. A number of automatic verification tools for TA have been developed and have proven to be efficient e.g., Uppaal (Larsen et al., 1997) and Kronos (Yovine, 1997). Nevertheless, specifying and ver-

ifying time constraints is becoming a more and more difficult task due to the widespread applications and increasing complexity of checked systems. Despite the different advantages proposed by TA, such as parallel composition, users often need to manually express the time properties into a set of clock variables with complex calculated clock constraints. This process is tedious, error-prone and requires sophisticated logical and/or mathematical skills.

On the other hand, in order to cope with the complexity of critical systems engineering, approaches based on **Model Driven Engineering (MDE)** seems to be very useful (Schmidt, 2004). The aim of this work is to introduce a new temporal requirements' verification method based on MDE. First, we define a Patterns' Basis for monitoring time constraints. Indeed, based on a new time constraints' classification, we developed a set of time observation patterns expressed in **Unified Modeling Language (UML) State Machines (SM)**(UML, 2009): this is expected to be a relatively inexpensive activity since this procedure is done once and for all. UML has been chosen since it is relatively intuitive, offers a graphic description, is

*This research has been partially supported by Region Nord Pas de Calais and European fund Feder under the FUI National project FerroCOTS, labelled by i-Trans.

implemented by several tools and finally is a standard notation well supported by the **Object Management Group**. This set of patterns facilitates high-level system design. These patterns cover a large class of common time constraints.

Since our aim is to keep a high precision level, a subsequent step consists in giving an accurate definition of each developed pattern. Hence, for each pattern, we give (1) a textual definition, (2) an UML SM model, (3) a structured English specification and finally (4) a temporal logic expression (**Timed Computational Tree Logic (TCTL)**) relative to the property concerned.

Concretely, the verification process is based on the set of patterns. The suitable patterns corresponding to the time constraints extracted from the system requirements are picked up and instantiated. This instantiation step generates a set of SM Observers. The SM observers are translated into more formal notation, the TA, which provides support for the properties' verification. The translation is made according to a transformation algorithm that will be discussed later in the paper. In this way, analyzers exploit the benefits of formal notations without having to go through the complex and expensive formal modeling phase. This transformation generates a set of TA Observers. A system's model, which is also generated by the same transformation algorithm, is synchronized with the obtained TA observers to obtain a global model. Hence, the verification task is performed, on this obtained model, with a reachability analysis while checking whether the observers' forbidden states - corresponding to constraints violation - are reachable.

The paper is organized as follows. In Section 2, we set the context and we briefly go through some related works. Section 3 describes our first contribution by introducing the new time-constraints taxonomy and the patterns basis. The second contribution of our method is outlined in Section 4 where the translation from UML SM, with time annotations, to TA is described. The method is illustrated using a **Level Crossing (LC)** case study in Section 5. Section 6 concludes the paper while drawing some future work.

2 CONTEXT AND BACKGROUND

2.1 Related Work

There are many recent research efforts in the field of time-constrained system validation. Only two of these research will be discussed in this section. First, based on the Dwyer (Dwyer et al., 1999) pattern basis,

Dhaussy (Dhaussy et al., 2009) defines a textual language, called "CDL", for requirement specification. The requirements are then translated into observer automata. Furthermore, Dhaussy defines for each requirement a path, called "context" where the requirement should be checked. Finally, the system model, the observer automata and the context are translated into IF notation (Intermediate Format). Then, the verification is carried out using the IFx tool. Second, Nascimento (Nascimento et al., 2009) presents an approach for automatic generation of network of timed automata from a functional specification depicted via UML class and sequence diagrams. Nascimento uses UML sequence diagrams for the property specification phase. However, sequence diagrams suffer from a limited expressiveness when dealing with temporal aspects, since they only depict order.

Unlike the above methods, our approach uses TA as target notation; TA are assumed to be more expressive and well supported.

On the other hand, several projects have introduced natural-language-based approaches where natural language is mapped into a more formal specification. (Dwyer et al., 1999) proposes several patterns applicable to properties specification expressed in different formalisms and logics such as LTL, CTL, GIL, and quantified regular expressions (QRE). (Konrad and Cheng, 2005) proposes an extension to Dwyer's classification and real-time properties are added to the original classification. Moreover, TCTL, MTL and RTGIL are used to specify the added real-time properties.

Comparatively to the above-mentioned works, our contribution offers the following advantages:

- Our method takes advantage of the flexibility and expressiveness of UML SM in modeling tasks and the precision of TA formalism in the verification tasks, also UML SM are more expressive than UML sequence diagrams or UML collaboration diagrams used in other works,
- TA are well supported,
- Patterns facilitate high-level specification and promote reusability and knowledge capitalization,
- The verification task is reduced to a reachability analysis, this allows us to overcome some limitations met with some existing tools, such as Upaal.

2.2 Observer Technique

We deal with observer whenever we set artifacts to watch system behavior (Dong et al., 2008). Let us recall here that the goal of our approach is to check whether the temporal requirements expected from a

given system are satisfied. Hence, we make use of observers in order to express the satisfaction vs the violation of the predefined requirements (Ghazel et al., 2009). Typically, checking a given temporal property consists in examining whether the error state of the corresponding observer is reachable.

3 OBSERVATION PATTERNS

In this section, we first propose a classification of all the common temporal requirements one may meet when dealing with critical systems. Then, we develop a structured English grammar that we use to express the predefined properties. Next, we introduce the patterns used in order to monitor the predefined temporal requirements. Finally, a standardized description of these patterns is suggested.

3.1 Main Time-constraints

We strive to identify all the common requirements one may meet when dealing with critical systems. The main identified requirements are defined and explained in Table 1 (The relation which denotes that a system **S** satisfies a requirement **R** is written **S|=R**) and also are depicted in the shape of a UML Class diagram (Figure 1). This classification offers the advantage that it deals with requirements on events only, since we used to express the requirements on states using two events: the first event represents the activation of the state and the other the deactivation.

3.2 Structured English

To facilitate the expression and the formalization of temporal properties, we have developed a structured English grammar. This grammar supports both qualitative and quantitative properties. Each sentence generated by our grammar describes a temporal property and serves as handler that helps expressing and understanding the requirement. Our grammar is expressed below using BNF (Backus-Naur Form) notation:

```

Property = { <Scope> : <Specification> };
Specification = { <Entity> <Obligation> occur <Reference> };
Scope = Global | Before <Entity> | After <Entity> | Between
    <Entity> and <Entity>;
Entity = "Event" | "Active( State)" | "Desactive(State)";
Obligation = must | cannot;
Reference = ((exactly at <time> over) | (After [a delay of <
    time> over] | (<Before [a delay of <time> over])) <Entity>;
Time = <Number> tu;
Number = <Digit>+;
Digit = {0|1|2|3|4|5|6|7|8|9};
    
```

Literal terminals are given in **bold** font, non-literal terminals are delimited by quotation marks (“ ”) and non-terminals are given in *italics*. The start symbol of the grammar is property and the language \mathcal{L} of the grammar is finite, since the grammar is non-circular and has no repetitions.

3.3 Observation Patterns Basis

A pattern is a commonly reusable model in software systems that guarantees a set of characteristics and functionalities. The identification of a pattern is based on the context in which it is used. The goal behind developing patterns is to offer a support for system design and development. Using patterns helps in keeping design standardized and useful and minimizes the reinventing in the design process, since they facilitate reusability and knowledge capitalization (Gamma et al., 1995).

In this work, we define a set of patterns which will serve as basis to generate observers for all the identified temporal requirements. The notation used is UML State Machines. The basis of patterns is introduced regardless the systems’ specification and is used to model all the common temporal requirement types that one may express. This pattern basis guarantees the reusability and the genericity of the mechanisms developed within our approach.

3.4 Pattern Formalization

We have introduced a new temporal requirement classification that is used in implementing our pattern repository. Additionally, we include a graphical representation of each pattern in the shape of UML SM diagram. This field will be used later as input model to the model transformation phase. Each pattern in the repository contains the following fields:

Pattern Name: The pattern name serves as a handle for the pattern’s use and describes the type of the pattern.

Pattern Definition: A short description and definition of the requirement for which the pattern is used.

Scoped Structured English Specification: The scoped structured English sentence captures the invoked property using the grammar defined previously. The scope, initially introduced by Dwyer in (Dwyer et al., 1999), is used to express the applicability interval (scope) of the property. Four scopes are used in our grammar: *globally*, *before* an event occurs, *after* an event occurs and *between* two events.

Table 1: Temporal Requirement's Taxonomy Descriptions.

Class	Category	Pattern Name	Description
Quantitative	QuantitativeAbsence	Forbidden Before	R ensures that an event (E_{mon}) must never occur before a minimum T_{before} (time unit over E_{Ref}). $S \models R$ is true if this event does not occur before T_{before} .
		Forbidden After	R ensures that an event (E_{mon}) must never occur after a deadline T_{after} (time unit over E_{Ref}). $S \models R$ is true if this event does not occur after T_{after} .
		Forbidden Between	R ensures that an event (E_{mon}) must never occur between a temporal interval $]t_{Begin}; t_{End}[$ (over E_{Ref}). $S \models R$ is true if this event does not occur between temporal interval $]t_{Begin}; t_{End}[$.
	QuantitativePresence	MinimumDelay	R ensures that an event (E_{mon}) must occur after a minimum time T_{min} (time unit over E_{Ref}). $S \models R$ is true if this event occurs after T_{min} .
		MaximumDelay	R ensures that an event (E_{mon}) must occur before a deadline T_{max} (time unit over E_{Ref}). $S \models R$ is true if this event occurs before T_{max} .
		Punctuality	R ensures that an event (E_{mon}) must occur at one punctual date t (time unit over E_{Ref}). $S \models R$ is true if this event occurs at the t date.
Qualitative	Recurrence	UnboundedRecurrence	R ensures that an event (E_{mon}) must occur infinity of time. $S \models R$ is true if this event occur.
		BoundedRecurrence	R ensures that an event (E_{mon}) must occur k time. $S \models R$ is true if this event occur k time.
	QualitativePresence	PresenceAfter	R ensures that an event (E_{mon}) must occur after E_{Ref} have been detected. $S \models R$ is true if this event occurs at least once after E_{Ref} .
		PresenceBefore	R ensures that an event (E_{mon}) must occur before E_{Ref} . $S \models R$ is false if E_{Ref} occurs before E_{mon} .
	QualitativeAbsence	AbsenceAfter	R ensures that an event must never occur after E_{Ref} . $S \models R$ is true if this event does not occur.
		AbsenceBefore	R ensures that an event (E_{mon}) must never occur before E_{Ref} . $S \models R$ is true if E_{mon} does not occur before E_{Ref} .

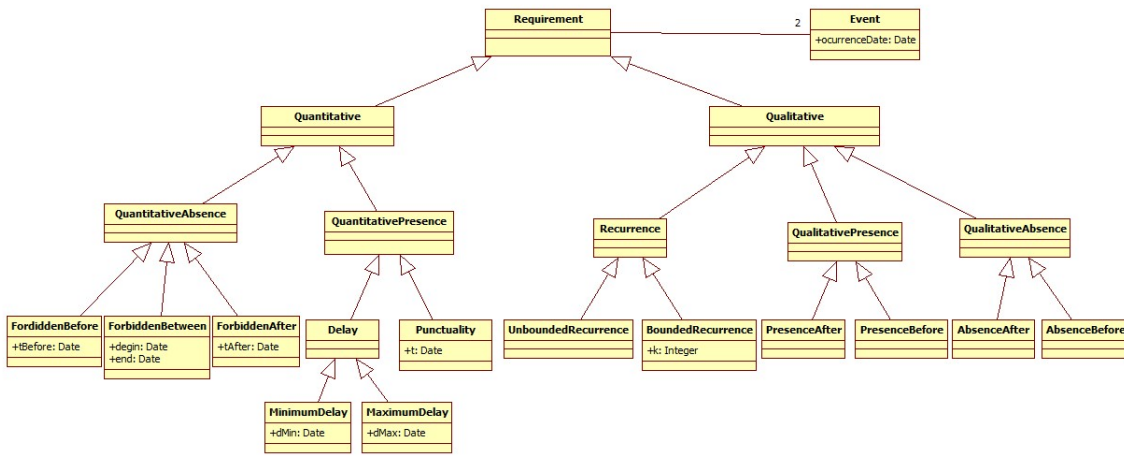


Figure 1: Temporal Requirements Classification.

Temporal Logic Description: Contains mappings of the property monitored by the pattern to TCTL for each of the four defined scopes. We chose TCTL since it allows quantitative temporal properties expression.

4 TRANSFORMATION APPROACH

Since the observation patterns' basis has been introduced, we will discuss now how to use this basis in the verification process. In practice, once the temporal requirements for the system under study are identified and extracted, the appropriate patterns for these

requirements are selected and instantiated with the suitable parameters, thus resulting in some SM observers. Each SM observer monitors an elementary requirement. The SM observers are then translated into TA observers. This transformation will be presented hereafter.

4.1 Transformation Idea

In spite of the number of automated analyzers developed for TA, these tools suffer from two main limitations: the first is that users must be familiar with their formal notations. The second is the lack of patterns for high-level system design (hierarchy notion namely). On the other hand, semi-formal languages, such as UML SM, are suitable for expressing system requirements. However, the automatic verification of these models is unfeasible directly. The temporal requirement verification approach that we propose takes advantage of the expression flexibility of SM and the analysis facilities offered by TA formalism.

The various rules of the transformation algorithm we have defined are expressed according to the **Model-Driven Architecture (MDA)** approach. MDA is an initiative and a standard proposed by the OMG, allowing developers to create systems entirely based on models. It points out the idea of separation of concerns by unlinking/uncoupling the application logic from the implementation platforms technology (Weis et al., 2003).

Figure 2 illustrates the use of the MDA four-layer metamodeling architecture for our transformation;

- The source model (resp. target model) is expressed according to the source metamodel (resp. target metamodel),
- The metamodels are defined and expressed according to the MOF metamodel (in our transformation, we used the Ecore metamodel, which is the Eclipse implementation of MOF),
- A metamodel is developed for TA. On the other side, we used the UML metamodel distributed in the Eclipse framework,
- All the rules are introduced at the metamodel level,
- The transformation takes a UML SM model as a source model and generates a TA model with a corresponding formatted code.

4.2 Time Annotations

Here we use SM as a modeling notation to take advantage of the flexibility they offer. However, since we strive to obtain an accurate specification, we should

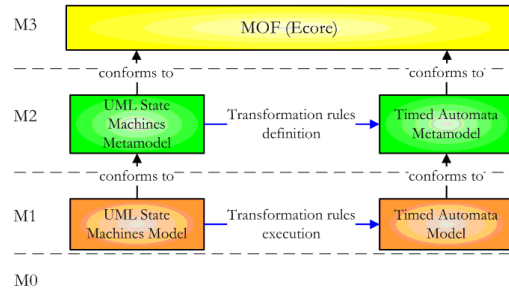


Figure 2: Transformation Approach.

guide the user while introducing the temporal constraints. Concretely, we propose a set of timed annotations in order to express the states' characteristics as well as the transition guards. Table 2 shows some examples of them and defines the signification of each annotation.

Table 2: Time annotations used.

	Time annotation	Signification
1	$at_most(T_{max})$	$t \leq T_{max}$
2	$at_least(T_{min})$	$t \geq T_{min}$
3	$after(d)$	$t = d$
4	$between(T_{min}, T_{max})$	$T_{min} \leq t \leq T_{max}$
5	$upper(T_{min})$	$t > T_{min}$
6	$lower(T_{max})$	$t < T_{max}$

4.3 Transformation Algorithm

One of the key parts of our method is the translation of UML SM with time annotations into TA. For sake of space, we will briefly describe the transformation rules while giving the source and target element for each of them in 3. For more details, the reader can refer to (Mekki et al., 2010).

Table 3: Transformation Rules.

Rule Name	Source element: UML SM	target element: TA
FromStateMachine	StateMachine	TA
Simple2Simple	State	State
Final2State	Final Pseudostate	State
OR2Automata	State	Automaton
AND2Automata	State	Automaton
Trans2Trans	Transition	Transition
Entry2State	EntryAction	State
Exit2State	ExitAction	State
Do2State	DoActivity	State

The main rule of this algorithm is *FromStateMachine* rule. This rule is the first one carried out by the transformation algorithm. It picks elements in the source model, then calls on other rules to translate the selected elements into TA elements in the tar-

get model. Likewise, the called rules behave in the same way; they select elements in the source model and call the appropriate rule for transforming them. For example, the *FromStateMachine* rule is applied to elements of type “UML::StateMachine” and translates them into a “TA::AutomataMachine” element. Also, different element types are selected and different rules are called on in this rule. First, the rule selects all the UML states. Then for each selected state, according to its type, the rule *Simple2Simple* or *OR2Automata* or *AND2Automata* is called on. Secondly, it translates the “UML::Transition” elements by invoking rule *Trans2Trans*. Also, this rule deals with another element type, the “UML::Pseudostate”, by invoking some other rules such as *Final2State*.

This internal transformation process is the same for all the rules; each rule transforms the source element into the target one. Then, it selects subelements of the source element and calls on the appropriate rule to transform them.

4.4 Verification Process

Once our observation patterns’ basis is implemented, we introduce a verification process based on this basis. This section will outline the global architecture of our approach. The architecture is depicted graphically in Figure 3 (Mekki et al., 2009).

Concretely, our approach is composed of four processes: first, temporal requirements for the system under study are identified and extracted. Second, the appropriate patterns for these extracted requirements are selected and instantiated with the suitable parameters. This second process results in some SM observers. Each SM observer corresponds to an elementary requirement. Third, the SM observers are translated into TA observers. In parallel and in the same way, the specification under study (SM model with time annotations) is abstracted and translated into a TA model. The translation from the UML SM to TA is performed using the MDA model transformation technique as shown previously.

Finally, the generated TA are synchronized with the formal system’s specification model (TA) to generate a global model holding both the system specification and the requirements’ monitoring. Thus, the verification task is reduced to an error-state reachability search on the obtained global model.

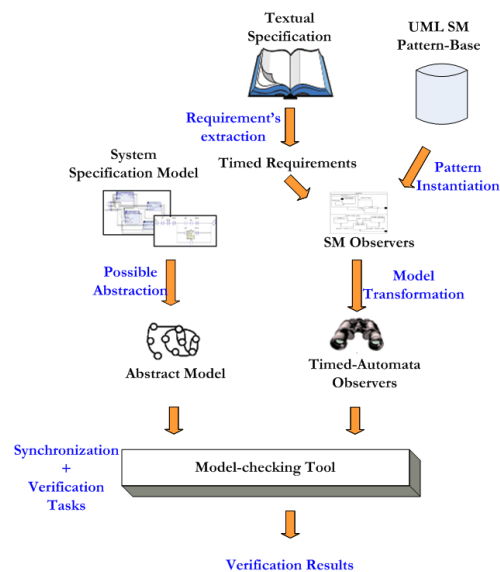


Figure 3: Method Global View.

5 CASE STUDY

5.1 Case Study Description

A classical automatic level crossing system is composed of several modules. The local **control** system which manages the traffic in the crossing area, a pair of barriers (**gate**), traffic **lights** whose role is to alert and prevent road users from entering the crossing zone and a **train**-sensing module which monitors trains approaching/leaving (Ghazel, 2009). The subsystems mentioned above execute in parallel and synchronize through events.

Several requirements * are given in textual specification in Figure 4. Next, using this textual specification, we will show how patterns are used to express and monitor requirements.

5.2 Using Patterns

For sake of space, only two requirements will be checked on the basis of the textual system specification. For each requirement, formalization is introduced using the defined generic template. First, a textual description of the requirement is presented, followed by an intuitive graphical representation in the shape of an UML SM (SM patterns). Then, a definition using our structured English grammar is given and finally a temporal logic expression is used to express formally the requirement. 1) The 1st require-

*<http://www.dagstuhl.de/fileadmin/redaktion/Programm/Seminar/07241/07241.CaseStudy.pdf>

1. "... As long as an approaching train runs over the activation sensor the sensor shall generate an occupied signal. When the last axle of the train has traversed the activation sensor it shall generate a free signal again. If the control unit receives an occupied signal from the activation sensor in the unsaved mode, it will enter the saving mode and gives the command to turn on the yellow lamp of the set of lights. **Three seconds after entering the saving mode the control unit has to give the command for switching off the yellow lamp and turning on the red lamp...** "
2. "... By entering saved mode the controller shall switch on the supervision signal to show signal aspect LCI, which means to turn on the blinking light. **Twelve seconds after the system has entered the safed mode it must start to lower the gate. The activity of lowering or raising the gate must not last longer than six seconds from one end position to the other. When the gate has reached the lower end position within the six seconds interval the control unit will enter the mode saved and gates closed...** "

Figure 4: Specification Example.

ments states that three seconds after receiving the *entering* signal, the controller should send command for *switching* lights. This requirement consists in a **punctuality** property (Figure 5) of 3 seconds between *entering* signal and *switch* signal.

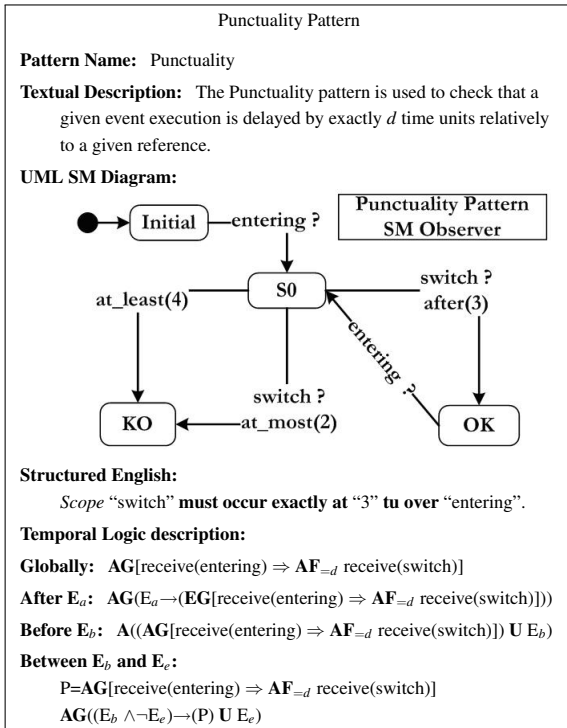


Figure 5: Punctuality Pattern.

2) The 2nd requirement states that the command signal *closed* should be detected at most 19 seconds after *entering* signal. This requirement consists in **maximumdelay** property (Figure 6) of 19 time units between *entering* and *closed* signals.

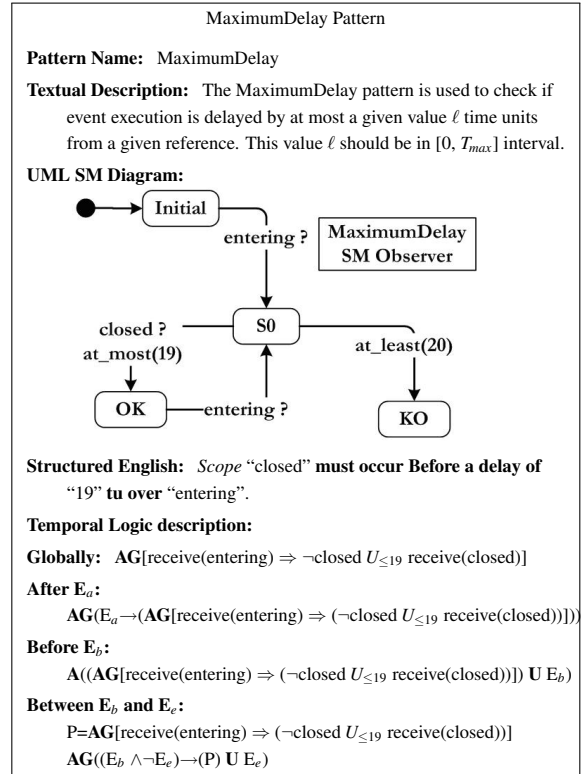


Figure 6: MaximumDelay Pattern.

5.3 Verification

The above step consists in instantiating the appropriate patterns in order to obtain observers for the extracted requirements. These observers are then translated into TA models to be synchronized with the system specification. The verification process consists in examining the reachability of the *KO*-states within the observers. The verification of our case study is carried out using the UPPAAL model checker.

6 CONCLUSIONS

In this paper, we have presented a model-based method applicable to formal specification and validation of time-constrained systems. The approach uses a set of observation patterns that we have established and which act as watch-dogs for the defined temporal requirements. Each pattern has been defined using a standard template we developed. Using patterns offers genericity and reusability.

On the other hand, we have developed a transformation algorithm to translate SM with time annotations into TA. The aim being to make a basis for the verification process. For this purpose, we have

introduced a TA metamodel using an extended definition of the original TA definition given by (Alur and Dill, 1994) (for briefness reasons, the TA metamodel description is omitted in this paper). Once the relationships (transformations rules) between UML SM metamodel elements and TA metamodel elements are defined, we expressed them in the **QVT** (Query/View/Transformation) language, defined by the OMG as the standard for the transformation phase. Then we used QVTo -an Eclipse Plugin- to run the algorithm.

Processing the verification upon the TA observers synchronized with the system specification reduces the verification task to a reachability analysis of the **KO**-nodes within the observers.

Validation of the model transformation algorithm we have developed is a key issue to ensure the correctness of our approach. Hence, a rigorous validation step is still needed where several properties should be checked such as; syntactic and behavioral equivalence, termination and confluence (Küster, 2006).

Based on the structured English grammar we have developed, a prototype tool which offers interesting facilities in terms of requirements specification and requirements consistency-check has been implemented. A subsequent step will be to extend this tool with a new module which automatically instantiates observers for the entered requirements using the observation patterns repository.

REFERENCES

- Alur, R. and Dill, D. (1994). A theory of timed automata. *Theoretical Computer Science*, 126:183–235.
- Dhaussy, P., Pillain, P.-Y., Creff, S., Raji, A., Traon, Y. L., and Baudry, B. (2009). Evaluating context descriptions and property definition patterns for software formal validation. In *12th ACM/IEEE International Conference, MODELS 2009*, Denver, Colorado, USA.
- Dong, J. S., Hao, P., Qin, S., Sun, J., and Yi, W. (2008). Timed automata patterns. *IEEE Transactions on Software Engineering*, 34(6):844–859.
- Dwyer, M. B., Avrunin, G. S., and Corbett, J. C. (1999). Patterns in property specifications for finite-state verification. In *In Proceedings of the 21st International Conference on Software Engineering*, pages 411–420.
- Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). *Design Patterns : Elements of Reusable Object-Oriented Software*. Addison Wesley.
- Ghazel, M. (2009). Using stochastic petri nets for level-crossing collision risk assessment. *IEEE Transaction on Intelligent Transportation Systems*, 10(4):668–677.
- Ghazel, M., Toguyeni, A., and Yim, P. (2009). State observer for DES under partial observation with time petri nets. *Journal of Discrete Event Dynamic Systems*, 19(2):137–165.
- Konrad, S. and Cheng, B. (2005). Real-time specification patterns. In *Proceedings of the 27th International Conference on Software Engineering (ICSE05)*, St Louis, MO, USA.
- Küster, J. M. (2006). Definition and validation of model transformations. *Software and System Modeling*, 5(3):233–259.
- Larsen, K., Pettersson, P., and Yi, W. (1997). Uppaal in a nutshell. *International Journal of Software Tools for Technology Transfer*, 1(1/2):134–152.
- Mekki, A., Ghazel, M., and Toguyeni, A. (2009). Validating time-constrained systems using uml statecharts patterns and timed automata observers. In *3rd International Workshop on Verification and Evaluation of Computer and Communication Systems Vecos09*, Rabat, Morocco.
- Mekki, A., Ghazel, M., and Toguyeni, A. (2010). Time-constrained systems validation using mda model transformation. In *Proceedings of the 8th ENIM IFAC International Conference of Modeling and Simulation*, Hammamet, Tunisia.
- Nascimento, F., Oliveira, M., and Wagner, F. (2009). Formal verification for embedded systems design based on mde. In *IESS'09 - International Embedded Systems Symposium*, Friedrichshafen, Germany.
- Schmidt, D. C. (2004). Model driven engineering. *IEEE Computer*, 23(2):25–31.
- UML (2009). *Unified Modeling Language Specification, Version 2.2*. OMG.
- Weis, T., Ulbrich, A., and Geihs, K. (2003). Model metamorphosis. *IEEE Software, IEEE Computer Society*, 20(5):46–51.
- Yovine, S. (1997). Kronos: a verification tool for real-time systems. *International Journal of Software Tools for Technology Transfer*, 1(1/2):123–133.

A PASSIVITY-BASED APPROACH TO DEPLOYMENT IN MULTI-AGENT NETWORKS

Heath LeBlanc, Emeka Eyisi, Nicholas Kottenstette, Xenofon Koutsoukos and Janos Sztipanovits
Institute for Software Integrated Systems (ISIS), Vanderbilt University, 2015 Terrace Place, Nashville, TN 37203, U.S.A.
{heath.j.leblanc, emeka.p.eyisi}@vanderbilt.edu, nkottens@isis.vanderbilt.edu
{xenofon.koutsoukos, janos.sztipanovits}@vanderbilt.edu

Keywords: Passivity, Compositionality, Deployment, Overlay network, Input-output stability, Distributed protocol.

Abstract: Surveillance and convoy tracking applications often require groups of networked agents for redundancy and better coverage. An important goal upon deployment is to establish a formation around a target. Although there exist distributed algorithms using only local communication that achieve this goal, they typically ignore destabilizing effects resulting from implementation uncertainties, such as network delays and data loss. This paper resolves these issues by introducing a discrete-time distributed design framework that uses a compositional, passivity-based approach to ensure l_2^m -stability regardless of overlay network topology, in the presence of network delays and data loss. For the restricted case of a uniform node degree in the overlay network topology, the paper shows that asymptotic formation establishment is achieved. Finally, simulations of velocity-limited unmanned air vehicles (UAVs) are presented that demonstrate the robustness of the network architecture to network delays and data loss.

1 INTRODUCTION

Modern surveillance and convoy tracking applications often require deploying groups of unmanned aerial vehicles (UAVs). The benefit of using multiple UAVs is redundancy, which reduces the likelihood of missing interesting events on the ground, in the presence of obstructions caused by nonuniform terrain, vegetation, or man-made structures. Further, the additional UAVs provide greater breadth of coverage. A central task for such multi-agent systems is to establish a formation around an area of interest. For example, an n -gon with a target as its center, at the appropriate radius, may simultaneously provide significant redundancy and breadth of coverage.

Performing coordinated tasks in multi-agent systems using only local information has been studied extensively over the past decade (Olfati-Saber et al., 2007), (Ren et al., 2005), (Olfati-Saber, 2006). Typically, in group coordination the desired formation emerges from the design of the control law. In (Fax and Murray, 2004), the so-called information filter is used for formation stability of LTI systems. For coordination of nonlinear systems, contraction theory with wave variable communication (Wang and Slotine, 2006), explicit design of Lyapunov vector fields (Lawrence et al., 2008),

and passivity (Arcak, 2007), (Ihle et al., 2007), (Bai et al., 2008), (Igarashi et al., 2008), have been used successfully.

Much of the above work - especially the passivity-based methods - has considered continuous-time systems; however, for implementation discrete-time design is needed. In addition, implementation uncertainties such as network delays and data loss must be taken into consideration. This paper focuses on decoupling the control design and discrete-time implementation by using a passivity-based framework inspired by work in telemanipulation (Chopra et al., 2008), port-Hamiltonian systems (Stramigioli et al., 2005), and network control (Kottenstette et al., 2009).

The unifying concept in the aforementioned work is the scattering formalism, which has traditionally been applied to power variables (effort and flow) while closing the loop on velocity. In this work, the scattering formalism is used abstractly (without the physical interpretation) to close the loop on position.

The contributions of this paper are three-fold. First, we introduce a compositional network control system (NCS) design approach that guarantees passivity of the networked system. Secondly, we show that the coupled multi-agent network is l_2^m -stable for any bidirectional overlay network with asymmetric

delays whenever the input-output mapping of each agent is strictly-output passive. The stability result holds for packet-switched networks using easily enforced constraints. Thirdly, for the single-input, single-output (SISO) case, we perform steady-state analysis and we show that the multi-agent network can establish an n -gon upon deployment. Finally, we provide simulations using Simulink/TrueTime to illustrate the approach for controlling velocity-limited quadrotor UAVs. Simulink is a graphical user environment (GUI) used for the modeling, simulation, and analysis of dynamical systems (MathWorks, 2008). TrueTime extends Simulink with platform related modeling concepts (i.e., networks, clocks, schedulers) and supports simulation of networked and embedded control systems with implementation effects (Ohlin et al., 2007).

The rest of the paper is organized as follows: Section 2 provides the formal problem statement and other preliminaries. The distributed NCS design framework is introduced in Section 3. The main theoretical results are detailed in Section 4. Section 5 presents simulations in Simulink/TrueTime illustrating our results. Finally, Section 6 provides concluding remarks and future work.

2 PRELIMINARIES

Consider the problem of n agents establishing a formation around a target in \mathbb{R}^2 . Assume a global inertial coordinate system and suppose the starting positions of the agents are arbitrary. The goal is to establish an n -gon, where the n agents tend to the coordinates of the vertices asymptotically. Formally, we assign a vertex v_i of the n -gon to agent i , with position $x_i(k)$, $i = 1, 2, \dots, n$. Then we require

$$\lim_{k \rightarrow \infty} \|x_i(k) - v_i\|_2 = 0. \quad (1)$$

We consider a network of n interacting agents with communication topology described by a connected undirected graph, $G = (V, E)$, where $V = \{1, 2, \dots, n\}$ describes the agents and $E \subset V \times V$ models the bidirectional communication. Additionally, each bidirectional link may have asymmetric, time-varying delays. The delays are denoted $d_{ij}(k)$ for link $(i, j) \in E$.

For the purpose of analysis, it is useful to introduce the *adjacency matrix*, $A = [a_{ij}]$, associated with graph G (Godsil and Royle, 2001). For an undirected graph, the adjacency matrix is a symmetric matrix (i.e., $A = A^T$), and is mathematically defined by

$$a_{ij} = \begin{cases} 1 & (i, j) \in E; \\ 0 & (i, j) \notin E. \end{cases} \quad (2)$$

Additionally, we define the set of *neighbors*, N_i , of a node i as those nodes which send messages to i , given by $N_i = \{j \in V \mid a_{ji} \neq 0\}$. Finally, we denote the number of neighbors by $|N_i| = n_i$.

The agents communicate and process signals in the extended l_2 -space of functions that map $\mathbb{N} \cup \{0\}$ to \mathbb{R}^m , denoted l_{2e}^m , which are mapped onto l_2^m by the truncation operator defined by

$$(f)_N = \begin{cases} f(k) & 0 \leq k \leq N-1; \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

Further, for all $f, g \in l_{2e}^m$ define

$$\langle f, g \rangle_N \triangleq \sum_{k=0}^{N-1} f^T(k)g(k). \quad (4)$$

We use definitions for l_2^m -stability and passivity for discrete-time systems, which are analogous to the continuous-time counterparts in (van der Schaft, 1999):

Definition 1. *Given a discrete-time system defined by its input-output mapping, $G: l_{2e}^m \rightarrow l_{2e}^m$, the discrete-time system is l_2^m -stable if*

$$u \in l_2^m \implies G(u) \in l_2^m. \quad (5)$$

Definition 2. *Let $G: l_{2e}^m \rightarrow l_{2e}^m$. Then, for all $u \in l_{2e}^m$:*

1. *G is passive if there exists some constant $\beta \in \mathbb{R}$ (called the bias) such that*

$$\langle G(u), u \rangle_N \geq -\beta, \quad \forall N \in \mathbb{N}; \quad (6)$$

2. *G is strictly output passive if there exists some constants $\beta \in \mathbb{R}$ and $\varepsilon > 0$ such that*

$$\langle G(u), u \rangle_N \geq \varepsilon \| (G(u))_N \|_2^2 - \beta, \quad \forall N \in \mathbb{N}. \quad (7)$$

We assume a synchronous network, with period T .¹ Further, each agent shares information only locally (no global shared resources). However, the desired setpoints are calculated prior to deployment. Finally, the agents begin execution at time index $k = 0$.

3 NCS DESIGN

This section details the distributed network control system (NCS) design. The objective is to provide a passive-by-construction, discrete-time multi-agent network. In general, the overlay network is bidirectional with asymmetric delays. For simplicity, consider the three node network shown in Figure 1.

¹We assume the agents use a clock synchronization algorithm prior to deployment to ensure this assumption holds.

Each node represents a UAV, with each edge modeling the communication between UAVs. Realistically, each link in the network is subject to delay imposed by packet handling and transmission delays. This is modeled by the time-varying delays (e.g., $d_{ij}(k)$), shown in Figure 1. The u and v variables in the figure are power wave variables, which are described in Section 3.2.

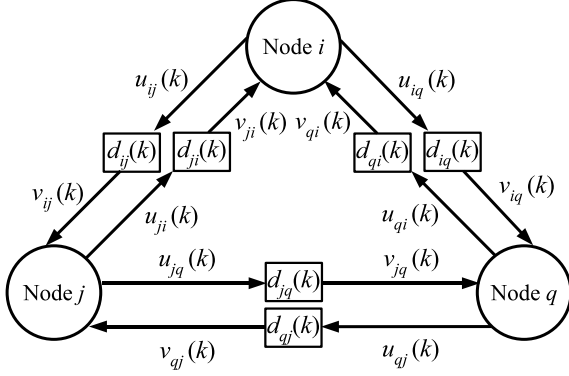


Figure 1: A three node network with time-varying delays in the communication links.

3.1 Agent Model

The agent model is shown in Figure 2. Each agent i receives an input reference, r_i , which influences the output, y_i , of the agent through the system mapping, H_i . H_i describes a compensated plant, and is required to be strictly output passive. The variables x_i and y_i are transformed into the wave domain through the scattering transformation. The node's wave variables u_{ii} and v_{ii} are coupled to other nodes through a power junction, PJ_i , which allows two or more systems to be connected in a passivity-preserving manner (Kottenstette et al., 2009). The scattering transformation and power junction are crucial to ensuring passivity of the networked system and will be described in the next section.

For simplicity, we model the UAVs with a point mass along two dimensions. We denote the point mass system, $H_p: f_I \rightarrow y_I$, in which $f_I \in \mathbb{R}^2$ is the inertial control force and $y_I \in \mathbb{R}^2$ is the inertial position as depicted in Figure 3. The equations of motion are

$$\begin{aligned} \dot{y}_I(t) &= v_I(t) \\ M\dot{v}_I(t) &= f_I(t). \end{aligned}$$

Using the point mass model for each agent i , we design an inertial position control system, which we denote $H_I: e_i \rightarrow y_I$, shown in Figure 3. The inner loop gain of the compensator is $\omega_c M$ ($\omega_c > 0$) and the outer

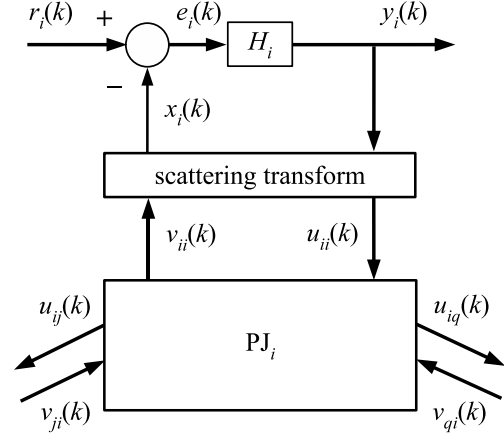


Figure 2: Node architecture.

loop gain $\frac{\omega_c}{2}$. The overall equation of motion

$$\ddot{y}_I = -\omega_c \dot{y}_I - \frac{\omega_c^2}{2} (y_I - e_i) = -2\zeta\omega_n \dot{y}_I - \omega_n^2 (y_I - e_i)$$

clearly indicates a stable second order system with natural frequency $\omega_n = \frac{\omega_c}{\sqrt{2}}$ and damping coefficient $\zeta = \frac{1}{\sqrt{2}}$, where $y_I = e_i$ at steady state. It can be shown that the inertial position control system is inside the sector $[a, 1]$, where $a = -\frac{1}{2(1+\sqrt{2})}$ (Zames, 1966), (Kottenstette and Porter, 2009). Therefore, the system $H_I: e_i \rightarrow y_I$ is not strictly output passive; however, by adding a high-pass filter in parallel, the system may be rendered strictly output passive, as depicted in Figure 4 (with $c = 2$). Since $e_i = y_I$ at steady state, the inertial position of the system may be directly controlled. This model is discretized using a bilinear-like transform, called the inner-product equivalent sample and hold (IPESH) transform, which preserves the conic properties of the system (Kottenstette et al., 2009).

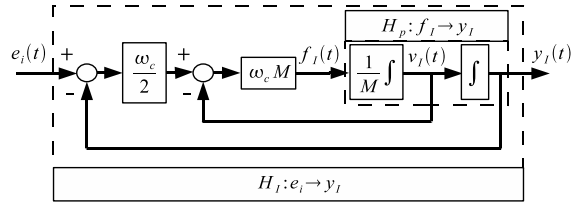


Figure 3: Inertial position control system.

3.2 Network Model

In distributed control applications the information transmitted across the network has inherent physical meaning. It is well known that transforming these physical variables into the wave domain can preserve

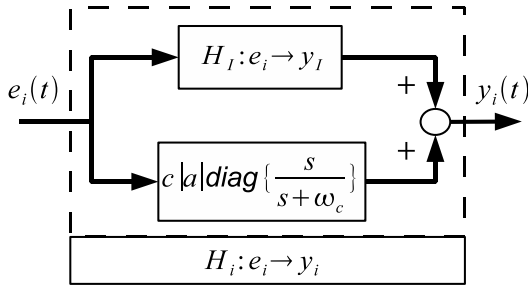


Figure 4: Strictly output passive inertial position control system.

passivity and stability for a single bidirectional connection (Chopra et al., 2008) and for star networks (Kottenstette et al., 2009). In this paper, we extend these approaches to distributed networks with arbitrary overlay topology. The network model is distributed in the sense that all nodes in the network communicate only locally.

We formally define the scattering transformation as follows. For each $i \in V$, the *scattering transformation* produces power waves $u_{ii}(k)$ and $v_{ii}(k)$ defined by

$$u_{ii}(k) = \frac{1}{\sqrt{2b_i}}(b_i y_i(k) + x_i(k)), \quad (9a)$$

$$v_{ii}(k) = \frac{1}{\sqrt{2b_i}}(b_i y_i(k) - x_i(k)). \quad (9b)$$

This definition is similar to the one in (Niemeyer and Slotine, 2004), with the force and velocity variables replaced with x_i and y_i . In general, we place no restriction on the physical meaning of x_i and y_i ; however, for our UAV model, x_i and y_i denote position. The scattering transformation is treated as a mathematical definition, with the characteristic impedance, b_i , having appropriate units for physical consistency.

Next, we define the power junction, which allows two or more systems to be connected in the wave domain in a passivity-preserving manner.

Definition 3. Fix $m, p \in \mathbb{N}, p \geq 2$. Then, a *power junction* is a function $f: l_{2e}^{mp} \rightarrow l_{2e}^{mp}$, which satisfies for all $\xi \in l_{2e}^{mp}$ and all $k \in \mathbb{Z}^+$ the inequality

$$\xi^T(k)\xi(k) \geq f(\xi(k))^T f(\xi(k)). \quad (10)$$

The vector $\xi(k)$ in the definition of the power junction is formed by concatenating the p inputs in l_{2e}^m into a single mp -dimensional column vector. For analyzing our network model, it is useful to pair the p inputs to their corresponding outputs in the output column vector, $f(\xi(k))$, and partition the set of pairs into two disjoint sets S_{in} and S_{out} . These sets denote the net flow of power into and out of the power junction, respectively. Formally, for $i \in S_{in}$ and $o \in S_{out}$, let

$u_i, v_o \in l_{2e}^m$ denote the inputs and $v_i, u_o \in l_{2e}^m$ denote the outputs of the power junction. Then (10) may be rewritten as

$$\sum_{i \in S_{in}} u_i^T(k) u_i(k) - v_i^T(k) v_i(k) \geq \sum_{o \in S_{out}} u_o^T(k) u_o(k) - v_o^T(k) v_o(k). \quad (11)$$

We implement each node's power junction as a linear set of equations. Specifically, we use the following equations. For each $i \in V$, $j \in N_i$, and $k \in \mathbb{Z}^+$, the outgoing waves are computed as

$$u_{ij}(k) = \frac{1}{\sqrt{n_i}} u_{ii}(k), \quad (12a)$$

$$v_{ii}(k) = \frac{1}{\sqrt{n_i}} \sum_{j \in N_i} v_{ji}(k). \quad (12b)$$

Although the functional form of the power junction is not constrained to be linear, these equations simplify the steady state analysis and exhibit a local averaging behavior in regular networks. This can be seen as follows. Consider the wave variables that influence the power junction at a given node i , shown in Figure 5, and suppose $n_i = n_j \equiv \eta$, $\forall i, j \in V$ (i.e., a regular network). Then, for each $j \in N_i$, $v_{ji}(k) = u_{ji}(k - d_{ji}(k))$. Thus, an expression for $v_{ii}(k)$ is given by

$$\begin{aligned} v_{ii}(k) &= \frac{1}{\sqrt{\eta}} \sum_{j \in N_i} v_{ji}(k) \\ &= \frac{1}{\sqrt{\eta}} \sum_{j \in N_i} u_{ji}(k - d_{ji}(k)) \\ &= \frac{1}{\sqrt{\eta}} \sum_{j \in N_i} \frac{1}{\sqrt{\eta}} u_{jj}(k - d_{ji}(k)) \\ &= \frac{1}{\eta} \sum_{j \in N_i} u_{jj}(k - d_{ji}(k)). \end{aligned}$$

Therefore, in regular networks, the input wave variable, $v_{ii}(k)$, is the average of its neighbors' delayed output wave variables, $u_{jj}(k - d_{ji}(k))$, $j \in N_i$.

Due to the presence of delays and data loss some (or all) of the $v_{ji}(k)$ may not be received at time k , in which case $v_{ji}(k) \triangleq 0$. Handling delayed and dropped packets as null packets satisfies the synchronous assumption and preserves passivity (Chopra et al., 2008). Before proceeding to describe the constraints on delayed and lost data, we prove our claim that the implementation given by (12) satisfies the definition of a power junction.

Lemma 1. The implementation defined by (12) satisfies the power junction constraint.

Proof: From the remarks following the power junction definition, it is sufficient to show that (12) satisfy (11). Clearly, a sufficient condition for satisfying (11) is to enforce the following constraints for

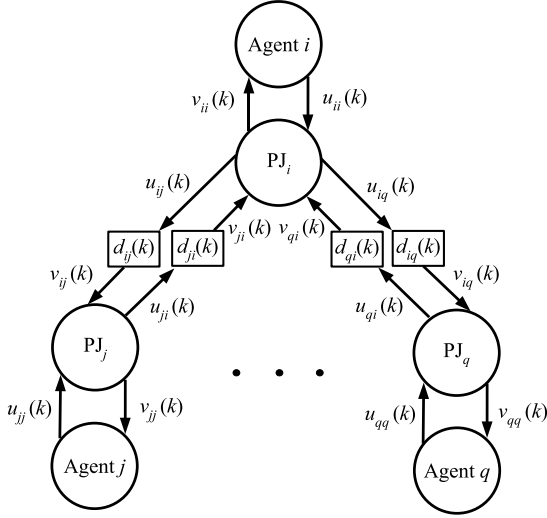


Figure 5: The neighbors of a node, showing the wave variables influencing the node through the power junction.

each component $l = 1, 2, \dots, m$,

$$\sum_{j \in N_i} u_{ij}^2(k) \leq u_{ii}^2(k), \quad (13a)$$

$$v_{ii}^2(k) \leq \sum_{j \in N_i} v_{ji}^2(k), \quad (13b)$$

where $S_{in} = \{ii\}$ and $S_{out} = \{ij \mid j \in N_i\}$. To show that (13a) is satisfied, we use (12a), which yields

$$\sum_{j \in N_i} u_{ij}^2(k) = \sum_{j \in N_i} \frac{1}{n_i} u_{ii}^2(k) = u_{ii}^2(k).$$

For (13b) we combine (12b) with the Cauchy-Schwartz inequality to get

$$v_{ii}^2(k) = \frac{1}{n_i} \left(\sum_{j \in N_i} v_{ji}(k) \right)^2 \leq \sum_{j \in N_i} v_{ji}^2(k).$$

□

Finally, we constrain the network model by preventing retransmission of data for each agent. Also, as mentioned above, whenever receiver's buffers are empty, we process null packets. Based on these assumptions, each channel $(i, j) \in E$ satisfies the following inequality regardless of time-varying delays and data loss (Chopra et al., 2008),

$$\|(v_{ij})_N\|_2^2 \leq \|(u_{ij})_N\|_2^2, \text{ holds } \forall N \in \mathbb{N}. \quad (14)$$

This inequality states that each channel, viewed as the input-output mapping shown in Figure 6, is passive.

4 ANALYSIS

4.1 Passivity of the Networked System

In this section we first prove that the network model is passive and then show that the input-output map-

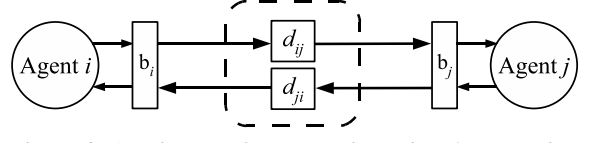


Figure 6: A point-to-point connection using the scattering formalism to ensure passivity of the bidirectional connection subject to asymmetric time-varying delays, shown inside the dashed box.

ping describing the networked system is strictly output passive. Figure 7 shows the passive network. The following lemma proves that the portion inside the dashed box of Figure 7 is passive.

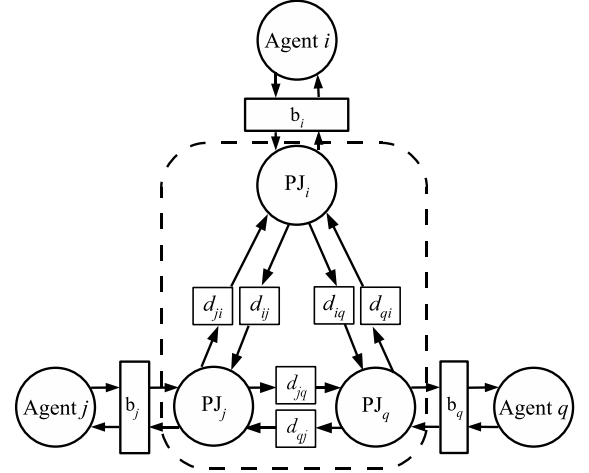


Figure 7: A three node network illustrating the passive network, shown inside the dashed box.

Lemma 2. Consider a network of n interacting dynamic systems constrained to the design framework described in Section 3. Then, the global energy constraint

$$\sum_{i=1}^n \{ \|(u_{ii})_N\|_2^2 - \|(v_{ii})_N\|_2^2 \} \geq 0 \quad (15)$$

is satisfied for all $N \in \mathbb{N}$, regardless of time-varying delays and data loss.

Proof: Sum the power constraints (11) of each node i , with $S_{in_i} = \{ii\}$ and $S_{out_i} = \{ij \mid j \in N_i\}$, from time $k = 0$ to $k = N - 1$ and sum the resulting inequalities over all nodes (rearranging the terms in the sums appropriately). Then, invoke (14) to obtain

$$\begin{aligned} & \sum_{i=1}^n \{ \|(u_{ii})_N\|_2^2 - \|(v_{ii})_N\|_2^2 \} \\ & \geq \sum_{i=1}^n \sum_{j \in N_i} a_{ij} \{ \|(u_{ij})_N\|_2^2 - \|(v_{ij})_N\|_2^2 \} \\ & \geq 0. \end{aligned}$$

□

The energy constraint of (15) also implies that collectively, the mapping from the x_i to the y_i , $i = 1, \dots, n$, is passive (see Figure 2). To show this, consider the following power constraint, which may easily be derived from (9a) and (9b)

$$\frac{1}{2}(u_{ii}^T(k)u_{ii}(k) - v_{ii}^T(k)v_{ii}(k)) = y_i^T(k)x_i(k). \quad (16)$$

Substitute (16) into (15) to obtain

$$\sum_{i=1}^n \langle y_i, x_i \rangle_N \geq 0. \quad (17)$$

Define $x(k)$ and $y(k)$ as the $nm \times 1$ column vectors formed by concatenating the $x_i(k)$ and $y_i(k)$, respectively, of each node. Then, it follows that

$$\langle y, x \rangle_N \geq 0,$$

which satisfies the definition of passivity (6), with $\beta = 0$.

We conclude the section by proving that the entire networked system (e.g., the three node system in Figure 7) is strictly output passive for arbitrary network topologies.

Theorem 1. *Consider a network of n interacting dynamic systems constrained to the design framework described in Section 3. Define $r(k)$ and $y(k)$ as the $nm \times 1$ column vectors formed by concatenating the $r_i(k)$ and $y_i(k)$, respectively, of each node. Finally, define the input-output mapping $H : l_{2e}^{nm} \rightarrow l_{2e}^{nm}$ such that $H(r(k)) = y(k)$. Then, H is strictly output passive.*

Proof: Since each H_i is strictly output passive, there exists $\varepsilon_i > 0$ and β_i , for all $i \in V$, such that

$$\langle y_i, e_i \rangle_N \geq \varepsilon_i \| (y_i)_N \|_2^2 - \beta_i. \quad (18)$$

Making the substitution, $x_i(k) = r_i(k) - e_i(k)$ into (17) and using the linearity of the inner-product, gives

$$\sum_{i=1}^n \langle y_i, r_i \rangle_N \geq \sum_{i=1}^n \langle y_i, e_i \rangle_N. \quad (19)$$

Substituting (18) into (19) yields

$$\sum_{i=1}^n \langle y_i, r_i \rangle_N \geq \varepsilon \sum_{i=1}^n \| (y_i)_N \|_2^2 - \beta, \quad (20)$$

where $\varepsilon = \min_i \{\varepsilon_i\}$ and $\beta = \sum_{i=1}^n \beta_i$. Finally, we rewrite (20) as

$$\langle y, r \rangle_N \geq \varepsilon \| (y)_N \|_2^2 - \beta. \quad (21)$$

□

4.2 Stability

The previous result shows that the networked system defined by the mapping H is strictly output passive. It then follows that H is l_2^m -stable.

Theorem 2. *The mapping $H(r(k)) = y(k)$ defined in Theorem 1 is l_2^m -stable.*

Proof: We begin with the notion of finite l_2^m -gain. The map G has finite l_2^m -gain if there exists finite constants γ, β such that for all $N \in \mathbb{N}$

$$\| (G(u))_N \|_2 \leq \gamma \| (u)_N \|_2 + \beta, \quad \forall u \in l_{2e}^m. \quad (22)$$

It is well known in continuous-time (van der Schaft, 1999) and has been shown for discrete-time (Kottenstette and Antsaklis, 2007) that a sufficient condition for a system to have finite l_2^m -gain is for the system to be strictly output passive. Therefore, by Theorem 1, H has finite l_2^m -gain.

Now suppose $u \in l_2^m$ (i.e., $\|u\|_2 < \infty$). Then take $N \rightarrow \infty$ in (22). This leads to

$$\|G(u)\|_2 \leq \gamma \|u\|_2 + \beta < \infty, \quad \forall u \in l_2^m.$$

Therefore, $H(u) \in l_2^m$. By Definition 1, H is l_2^m -stable. □

From the proof of Theorem 2, we see that any system that is strictly output passive is necessarily l_2^m -stable. Therefore, each agent described by H_i is inherently stable. The benefit of the passivity-based network framework is that it ensures that interactions caused by the network do not destabilize the networked multi-agent system. This result holds even in the presence of time-varying delays and data loss (under the assumptions outlined in Section 3.2) because the passivity results hold. Moreover, the networked multi-agent system will remain stable regardless of network topology.

4.3 Steady-state Analysis

To analyze the behavior of the coupled multi-agent system, we consider the system at steady-state. In order to do this, we assume that each strictly output system, H_i , admits a steady-state solution whenever a constant input is applied. With this assumption, there exists a steady-state solution for the multi-agent system (provided there is no data loss), since the rest of the networked system is linear. For simplicity, we assume the system is SISO. If the degrees of freedom of the system are decoupled, this result may be applied to MIMO systems.

Theorem 3. *Consider a network of n interacting SISO agents designed using the framework described in Section 3 and ignore time delays and data loss. Assume the inputs, r_i , reach steady-state and consider*

the outputs, y_i , as $k \rightarrow \infty$. If H_i at each node i has steady-state gain g_i , then the steady-state output of node i is given by

$$y_i = \frac{g_i}{b_i g_i + 1} \left[r_i + \frac{\sqrt{2b_i}}{\sqrt{n_i}} \sum_{j \in N_i} \frac{1}{\sqrt{2b_j n_j}} \left[\frac{b_j g_j - 1}{g_j} y_j + r_j \right] \right] \quad (23)$$

Proof: Since time delays and data loss are ignored, we drop the time index. Using the relation $e_i = r_i - x_i$ and replacing H_i with g_i , the input-output relation $y_i = H_i(e_i)$ may be written as

$$y_i = g_i(r_i - x_i). \quad (24)$$

Next, substituting (24) into (9b) and solving for x_i yields

$$x_i = \frac{-\sqrt{2b_i}}{b_i g_i + 1} v_{ii} + \frac{b_i g_i}{b_i g_i + 1} r_i. \quad (25)$$

Substituting (25) into (24) and reducing gives us

$$y_i = \frac{g_i}{b_i g_i + 1} r_i + \frac{\sqrt{2b_i} g_i}{b_i g_i + 1} v_{ii}. \quad (26)$$

Combining $v_{ji} = u_{ji}$ with (12a) at node j (roles of j and i are reversed), produces

$$v_{ji} = \frac{1}{\sqrt{n_j}} u_{jj}.$$

Substituting this into (12b) for node i yields

$$v_{ii} = \frac{1}{\sqrt{n_i}} \sum_{j \in N_i} \frac{1}{\sqrt{n_j}} u_{jj}. \quad (27)$$

Now, solving (24) at node j for x_j and substituting into (9a) at node j produces

$$u_{jj} = \frac{1}{\sqrt{2b_j}} \left(\frac{b_j g_j - 1}{g_j} y_j + r_j \right) \quad (28)$$

Substitute (28) into (27) to get

$$v_{ii} = \frac{1}{\sqrt{n_i}} \sum_{j \in N_i} \frac{1}{\sqrt{2b_j n_j}} \left(\frac{b_j g_j - 1}{g_j} y_j + r_j \right) \quad (29)$$

Finally, substitute (29) into (26) to obtain (23). \square

Theorem 3 provides a system of n equations describing the system asymptotically (as $k \rightarrow \infty$). The system of equations described by (23) are clearly coupled and depend on the overlay network structure. For the case of a regular topology, the following corollary characterizes the system of equations and provides the means to precalculate the reference inputs to asymptotically achieve a desired setpoint. For the two-dimensional agent model described in Section 3.1 the two degrees of freedom are decoupled, so we use this corollary to establish an n -gon around the target, as described in Section 2.

Corollary 1. Consider a network of n SISO agents with a regular overlay network topology (i.e., $n_i =$

$n_j \equiv \eta \forall i, j \in V$). If all of the systems H_i have identical steady-state gain g and each scattering transformation has the same impedance b , the system of steady-state equations may be written as

$$y = \frac{g}{bg+1} \left(r + \frac{1}{\eta} A \left[\frac{bg-1}{g} y + r \right] \right), \quad (30)$$

where y and r are defined in Theorem 1 and A is the adjacency matrix of the regular overlay network topology. Assuming the inverse of $(\eta I + A)$ exists, we may solve this equation for r to obtain

$$r = \frac{1}{g} (\eta I + A)^{-1} ((bg+1)\eta I - (bg-1)A)y. \quad (31)$$

5 SIMULATIONS

The experimental setup involves a network of eight UAVs that communicate in a regular overlay network topology, each with degree $\eta = 4$, and a synchronous sampling period of $T = 0.01$ seconds. Each UAV moves in the plane, influenced by its own input and the wave variables received from its neighbors. We model the UAVs as described in Section 3.1, so that each has a steady state gain, $g = 1$, and characteristic impedance, $b = 1$. The dynamics of the velocity limited UAVs are implemented using Simulink models while TrueTime is used to simulate the network dynamics and communication between neighboring UAVs. The network protocol used is IEEE 802.11b, with a speed of 11 Mbps.

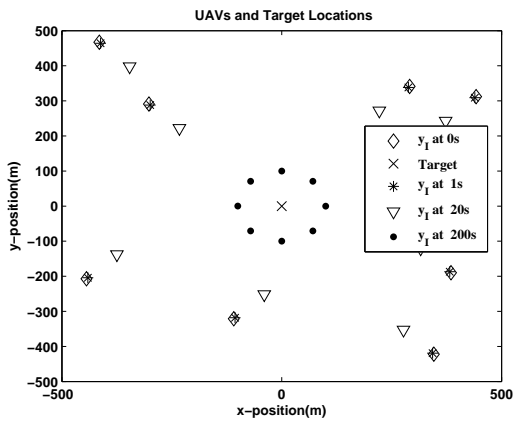
5.1 Evaluation

We present five scenarios to demonstrate our design framework.

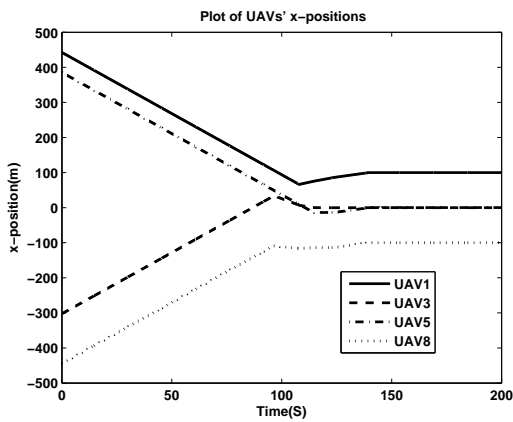
Experiment 1: Nominal Case. In this experiment, delays and data loss are ignored. Figure 8a shows a plot of the eight UAVs enclosing the target at the origin, within a radius of 100m. The data points show the evolution of the UAVs from their initial positions to their final positions. The UAVs cooperatively enclose the target and each agent is 100m away from the target, thus achieving the desired goal.

Figure 8b shows the x-positions of four agents (UAVs 1, 3, 5, and 8) over a period of 200 seconds. The desired x-positions for the four agents are 100m, 0m, 0m and -100m, respectively. These values correspond to the desired configuration and are achieved in about 160 seconds.

Figure 9 shows the average and maximum errors of all the UAVs' positions relative to the desired configuration. From the figure, the average and maximum error reach the value of zero after 160 seconds



(a) UAVs and target positions.



(b) Plot of UAVs' x-position over time.

Figure 8: Network of UAVs (nominal case).

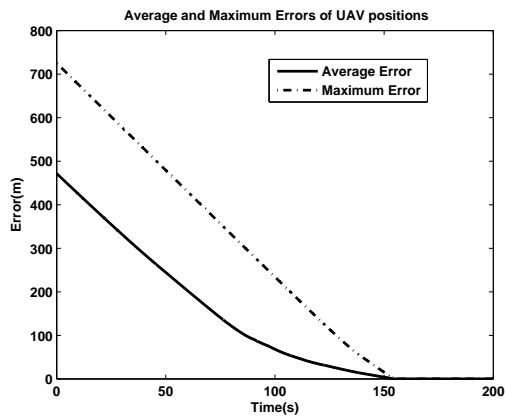


Figure 9: Average and maximum errors of UAVs' positions (nominal case).

which corresponds to the time the UAVs achieve the desired configuration.

Experiment 2: Nonuniform Constant Time Delays. This experiment demonstrates the robustness of

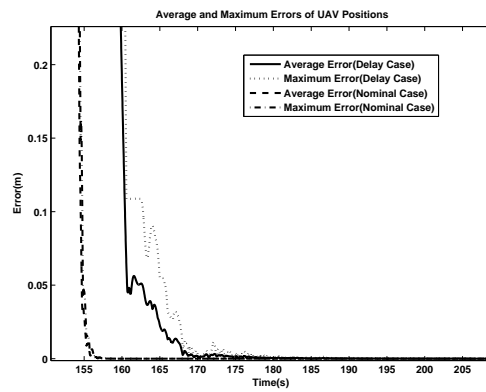


Figure 10: Average and maximum errors of UAVs' positions (nonuniform constant delay in all communication channels).

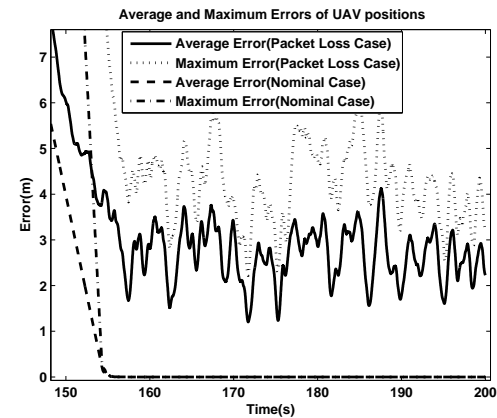


Figure 11: Average and maximum errors of UAVs' positions (ten percent probability of packet loss).

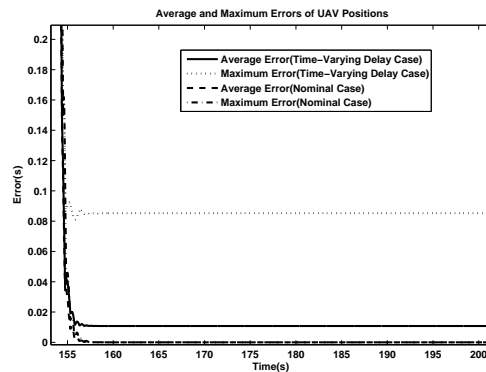


Figure 12: Average and maximum errors of UAVs' positions (time-varying delay case).

the distributed network of UAVs to nonuniform constant delays. We introduce nonuniform time delays, between 1 to 10 seconds, in all the communication channels of the network. Figure 10 shows the average

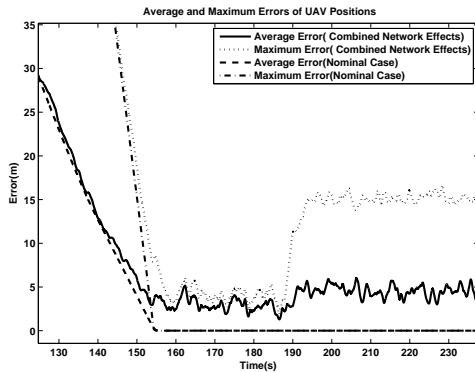


Figure 13: Average and maximum errors of UAVs' positions (time-varying delay and packet loss case).

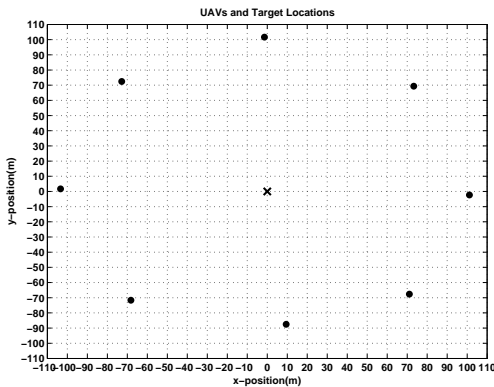


Figure 14: UAVs and target positions (time-varying delay and packet loss case).

and maximum errors, comparing the nominal case to the case with nonuniform constant delays. From the figure, the average and maximum errors for the delay case reach the value of zero after 200 seconds, taking about 40 seconds more time than in the nominal case to reach the desired configuration. The presence of time delays in the network does not prevent the agents from reaching the desired configuration; however, the time delays increase the time it takes to reach the desired configuration.

Experiment 3: Data Dropouts. This experiment demonstrates the effect of packet loss on the behavior of the UAVs. A probabilistic model is used to implement the loss of packets in the channels. For our studies, we simulate the case of a ten percent probability of packet loss. Figure 11 shows the average and maximum errors, comparing the nominal case to the case of ten percent probability of packet loss. The plot shows that even with ten percent packet loss, the UAVs still manage to come very close to the desired configuration, demonstrating the resilience of the net-

work. Due to the packet loss, the UAVs will never reach a steady state; however, the UAVs' positions end up within a maximum error of 6 meters and an average error of 4 meters of the desired configuration.

Experiment 4: Time-varying Delays. This experiment demonstrates the effect of time-varying delays on the behavior of the UAVs. To simulate the case of time-varying delays, we incorporate a disturbance node in the network. The sampling period of the disturbance node is set to a value of 0.05 seconds, and the disturbance node floods the network with disturbance packets based on a Bernoulli process with parameter d . The disturbance node samples a uniformly distributed random variable $X[k] \in [0, 1]$ every 0.05 seconds. If $X[k] > d$, a disturbance packet is forced on the network. Figure 12 shows the average and maximum errors, comparing the nominal case to the time-varying delay case, with $d = 0.5$. The plot shows that in the presence of time-varying delays, the UAVs remain stable and settle within a maximum error of 0.09 meters and an average error of 0.02 meters from the desired configuration.

Experiment 5: Combined Network effects. The experiment demonstrates the combined effects of ten percent packet loss and time-varying delays on the behavior of the UAVs. This experiment studies the combined network effects of time-varying delays and data loss in order to simulate the real world dynamics of the network. Again, the time-varying delays are introduced through the disturbance node with $d = 0.5$. Figure 13 shows the average and maximum errors, comparing the nominal case to the case with the combined network effects. The figure shows that the average and maximum errors increase slightly after appearing to settle near 5 meters. This occurs because one of the UAVs actually moves interior to the circle around the target, shown in Figure 14. The UAV directly below the target is approximately 15 meters away from its desired location, and causes the maximum error seen in Figure 13.

6 CONCLUSIONS

Discrete-time implementation of networked multi-agent systems introduces significant challenges caused by effects such as network delays and data loss. This paper proposes a passive-by-construction distributed network control design framework that ensures l_2^m -stability in the presence of these network effects. Using steady-state analysis, we show how to control the agents in the multi-agent network in order to establish an n -gon upon deployment. Simulations supporting the theoretical results are presented on the

Simulink/TrueTime platform. In future work, we plan to extend the design framework to achieve other group oriented tasks such as output synchronization, formation control, and rendezvous. We will also extend the work to formations in \mathbb{R}^3 .

ACKNOWLEDGEMENTS

This work is supported in part by the National Science Foundation (NSF CCF-0820088), the U.S. Army Research Office (ARO W911NF-10-1-0005), the U.S. Air Force Office of Scientific Research (MURI FA9550-06-0312), the U.S. Army Research Laboratory (ARL W911NF-087-2-0004), and Lockheed-Martin. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government.

REFERENCES

- Arcak, M. (2007). Passivity as a design tool for group coordination. *IEEE Transactions on Automatic Control*, 52(8):1380–1390.
- Bai, H., Arcak, M., and Wen, J. T. (2008). Rigid body attitude coordination without inertial frame information. *Automatica*, 44(12):3170 – 3175.
- Chopra, N., Berestesky, P., and Spong, M. (2008). Bilateral teleoperation over unreliable communication networks. *IEEE Transactions on Control Systems Technology*, 16(2):304–313.
- Fax, J. A. and Murray, R. M. (2004). Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465 – 1476.
- Godsil, C. and Royle, G. (2001). *Algebraic Graph Theory*. Springer-Verlag New York, Inc.
- Igarashi, Y., Hatanaka, T., Fujita, M., and Spong, M. (2008). Passivity-based output synchronization in se(3). In *American Control Conference*, pages 723–728.
- Ihle, I.-A. F., Arcak, M., and Fossen, T. I. (2007). Passivity-based designs for synchronized path-following. *Automatica*, 43(9):1508 – 1518.
- Kottenstette, N. and Antsaklis, P. (2007). Stable digital control networks for continuous passive plants subject to delays and data dropouts. *46th IEEE Conference on Decision and Control*, pages 4433–4440.
- Kottenstette, N., Hall, J., Koutsoukos, X., Antsaklis, P., and Sztipanovits, J. (2009). Digital control of multiple discrete passive plants over networks. *International Journal of Systems, Control and Communications (IJSCC): Special Issue on Progress in Networked Control Systems*. To Appear.
- Kottenstette, N. and Porter, J. (2009). Digital Passive Attitude and Altitude Control Schemes for Quadrotor Aircraft. *7th International Conference on Control and Automation*.
- Lawrence, D. A., Frew, E. W., and Pisano, W. J. (2008). Lyapunov vector fields for autonomous uav flight control. *AIAA Journal of Guidance, Control, and Dynamics*, 31(5):1220–1229.
- MathWorks, I. T. (2008). Simulink. *Dynamic System Simulation for MATLAB, Version 7.1*.
- Niemeyer, G. and Slotine, J.-J. E. (2004). Telemanipulation with time delays. *International Journal of Robotics Research*, 23(9):873 – 890.
- Ohlin, M., Henriksson, D., and Cervin, A. (2007). *TrueTime 1.5 Reference Manual*. Dept. of Automatic Control, Lund University, Sweden. <http://www.control.lth.se/truetime/>.
- Olfati-Saber, R. (2006). Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420.
- Olfati-Saber, R., Fax, J. A., and Murray, R. M. (2007). Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233.
- Ren, W., Beard, R., and Atkins, E. (2005). A survey of consensus problems in multi-agent coordination. In *Proceedings of the American Control Conference*, pages 1859–1864 vol. 3.
- Stramigioli, S., Secchi, C., van der Schaft, A. J., and Fantuzzi, C. (2005). Sampled data systems passivity and discrete port-hamiltonian systems. *IEEE Transactions on Robotics*, 21(4):574 – 587.
- van der Schaft, A. (1999). *L2-Gain and Passivity in Nonlinear Control*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- Wang, W. and Slotine, J.-J. (2006). Contraction analysis of time-delayed communications and group cooperation. *IEEE Transactions on Automatic Control*, 51(4):712–717.
- Zames, G. (1966). On the input-output stability of time-varying nonlinear feedback systems part one: Conditions derived using concepts of loop gain, conicity, and positivity. *IEEE Transactions on Automatic Control*, 11(2):228–238.

SHORT PAPERS

CHARACTERISTICS OF DEFINING HYPERPLANES OF CONSTANT RETURNS TO SCALE TECHNOLOGY IN DEA

Sevan Sohraiee

*Department of Mathematics, Tehran North Branch, IAU, Darband st. Tehran, Iran
sohraiee@yahoo.com*

Keywords: Data envelopment analysis, Constant returns to scale technology, Defining hyperplanes.

Abstract: In this paper characteristics of defining hyperplanes of constant returns to scale technology in DEA have been investigated. A defining hyperplane namely H is a type of hyperplane that with the elimination of H , the production possibility set (PPS) will be enlarged (In this paper a defining hyperplane exactly is the full dimensional efficient facet (FDEF) and may be found in Olesen and Peterson (1996, 2003)). The point of view of some of the characteristics is conceptual and the interpretation of defining hyperplanes of constant returns to scale technology can be achieved by these conceptual characteristics. However, some of the characteristics are practical and one can easily utilize them in practice. Some parts of topology and convex analysis have been considered to show the truth of characteristics.

1 INTRODUCTION

Data envelopment analysis (DEA) is a non-parametric approach which was suggested by Charnes et al. (1978) to measure the relative efficiency of a decision making unit (DMU) and provide DMUs with relative performance assessment on multiple inputs and outputs. Based on different essential properties and corresponding to different characteristics of the production possibility set (PPS) and production frontiers, different DEA models, such as the CCR model, the BCC model and the FDH model, have been introduced.

An important task of DEA is to identify the returns to scale (RTS) of DMUs based on the position of the supporting hyperplanes of efficient frontier. Therefore, the investigation of different types of hyperplanes of efficient frontier or PPS is an important part of DEA.

No many papers in DEA have been written on the subject of "investigation of efficient frontier" and "characteristics of different types of hyperplanes". Finding of the piecewise linear frontier of production function which identifies the efficient frontier and efficient DMUs in DEA has been investigated by Jahanshahloo et al. (2005), in particular the aim of their study was to develop a way to obtain efficient frontier by using 0-1 integer programming, then by means of it, identification of

efficient DMUs and their returns to scale characteristics. Also, searching of efficient frontier in DEA, has been considered by Korhonen (1997). Korhonen tried to provide the decision maker (DM) an interactive method which allows him or her to incorporate performance information in to the efficient frontier analysis by enabling him or her to make a free search on efficient frontier, furthermore, Korhonen provided the DM all references of an inefficient DMU, enabling him or her to choose the most preferable unit as reference. Furthermore, Jahanshahloo et al. (2007) suggested a way of finding strong defining hyperplanes of production possibility set in DEA, particularly their method is based on the relation between efficient surfaces and strong defining hyperplanes of production possibility set. Also, Cooper et al. (2007) make it possible to select the weights, obtained by the multiplier model in DEA, associated with the facets of higher dimension that a DMU generates, in particular their method supplies model for locating facets of the maximum possible dimension of the efficient frontier. Furthermore, the construction of all DEA efficient frontiers in generalized data envelopment analysis (GDEA) has been discussed by Yu. et al. (1996).

Almost in all of the abovementioned researches, there is no investigation about the characteristics of defining hyperplanes of production possibility set that is so essential in DEA. In this paper, we have

presented some essential theorems in order to identify the defining hyperplanes of constant returns to scale (CRS) technology. These theorems enable us to recognize whether a hyperplane obtained by the optimal solution of the multiplier form of CCR model is a defining hyperplane.

Furthermore, one of the most important task of defining hyperplanes of production possibility set is sensitivity analysis that enable us to determine the amounts of perturbations of data that can be tolerated by a DMU on efficient frontier before becoming inefficient. Also, we can utilize the concept defined in this paper in order to evaluate the efficiency of DMUs by using the defining hyperplanes of PPS, which efficient DMUs are on them.

Some of the characteristics presented in this paper are more conceptual, however others are more practical. Furthermore, the conceptual point of view of theorems presented in this paper, enable us to interpret the characteristics of defining hyperplanes of CRS technology. Although some of the theorems are so practical and one can easily utilize them in practice. Not only, the conceptual point of view of theorems is essential and is so useful to interpretation of defining hyperplanes of CRS technology, but also the practical point of view of theorems is a necessity and enable us to utilize the characteristics in practice.

The aim of this paper is to use the conceptual point of view of some parts of topology and convex analysis and a combination of them with DEA to present some conceptual and practical characteristics in order to determine when a hyperplane of PPS is a defining hyperplane. The main idea of this paper is based on the geometrical interpretation of efficient facets of the highest dimension of the frontier that the DMU under assessment contributes to span. In particular a defining hyperplane is a full dimensional efficient facet (FDEF) and may be found in Olesen and Peterson (2003). These geometrical interpretations enable us to establish the presented characteristics. Some of these characteristics are conceptual that we will not be able to utilize them in practice. Although, we use these conceptual characteristics in order to establish some practical characteristics that one may easily utilize them in practice.

The sections of this paper are organized as follows. In the next section, Section 2, we provide additional background of our paper. In Section 3, we give basic concepts of some parts of topology, convex analysis and DEA models. Section 4 investigates the characteristics of defining

hyperplanes of constant returns to scale (CRS) technology. In Section 5, we present an example to illustrate the characteristics.

2 BACKGROUND

As previously noted, this paper is dealt with the characteristics of defining hyperplanes of CRS technology in DEA. These defining hyperplanes play an important role in DEA as previously mentioned.

In this paper, we restrict attention to geometrical differences between defining hyperplanes of CRS technology and those supporting hyperplanes of CRS technology that are not defining. As we know, these two kinds of hyperplanes play a crucial role in DEA, since they are generally utilized to determine different types of concepts such as efficiency, bench mark DMUs, rates of substitution and transformation, returns to scale, sensitivity analysis and etc.

The main idea of this paper is based on geometrical interpretation of defining hyperplanes of CRS technology. In order to state a geometrical characteristics of defining hyperplanes of CRS technology, we use a combination of different kinds of concepts such as interior points of a set, an ε -neighborhood around a point and geometrical interpretation of CRS technology efficient frontier to state a specific relation between the dimension of intersection of each defining hyperplanes with the production possibility set (PPS) of CRS technology that we use this characteristics to show the truth of others stated characteristics.

Secondly, we utilize a model proposed by Cooper et al. (2007) to determine a hyperplane that is binding at the maximum number of extreme efficient units. With utilizing the abovementioned hyperplane namely H^* , we define a created DMU obtained by center of gravity of extreme efficient units that the abovementioned hyperplane H^* is binding at them. Eventually, a set of feasible directions obtained by connecting the created DMU to each extreme efficient unit that the hyperplane H^* is binding at them has been defined to present a practical characteristic.

3 THEORETICAL CONSIDERATIONS

3.1 Some Basic Concepts of Topology

In this subsection we review some topological properties of sets and some basic results from convex analysis.

Definition 1. Given a point $x \in R^n$, a ε -neighborhood around it is the set

$$N_\varepsilon(x) = \{y \mid \|y - x\| < \varepsilon\} \quad (1)$$

Definition 2. Let X be an arbitrary set in R^n . x is said to be in the *interior* of X , denoted by $\text{int } X$, if $N_\varepsilon(x) \subset X$ for some $\varepsilon > 0$.

Definition 3. Let X be an arbitrary set in R^n . x is said to be in the *boundary* of X , denoted by ∂X , if $N_\varepsilon(x)$ contains at least one point in X and one point not in X for every $\varepsilon > 0$.

Definition 4. A set X in R^n is called a *convex set* if given any two points x_1 and x_2 in X then $\lambda x_1 + (1 - \lambda)x_2 \in X$ for each $\lambda \in [0, 1]$.

Definition 5. A point x in a convex set X is called an *extreme point* of X , if x can not be represented as a strict convex combination of two distinct points in X .

Definition 6. A *hyperplane* H in R^n is a set of the form

$$\{x \mid px = k\} \quad (2)$$

where p is a non-zero vector in R^n and k is a scalar. Also, p is usually called the *normal* or the *gradient* to the hyperplane.

Definition 7. A hyperplane divides R^n into two regions, called half spaces. Hence two *half spaces* H^+ and H^- may be defined in the following manner:

$$H^+ = \{x \mid px \geq k\} \quad (3)$$

$$H^- = \{x \mid px \leq k\} \quad (4)$$

where p is a non-zero vector in R^n and k is a scalar. Also,

$$H = H^+ \cap H^- \quad (5)$$

Definition 8. A *polyhedral set* or *polyhedron* is the intersection of a finite number of halfspaces. A bounded polyhedral set is called a *polytope*.

Suppose that the polyhedral set under discussion in the following definitions has the form

$$X = \{x \mid Ax \leq b, x \geq 0\} \quad (6)$$

where A is $m \times n$ and b is an m -vector. The hyperplanes associated with the $(m + n)$ defining halfspaces

$$\{x \mid a^i x \leq b^i\}, i = 1, \dots, m \quad (7)$$

And

$$\{x \mid e_j x \geq 0\}, j = 1, \dots, n \quad (8)$$

are called *defining hyperplanes* of X .

Definition 9. Let $\bar{x} \in X$. A constraint $a^l x \leq b^l$ is *binding*, or *tight*, or *active*, at $\bar{x} \in X$, if

$$a^l \bar{x} = b^l \quad (9)$$

Definition 10. A hyperplane H is a *supporting hyperplane* of X , if

$$H \cap X \neq \Phi \text{ \& } (X \subseteq H^+ \text{ or } X \subseteq H^-) \quad (10)$$

Definition 11. The set of points in X that correspond to some non-empty of binding defining hyperplanes of X are called *faces* of X . Given any face F of X if $r(F)$ is the maximum number of linearly independent defining hyperplanes binding at all points feasible to F , then the *dimension* of F , denoted by $\text{dim}(F)$, is equal to $n - r(F)$.

Also, the highest dimensional face of X is of dimension $\text{dim}(X) - 1$ and it is called a *facet* of X .

3.2 DEA Background

Assume that we have n DMUs each consuming m inputs and producing s outputs. Let \bar{X} be an

$(m \times n)$ -matrix and Y be a $(s \times n)$ -matrix consisting of non-negative elements, containing observed input and output measures for the DMUs, respectively. We denote by $X_j \geq 0, X_j \neq 0, j = 1, \dots, n$ (the j th column of \bar{X}) the vector of inputs consumed by DMU $_j$. A similar notation is used for outputs.

The traditional CCR models, as introduced by Charnes et al. (1978) are fractional linear programs, which can easily be formulated and as linear programs. Those models are so-called constant returns to scale (CRS) models. Later Banker et al. (1984) developed the so-called BCC models with variable returns to scale (VRS).

The CCR and BCC models are the basic model types in DEA. Those basic models can be presented in a primal or dual form. The usage of primal and dual varies in the literature, and it is more straightforward to call them multiplier and envelopment models, respectively. The multiplier model provides information on the weights of inputs and outputs. The weights are interpreted as prices in many applications. The envelopment models provide the user with information on the lacks of outputs and the surplus of inputs of a unit. Also, the envelopment model characterizes the reference set for the units. Moreover, the production possibility set (PPS) of CCR and BCC models can be interpreted from the structure of envelopment models. Since, we are interested in CCR models in this paper, we represent the PPS of constant returns to scale (CRS) technology in the following manner:

$$T_c = \left\{ Z = (X, Y)^T \mid \mathbf{X}\lambda \leq X, \mathbf{Y}\lambda \geq Y, \lambda \geq 0 \right\} \quad (11)$$

Based on the PPS of CRS technology the envelopment form of CCR model is in the following manner:

$$\begin{aligned} \text{Min} \quad & \theta \\ \text{s.t.} \quad & \mathbf{X}\lambda \leq \theta X_o \\ & \mathbf{Y}\lambda \geq Y_o \\ & \lambda \geq 0. \end{aligned} \quad (12)$$

The multiplier form of model CCR based on the dual of model (12) is as follows:

$$\text{Max} \quad U^T Y_o \quad (13)$$

$$\text{s.t.} \quad V^T X_o = 1 \quad (13.1)$$

$$\begin{aligned} U^T Y_j - V^T X_j &\leq 0, j = 1, \dots, n \\ U &\geq 0, V \geq 0 \end{aligned} \quad (13.2)$$

We know that in the optimal solution (U^*, V^*) of model (13), at least one constraints of (13.2) is binding. Also, it is easy to show, this optimal solution (U^*, V^*) is the normal vector of a supporting hyperplane

$$H^* = \left\{ (X, Y)^T \mid U^{*T} Y - V^{*T} X = 0 \right\} \quad (14)$$

which, supports T_c constructed by observed data.

Definition 12. DMU $_o$ is an *extreme efficient unit* if in the evaluation of DMU $_o$, the optimal solution of model (12) is unique and

$$\lambda_o^* = 1, \lambda_{j \neq o}^* = 0 \quad (15)$$

Also, the indices of all extreme efficient units is denoted by

$$E = \left\{ j \mid \text{DMU}_j \text{ is an extreme efficient unit} \right\} \quad (16)$$

We know that T_c is the intersection of some hyperplanes. We call some of these hyperplanes as *defining hyperplanes* if with the elimination of these hyperplanes, T_c will be enlarged.

Definition 13. A hyperplane H is a defining hyperplane of T_c if with the elimination of H, T_c will be enlarged (A defining hyperplane used in this paper exactly is FDEF defined by Olesen and Peterson (2003)).

4 CHARACTERISTICS OF DEFINING HYPERPLANES OF CRS TECHNOLOGY

In this section, we present some essential theorems in order to recognize all defining hyperplanes of T_c . In these theorems some important characteristics of defining hyperplanes of T_c have been identified. These theorems enable us to recognize when a hyperplane is a defining hyperplane of T_c . Therefore, using these theorems one will be able to recognize any defining hyperplanes of T_c which was not possible before. As mentioned in previous

sections, if (U^*, V^*) is an optimal solution of the multiplier model of CRS technology (13), then

$$\{(X, Y) \mid U^{*T}Y - V^{*T}X = 0\} \quad (17)$$

will be a supporting hyperplane of T_c . Assume that

$$H_t = \left\{ (X, Y)^T \mid U_t^T Y - V_t^T X = 0 \right\} \quad (18)$$

$t = 1, \dots, K$

are all defining hyperplanes of T_c which we are interested in. Also consider two defining half-spaces

$$H_t^+ = \left\{ (X, Y)^T \mid U_t^T Y - V_t^T X \geq 0 \right\} \quad (19)$$

and

$$H_t^- = \left\{ (X, Y)^T \mid U_t^T Y - V_t^T X \leq 0 \right\} \quad (20)$$

obtained by hyperplane H_t for each $t = 1, \dots, K$. With out loss of generality, we can assume that T_c is the intersection of all defining half-spaces H_t^- , $t = 1, \dots, K$ in the following manner:

$$T_c = \bigcap_{t=1}^K H_t^- \quad (21)$$

Theorem 1. *The hyperplane H is a defining hyperplane of T_c if and only if the dimension of $(T_c \cap H)$ equals $m + s - 1$.*

Proof. Assume that the dimension of $(T_c \cap H)$ equals $m + s - 1$. On one hand, since, $T_c \cap H \neq \Phi$, we can find a point such as $\bar{Z} = (\bar{X}, \bar{Y})^T \in T_c \cap H$ for which there exists a $(m + s - 1)$ dimensional ε -neighborhood $N_\varepsilon(\bar{Z}) \subseteq T_c \cap H$. On the other hand, $\bar{Z} = (\bar{X}, \bar{Y})^T$ is a point contained in T_c for which, only one hyperplane such as H is binding. Therefore, with the elimination of half-space H^- obtained by hyperplane H from T_c (without loss of generality assume that $T_c \subseteq H^-$), the point $\bar{Z} = (\bar{X}, \bar{Y})^T$ will be an interior point of T'_c (T'_c is

the set obtained by the elimination of half-space H^- from T_c). Note that $T_c \subseteq T'_c$. Since, $\bar{Z} = (\bar{X}, \bar{Y})^T$ is an interior point of T'_c , therefore, there exists an $\bar{\varepsilon} > 0$, for which, $N_{\bar{\varepsilon}}(\bar{Z}) \subseteq T'_c$. Also, since $\bar{Z} = (\bar{X}, \bar{Y})^T \in \partial T_c$, each $N_\varepsilon(\bar{Z})$ contains at least one point in T_c and one point not in T_c for every $\varepsilon > 0$. Now, assume that this neighborhood is $N_{\bar{\varepsilon}}(\bar{Z})$. Thus, there exists a point such as Z_o for which $Z_o \in N_{\bar{\varepsilon}}(\bar{Z})$ and $Z_o \notin T_c$. This shows that $Z_o \in T'_c$ and $Z_o \notin T_c$. Consequently, $T_c \subset T'_c$ and therefore, it means that with the elimination of half-space H^- from T_c , T_c has been enlarged. Thus, H is a defining hyperplane of T_c .

To show the converse, assume that the hyperplane H is a defining hyperplane of T_c . It is obvious that the dimension of each hyperplane such as H in R^{m+s} such as T_c is equal to $(m + s - 1)$. Since, H is a defining hyperplane of T_c therefore, there exists a point such as $\bar{Z} = (\bar{X}, \bar{Y})^T$ in the interior of $T_c \cap H$. Now, with the elimination of half-space H^- from T_c , we will encounter with a set called T'_c ($T_c \subset T'_c$). Since, $\bar{Z} = (\bar{X}, \bar{Y})^T \in \text{int}(T_c \cap H)$, thus, $\bar{Z} = (\bar{X}, \bar{Y})^T$ will be an interior point of T'_c . This implies that there exists an $\bar{\varepsilon} > 0$ for which the $(m + s)$ -dimensional $N_{\bar{\varepsilon}}(\bar{Z}) \subset T'_c$. Now, it is trivial that $N_{\bar{\varepsilon}}(\bar{Z}) \cap H$ is $(m + s - 1)$ -dimensional and this implies that $T_c \cap H$ is $(m + s - 1)$ -dimensional. This completes the proof. \square

Theorem 1 shows a characteristic of defining hyperplane of T_c . In order to simplify and find more simple methods for introduction of defining hyperplanes of T_c , we need to use the following model that has been introduced by Cooper et al. (2007) with some minor modification:

$$\begin{aligned}
 & \text{Min} \quad \sum_{j=1}^n l_j \\
 & \text{St.} \quad U^T Y_p = 1 \\
 & \quad \quad V^T X_p = 1 \\
 & \quad \quad U^T Y_j - V^T X_j + t_j = 0, j \in E \\
 & \quad \quad t_j - l_j M \leq 0, j \in E \\
 & \quad \quad l_j \in \{0, 1\}, j \in E \\
 & \quad \quad U \geq 0, V \geq 0, t_j \geq 0, j \in E.
 \end{aligned} \tag{22}$$

Where M is sufficiently large positive number and E is the set of indices of all extreme efficient DMU's defined in previous sections.

Since, $E \neq \Phi$, thus model (22) finds a hyperplane which, is binding, at the maximum number of extreme efficient units.

Assume that

$$\begin{aligned}
 (U^*, V^*, T^*, L^*), T^* &= (t_1^*, \dots, t_{|E|}^*) \\
 L^* &= (l_1^*, \dots, l_{|E|}^*)
 \end{aligned} \tag{23}$$

is an optimal solution of model (22). We define the hyperplane H^* in the following manner:

$$H^* = \left\{ (X, Y)^T \mid U^{*T} Y - V^{*T} X = 0 \right\} \tag{24}$$

The following theorem emphasizes the existence of a defining hyperplane of T_c at each extreme efficient unit.

Theorem 2. *There exists at least one defining hyperplane of T_c such as H for each $j \in E$, for which, $Z_j = (X_j, Y_j)^T \in H$.*

Proof. As we know, $T_c = \bigcap_{t=1}^K H_t^-$. To the contrary of the desired result, suppose that there is no defining hyperplane of T_c which is binding at $Z_j = (X_j, Y_j)^T, j \in E$. Therefore,

$$Z_j = (X_j, Y_j) \in \text{int}(H_t^-), t = 1, \dots, K.$$

Consequently, $Z_j = (X_j, Y_j) \in \text{int}(T_c)$, which is in contradiction with $j \in E$. This completes the proof. \square

In order to improve the conditions under which one can more easily identify the defining hyperplanes of T_c and present more practical characteristics of determining defining hyperplanes

of T_c , we define a set based on the optimal solution of model (22) as follows:

$$E^* = \left\{ j \mid t_j^* = 0, \text{in (22)} \right\} \tag{25}$$

The following theorem, shows that, E^* is not vacuous.

Theorem 3. $E^* \neq \Phi$.

Proof. The proof is obvious and omitted. \square

The improvement of conditions and characteristics of determining defining hyperplanes of T_c made us define a created DMU in the following manner:

$$Z^* = (X^*, Y^*)^T = \frac{1}{|E^*|} \sum_{j \in E^*} (X_j, Y_j)^T \tag{26}$$

Particularly, $Z^* = (X^*, Y^*)^T$ is the center of gravity of extreme efficient units for which, the hyperplane H^* (defined based on the optimal solution of model (22)) is binding. The following theorem states that, $Z^* = (X^*, Y^*)^T$ is in boundary of T_c .

Theorem 4. $Z^* = (X^*, Y^*)^T \in \partial T_c$.

Proof. Noting theorem 3, we have $E^* \neq \Phi$.

Therefore, H^* defined in (24) is a supporting hyperplane of T_c . Since, T_c is a convex set therefore, $Z^* = (X^*, Y^*)^T \in T_c$. Also, we have

$$\begin{aligned}
 U^* Y^* - V^* X^* &= \\
 U^* \left(\frac{1}{|E^*|} \sum_{j \in E^*} Y_j \right) - V^* \left(\frac{1}{|E^*|} \sum_{j \in E^*} X_j \right) &= \\
 \frac{1}{|E^*|} \sum_{j \in E^*} (U^* Y_j - V^* X_j) &= 0
 \end{aligned} \tag{27}$$

This shows that the defining hyperplane H^* is binding at $Z^* = (X^*, Y^*)^T \in T_c$ and it means that $Z^* = (X^*, Y^*)^T \in \partial T_c$ and this completes the proof. \square

Theorem 5. *If the optimal solution of model (13) in the evaluation of created unit $Z^* = (X^*, Y^*)^T$ is unique then the hyperplane H^* will be a defining hyperplane of T_c .*

Proof. Consider the following model, which is the multiplier form of CCR model based on the set E when unit $Z^* = (X^*, Y^*)^T$ is under evaluation:

$$\begin{aligned}
 & \text{Max} && U^T Y^* \\
 & \text{S.t.} && U^T Y_j - V^T X_j \leq 0, j \in E \\
 & && U^T Y^* - V^T X^* \leq 0 \\
 & && V^T X^* = 1 \\
 & && U \geq 0, V \geq 0.
 \end{aligned} \tag{28}$$

Assume that model (28) has unique optimal solution (\bar{U}, \bar{V}) . Define,

$$\bar{H} = \left\{ (X, Y)^T \mid \bar{U}^T Y - \bar{V}^T X = 0 \right\} \tag{29}$$

It is obvious that $\bar{U}^T Y^* - \bar{V}^T X^* = 0$ and \bar{H} is the only supporting hyperplane of T_c at $Z^* = (X^*, Y^*)^T$. Therefore, we can define a face of T_c for which, $Z^* = (X^*, Y^*)^T$ is on it as follows:

$$\bar{F} = \bar{H} \cap T_c$$

It is trivial that \bar{F} is the only face contained $Z^* = (X^*, Y^*)^T$, therefore, the dimension of \bar{F} equals to $m + s - 1$ and this means that the dimension of $\bar{H} \cap T_c$ equals to $m + s - 1$. Therefore, considering Theorem 1, \bar{H} is a defining hyperplane of T_c . Note that \bar{H} is equivalent to H^* and this means that H^* is a defining hyperplane of T_c , thus the proof is complete. \square

To simplify and improve better recognition of defining hyperplanes of T_c , we utilize the following definition of feasible directions constructed by connecting $Z^* = (X^*, Y^*)^T$ to each extreme efficient unit that H^* is binding at them:

$$D = \left\{ d_j \mid d_j = (X_j, Y_j)^T - (X^*, Y^*)^T, j \in E^* \right\} \tag{30}$$

Theorem 6. *If the dimension of D equals to $m + s - 1$ then H^* is a defining hyperplane of T_c .*

Proof. Noting that $Z^* = (X^*, Y^*)^T \in T_c$, $Z_j = (X_j, Y_j)^T \in T_c$ for each $j \in E^*$ and T_c is a

convex set, we have $Z^* + \lambda(Z_j - Z^*) \in T_c$ for each $j \in E^*$ and $\lambda \in [0, 1]$. Also, since $Z^* = (X^*, Y^*)^T \in H^*$, $Z_j = (X_j, Y_j)^T \in H^*$ for each $j \in E^*$ and H^* is a convex set, therefore $Z^* + \lambda(Z_j - Z^*) \in H^*$ for each $j \in E^*$ and $\lambda \in [0, 1]$. Thus, these imply that $Z^* + \lambda(Z_j - Z^*) \in T_c \cap H^*$ for each $j \in E^*$ and $\lambda \in [0, 1]$. Therefore, since, the dimension of D equals to $m + s - 1$, thus we have $m + s - 1$ independent feasible direction at Z^* in $T_c \cap H^*$. This implies that the dimension of $T_c \cap H^*$ equals to $m + s - 1$ and by theorem 1, H^* is a defining hyperplane of T_c . Therefore the proof is complete. \square

5 ILLUSTRATIVE EXAMPLE

In order to illustrate the characteristics of Theorems, we present a numerical example with the data set as in table 1. The CRS technology based on the data set in Table 1, has been illustrated in Fig. 1. This figure can be viewed as representing a section at a given output level, say $y = 1$, of the PPS generated two DMUs (A and B) that use two inputs and produce the same quantity of output ($y = 1$). The optimal solutions of (12) when assessing the efficiency of the extreme efficient DMU A or DMU B correspond to the coefficients of the supporting hyperplanes at A or B, which pass through origin. Model (22) then selects the hyperplane represented with a dark solid line connecting as distinct from the ones represented by the lighter dotted lines. The first one is obviously preferable to the latter because it is supported by two units (A and B) instead of by only one (A) or one (B). Moreover, in this particular case, this also means that it contains a FDEF of the frontier that DMU A and DMU B contribute to generate.

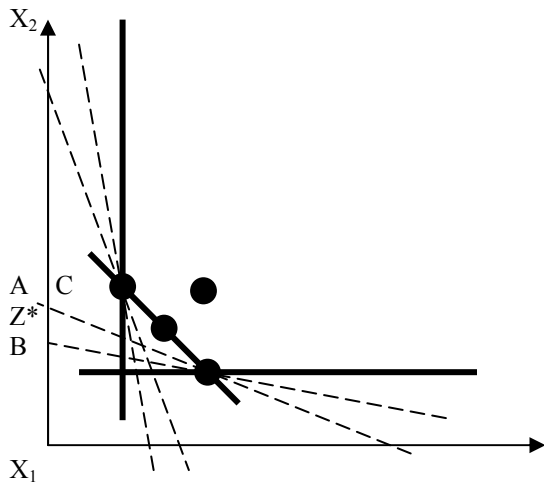


Figure 1: The graph of example.

Table 1: Data set.

DMU	Input 1	Input 2	Output
A	1	2	1
B	2	1	1
C	2	2	1

It is obvious that the defining hyperplanes of T_c are in the following manner:

$$\begin{aligned} H_1 &= \{(x_1, x_2, y)^T \mid 2y - 2x_1 = 0\} \\ H_2 &= \{(x_1, x_2, y)^T \mid 3y - x_1 - x_2 = 0\} \\ H_3 &= \{(x_1, x_2, y)^T \mid 2y - 2x_2 = 0\} \end{aligned}$$

We can see that $\dim(T_c \cap H_1) = 2$, $\dim(T_c \cap H_2) = 2$ and $\dim(T_c \cap H_3) = 2$ as it has been shown in Theorem 1. Therefore, the condition of Theorem 1, $\dim(T_c \cap H_i) = m + s - 1$ has been satisfied and this shows the truth of Theorem 1.

The optimal solution of model (3) shows that $E = \{1, 2\}$. As stated in Theorem 2, the hyperplanes H_1 and H_2 are two defining hyperplanes of T_c associated with *DMU A* and the hyperplanes H_2 and H_3 are two defining hyperplanes of T_c associated with *DMU B*. These show the truth of Theorem 2.

If we solve the model (22) it will be obtained that $E^* = \{1, 2\}$ and this shows the truth of Theorem 3.

If we utilize the relation (26) we will encounter with a created DMU,

$$\begin{aligned} Z^* &= (X^*, Y^*)^T \\ &= \frac{1}{2}(1, 2, 1)^T + \frac{1}{2}(2, 1, 1)^T = (1.5, 1.5, 1)^T \end{aligned} \quad (31)$$

which has been shown in Fig. 1. It is trivial that the hyperplane H_2 is binding at $Z^* = (1.5, 1.5, 1)^T$. Consequently, $Z^* = (1.5, 1.5, 1)^T \in \partial T_c$ and this shows the truth of Theorem 4.

If we solve model (11) associated with created DMU, $Z^* = (1.5, 1.5, 1)^T$, we will obtain a unique optimal solution $(u^*, v_1^*, v_2^*) = \left(\frac{1}{3}, \frac{1}{3}, 1\right)$. Now, based on the optimal solution of model (28), the hyperplane H^* will be in the following manner:

$$H^* = \left\{ (x_1, x_2, y)^T \mid y - \frac{1}{3}x_1 - \frac{1}{3}x_2 = 0 \right\} \quad (32)$$

that is exactly the hyperplane H_2 . This shows the truth of Theorem 5.

The set D as stated in (13) is as follows:

$$D = \left\{ \begin{aligned} d_1 &= (-0.5, 0.5, 0)^T, \\ d_2 &= (0.5, -0.5, 0)^T \end{aligned} \right\} \quad (33)$$

It is obvious that $\dim(D) = 1$. Since, H^* is a defining hyperplane of T_c , the converse of Theorem 6 does not hold and this shows that Theorem 6 is only a sufficient condition.

6 CONCLUSIONS

In this paper, some parts of topology and convex analysis have been utilized in order to state some characteristics of defining hyperplanes of CRS technology in DEA. These characteristics enable us to recognize whether a hyperplane obtained by the optimal solution of multiplier form of CCR model is a defining hyperplane. Some of the characteristics are conceptual and some of them can be easily utilized in practice. An illustrative example has been considered, in order to show the truth of characteristics stated in this paper.

We suggest as a future research, introduction of an algorithm to recognize all defining hyperplanes of

CRS technology based on characteristics presented in this paper. Also, we look for similar characteristics in the case of variable returns to scale technology as a future research.

REFERENCES

- Banker, R. D., Charnes, A., Cooper, W. W., 1984. Some models for estimating technical and scale inefficiencies in data envelopment analysis. 30 1078-1092, *Management Science*.
- Charnes, A., Cooper, W. W., Rhodes, E., 1978. Measuring the efficiency of decision making units, 2 429-444. *European Journal of Operational Research*.
- Cooper, W. W., Ruiz, J. L., Inmaculada Sirvent, 2007. Choosing weights from alternative optimal solutions of dual multiplier models in DEA, 180 443-458. *European Journal of Operational Research*.
- Jahanshahloo, G. R., Hosseinzadeh Lotfi, F., Zhiani Rezai, H., Rezai Balf, F., 2007. Finding strong defining hyperplanes of Production Possibility Set, 177 42-54. *European Journal of Operational Research*.
- Jahanshahloo, G. R., Hosseinzadeh Lotfi, F., Zohrehbandian, M., 2005. Finding the piecewise linear frontier production function in Data Envelopment Analysis, 163 483-488. *Applied Mathematics and Computation*.
- Korhonen, P., 1997. Searching the efficient frontier in Data Envelopment Analysis, IR-79-97. *IIASA*.
- Olesen, O., Petersen, N. C., 1996. Indicators of ill-conditioned data sets and model misspecification in data envelopment analysis: An extended facet approach, 42 205-219. *Management Science*.
- Olesen, O., Petersen, N. C., 1996. Identification and use of efficient faces facets in DEA, 20 323-360. *Journal of Productivity Analysis*.
- Yu, G., Wei, Q., Brockett, P., Zhou, L., 1996. Construction of all DEA efficient surfaces of Production Possibility Set under the generalized Data Envelopment Analysis model, 95 491-510. *European Journal of Operational Research*.

VERIFICATION OF AN INDUSTRIAL COMPUTER NETWORK OF HIGH RISK OPERATION PLANTS

A NPP APCS Example

Vitaly Promyslov and Stanislav Masolkin

*V.A. Trapeznikov Institute of Control Sciences, 65 Profsoyuznaya, Moscow 117997, Russia
vp@ipu.ru*

Keywords: Modeling, Network calculus, Computer networks, Nuclear power plants.

Abstract: A modern APCS (automated process control system) of large plants, involving nuclear power plants (NPP), is implemented as networked control system. In the paper, model based on the “network calculus” for an NPP APCS segment is presented. A method of calculation of time characteristics of the system under worst combination of input conditions is verified.

1 INTRODUCTION

An advanced APCS (automated process control system) of large plants, involving nuclear power plants (NPP), is implemented as a distributed in functions and tools system with components interacting with each other and with the plant by use of a local area network (LAN). The time of passage of a signal from a source to a receiver is an important time characteristics of an APCS. The signal source may be both an operator initiating an action (control signal) – then the receiver is a controller (gateway) transferring the command directly to the lower level to an actuator, and a digitized sensor signal– then the receiver is a workstation at which the sensor signal is displayed.

Measuring this parameter within the process of performance and adjustment of an NPP APCS is, as a rule, a diagnostic function of the APCS. However, direct measurements do not provide a required quality of implementing the function, what is determined by influence of the following factors:

- Some modes of LAN performance may not be achievable under normal operation (being emergence ones),
- Stochastic nature of interaction of components of APCS software with each other, presence of network equipment leads to the fact that the measured parameter is a random value having a complex distribution (Chen et. al. 2009).

For high risk operation plants, in addition to direct measurements of the signal passage time with

subsequent statistical processing, one should use a method which enables one to estimate the parameter theoretically under worst combination of all possible conditions influencing the measured parameter. We investigate an applicability of a method known as “network calculus” (Le Boudec and Thiran 2001) to calculating LAN parameters. The calculation and verification have been implemented for software of the top level (SCADA) of prospective Russian NPP APCSs (Byvaikov et. al. 2006) developed at the V.A. Trapeznikov Institute of Control Sciences of the Russian Academy of Sciences. All main data presented in the paper have been received by the authors in course of implementation of the “Kudankulam” (India) NPP APCS.

2 PROBLEM STATEMENT

The NPP APCS LAN is partitioned on several segments in accordance to technological compartments (reactor compartment, turbine compartment, etc.). The main data array circulates inside a separate APCS LAN, amount of data transferred between the segments of the APCS LAN and NPP is small.

In Figure 1, a typical make-up of one segment of NPP APCS is presented. Each segment is a set of servers, gateways, and workstations united by the network through a switchboard. For communication between system components via the LAN, the TCP/IP protocol of the class A is used, as a channel

level, the Ethernet network has been selected. Such a solution, as practice shows, provides under steady state acceptable stability and small time of propagation a signal over the network between components for given conditions (Profinet V1, ModBus/IDA, Ethernet/IP) (Witschl et. al. 2006).

To analyze information flows circulating within one segment of the NPP APCS, the segment LAN has been presented as a block-scheme (Figure 1). In the scheme nodes, there are indicated gateways (G1-G5), server processing data (DB), and switchboard. Also, in the segment scheme, there were introduced nodes reflecting logical structure of the used SCADA, functioning within a workstation, indicated as IZ and AB. Internally IZ and AB are separate processes which serve a data within a single workstation and share same computing resources.

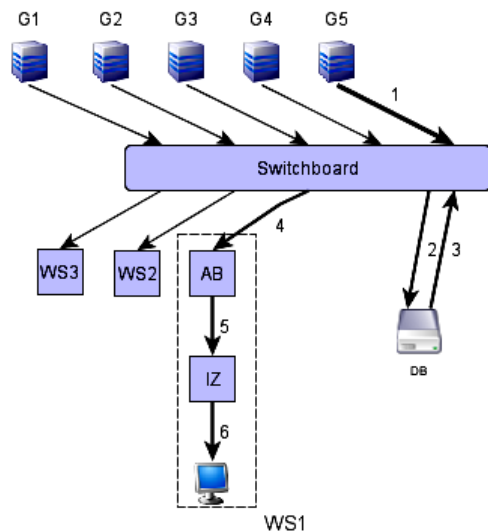


Figure 1: Model of APCS segment as a block-scheme. G1-G5 – gateways, WS1-3 – workstations.

A characteristic of the system, being of undisputable interest, is the time of passage of a signal from a source to a receiver. It is assumed that in the computer system there are provided absence of loss of information and operability.

To calculate the system parameter, the method of analysis of deterministic systems “network calculus” has been selected. The method is successively applied to calculating parameters of both large (Internet), and small (Intranet) computer networks based on the TCP/IP protocol (Vantanski et. al. 2007 and Hwangnam & Hou 2004). In the paper, the time of passage of a signal from the gateway to image in the display is investigated. Sources of information are gateways connected to the segment, servers (in a general sense of the word as devices processing

information) are the switchboard and components DB, AB, IZ. The terminal device is a workstation display.

3 NETWORK CALCULUS: BASIC CONCEPTS

The “network calculus” is a relatively new method applied to analyze deterministic systems with a queue, using the notion of plus-mini algebra. Basic principles of the method have been installed in papers (Cruz 1991a and Cruz 1991b), which, in turn, have been based on paper (Turner 1986), a description of the method may be found in work (Le Boudec & Thiran 2001).

Let us present basic notions of the applied method by use of a networked system (Figure 2) consisting of two components.

Let us define a flow function as a non-negative non-decreasing function in time: $A(t) = 0, t < 0; A: R \rightarrow R_+ \cup \{+\infty\}$. The flow function may be considered as a counter counting data inputted into the component and outputted from it. Then one says on input/output flow function correspondingly.

Before defining the next important notion, the service function, let us define an operation of “convolution” and an operation of “deconvolution”. Let there are given two functions of a flow, A and S , the convolution of A и S , is a function $A_1 = A * S: R \rightarrow R_+ \cup \{+\infty\}$, such that :

$$A(t) * S(t) := \inf_{\tau \in R} \{A(\tau) + S(t - \tau)\}.$$

It is easily to see that the function A_1 is also a function of the flow and it is non decreasing and right continuous. A binary operation of the “deconvolution”:

$$(A \oslash S)(t) := \sup_{\tau \in R} \{A(t + \tau) - S(\tau)\}.$$

Let us define functions $\gamma_{r,b}(t)$ (an affine function) and $\beta_{R,T}(t)$ (a rate-latency function) of the following form:

$$\gamma_{r,b}(t) := \begin{cases} rt + b, & t > 0 \\ 0, & t \leq 0 \end{cases},$$

$$\beta_{R,T}(t) := \begin{cases} R(t - T), & t > T \\ 0, & t \leq T \end{cases}$$

where, under modeling, r, b, T, R are frequently interpreted as flow rate, flow burstiness, flow delay, and flow capacity correspondingly.

Let us assume that an input of a network element is a flow described by a flow function A , output flow is described by a flow function A_1 . The network element has a minimal and maximal service function, S and \bar{S} correspondingly, of these meet the conditions:

$$A_1 \geq A * S, A_1 \leq A * \bar{S}.$$

Let us define a notion of “envelope” for a flow. A function E is an “envelope” of the flow A , if $A \leq A * E$.

The integral minimal (maximal) service function $S(\bar{S})$ for subsequently N connected components of a networked system without losses with minimal (maximal) service function $S_i(\bar{S}_i)$ of an i -th, $i = 1 \dots N$, component is equal to:

$$S = S_1 * S_2 * \dots * S_N$$

$$\bar{S} = \bar{S}_1 * \bar{S}_2 * \dots * \bar{S}_N$$

For the a system (Figure 2) $A^* \geq A * S$.

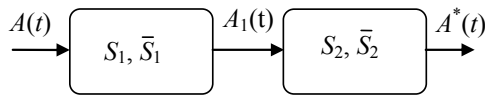


Figure 2: A networked system with two subsequently connected network components.

Let us define a delay of passage of data through a network component as:

$$d(t) := \inf_{\tau > 0} \{ \tau : A(t) \leq A_1(t + \tau) \},$$

then the maximal delay is: $\bar{D} := \inf_{d \geq 0} \{ E * \delta_d \leq S \}$,

where δ_d is burst delay function:

$$\delta_d := \begin{cases} 0, & t < 0 \\ \infty, & t \geq 0 \end{cases}$$

The minimal delay is:

$$D := \sup_{t \geq 0} \{ t : \bar{S}(t) = 0 \}.$$

4 CALCULATING DELAY UNDER PASSGE OF A SIGNAL FROM GATEWAY TO WORKSTATION

In the block-scheme (Figure 1), there is emphasized a way of signal passage from signal source

(gateway) to appear in the workstation display (edges 1-5).

In order to preserve “transparency” of the calculation (where it does not contradict to authenticity of obtained results), to set characteristics of components of the block-scheme and input flows “simple” functions were used, enabling one to receive analytical expressions of desired parameters. As a function of “envelope” of the input flow from gateways G1-G5, the function $\gamma_{r,b}(t)$ is used, as a service function for all servers (DB, AB, IZ, switchboard), the $\beta_{R,T}(t)$ is used.

Application for the “envelope” function of the input flow with the dependence $\gamma_{r,b}(t)$ is justified by an algorithm of the interaction between the gateway and DB server. In accordance to the algorithm, the gateway sends packets of a fixed size, equal to k unit of data to DB server, with a period T_γ . As a result of heterogeneity of time characteristics of the server and time of passage of the request over the network, the sampling period is kept with accuracy $\tau, \tau \ll T_\gamma$. It is known (Le Boudec and Thiran 2001), that the flow “envelope” received from the gateway is described, in that case, by the function $\gamma_{r,b}(t)$ with parameters $r = k/T_\gamma, b = k(\tau + T_\gamma)/T_\gamma$.

The algorithm of performance of the servers is related to a type of algorithms with guaranteed rate server, i.e. $h_n \leq f_n + T$, where:

$$\begin{cases} f_0 = 0 \\ f_n = \max(a_n, f_{n-1}) + l/R \end{cases}$$

where l, h_n, a_n is the size of the data package, time of completion of processing, and time of appearance of information for the input n correspondingly. For the server of this type, it is known applicability of the function $\beta_{R,T}(t)$ for setting its service function (Le Boudec & Thiran 2001). The servers have two function modes differing by parameters R and T . The modes are selected automatically in dependence on make-up of received information and operator’s actions. The minimal and maximal service functions correspond to these two modes.

To determine the parameters R_i, T_i service function of the components DB, AB, IZ and input flow r, b , their direct measurements at the plant are used, influence of the switchboard may be neglected.

Let us consider two separate types of the system behavior. The first type is preserving the uniform flow from the source to the receiver, i.e. each server at the path of passage of the signal process data in the flow only and do not change the total information amount. The second type a generalization of the first, is a variable bit rate (VBR), when a server changes amount of transferred information. The first type is rare in practice of computer systems, at least within complex system, however due to its computational simplicity it may be used under preliminary analysis of a total system and to calculate parameters of a system with uniform flow. The second type is computationally more hard, however it reflects a practical status more precisely.

4.1 Model with Uniform Flow

One can easily be seen (see Section 3) that the end-to-end minimal and maximal service functions for a flow of interest propagated along the emphasized path (Figure 1) are:

$$S = \beta_{R_s, T_s}(t) \quad (1)$$

and

$$\bar{S} = \overline{\beta_{R_M, T_M}}(t) \quad (2)$$

where (Zdarsky & Martinovic 2008):

$$R_s = (R_{DB} - \sum_{i=G1, G2, G3, G4} r_i) \wedge R_{AB} \wedge (R_{IZ} - \sum_{i=G1, G2, G3, G4} r_i); \quad (3)$$

$$T_s = T_{DB} + T_{AB} + T_{IZ} + \frac{\sum_{i=G1, G2, G3, G4} b_i + \sum_{i=G1, G2, G3, G4} r_i (T_{DB} + T_{AB} + T_{IZ})}{R_s}, \quad (4)$$

$$R_M = \min R; T_M = \sum T_i. \quad (5)$$

Equations (3), (4) are written for assumption of blind multiplexing flows in the channel, indexes G1-G5, AB, DB, IZ define parameters for a corresponding component of the system (Figure 1). In the given case, the flows intersecting with the main flow from the gateway G5 are flows from the gateways G1-G4.

Finding the minimum and summation in equation (5) is implemented over corresponding parameters of the maxima service function \bar{S}_i for each component (node) at the path of passage of the data flow correspondingly.

The minimal delay of signal passage in the system is $D = T_s$, the maximal delay is

$$\bar{D} = T_s + b_{G5} / R_s. \quad (6)$$

4.2 Model with the VBR

This model type assumes that after passing a server, at the path from source to receiver the flow may change its amount. We will consider a case when the output flow A_1' depends linearly on the output flow with complete data set A_1 :

$$A_1' = \alpha \cdot A_1 : \alpha \in R^+$$

with the “envelope”:

$$E_1' = \alpha \cdot E_1 : \alpha \in R^+,$$

where the “envelope” of the output flow may be represented via “envelope” of the input flow E and maximal and minimal service functions of the component:

$$E_1 := (E * \bar{S}) \oslash S. \quad (7)$$

In the modeled APCS segment, a significant change of the flow amount takes place after passing the server DB. In accordance to that, the block-scheme is partitioned on two parts: from the gateway to the server DB and from the DB to the operator’s display; the maximal delay of the signal passage has been being calculated separately in each part, the total delay is equal to sum of delays in each part.

For practical calculations of equation (7) when $E = \gamma_{r,b}(t)$, and the first and the second part may be described in the form of service functions $\beta_{R_1, T_1}(t)$ and $\beta_{R_2, T_2}(t)$ correspondingly, then E_1 may be substituted by the approximation:

$$E_1 \approx rt + (b + rT_1).$$

It is known that the maximal delay calculated separately over path parts after summation becomes more than the maximal delay calculated over the total path (the “pay burst only once” principle) (Le Boudec and Thiran 2001). For the system partitioned on two components, the difference is:

$$\delta \bar{D} = b / R_2 + rT_1 / R_2,$$

where T_1, T_2, R_1, R_2 are determined for each of parts by formulae being analog to formulae (3)-(5).

5 VERIFICATION OF THE LAN MODEL OF THE APCS SEGMENT

The NPP APCS segment model has been verified by comparison of real and calculated by use of the VBR model results. Measurement of the real data has been implemented at a NPP APCS test site at the V.A. Trapeznikov Institute of Control Sciences where a prototype of the “Kudankulam” NPP APCS segment (Figure 1) has been assembled, by use of a set of hardware and software tools being identical to the real plant (Byvaikov et. al. 2006). To assign input flows from the gateway G1-G5, simulators of information flows validated for the “Kudankulam” NPP APCS were used. At the prototype, measuring the corresponding parameters involving into the flow functions (equations 1-5) has been implemented.

Let for the components of the block-scheme (Figure 1) the following input parameters are defined: the flow from the gateways G1-G5 is equivalent and bounded by the function $\gamma_{r,b}(t) = \gamma_{85e3,1e5}(t)$; for the component DB, the minimal service function $\beta_{R,T}(t) = \beta_{1e5,0.2}(t)$ and the maximal service function $\overline{\beta_{R,T}(t)} = \beta_{5e5,0.5}(t)$, $\alpha = 0.6$; for the server AB, $\beta_{R,T}(t) = \beta_{1e6,0.1}(t)$, $\overline{\beta_{R,T}(t)} = \beta_{1e7,0.3}(t)$; for the server IZ $\beta_{R,T}(t) = \beta_{1e6,0.02}(t)$, $\overline{\beta_{R,T}(t)} = \beta_{1e7,0.2}(t)$, these values has dimension of bits, bits per second, and seconds for b , (R,r) and T correspondingly. These input parameters will be considered as “normal”, which are assumed to be set for all calculations and measurements in the paper, if another is not specified. To assign the flow parameters, the flow rate (r) as well as the size of the output gateway buffer to assign heterogeneity of the flow (b) were varied.

In general, the difference between measured at the prototype and calculated (VBR model) data is not large, and does not exceed 30-50% at the work range of the input data; at marginal values, considerable mismatch (up to 100% of the measured value) are possible, what perhaps reflects a non-linearity of the investigated system.

In Figure 3 (a), there are presented character results of the modeling and real data on the maximal signal passage time from the gateway to an image at the workstation display. In the plot, the axis X represents capacity of the server element DB in percents of the parameter $R = 1e5$ bit/sec accepted

as 100%. Decreasing the capacity of the DB node has been being achieved by parallel performance on the computer of a background extra task of a required capacity. The axis Y represents the maximal delay in seconds. Each point of the real dependence in the plot has been being selected, as a maximum, at the interval 1 hour during measuring the parameter.

Figure 3 (b) presents a dependence of the maximal delay of passage on the parameter b_{G5} representing heterogeneity of the input flow from the gateway. As can be seen from equation (6), one should expect linear dependency of the maximal delay on flow heterogeneity, what is perfectly corresponded to the measured data.

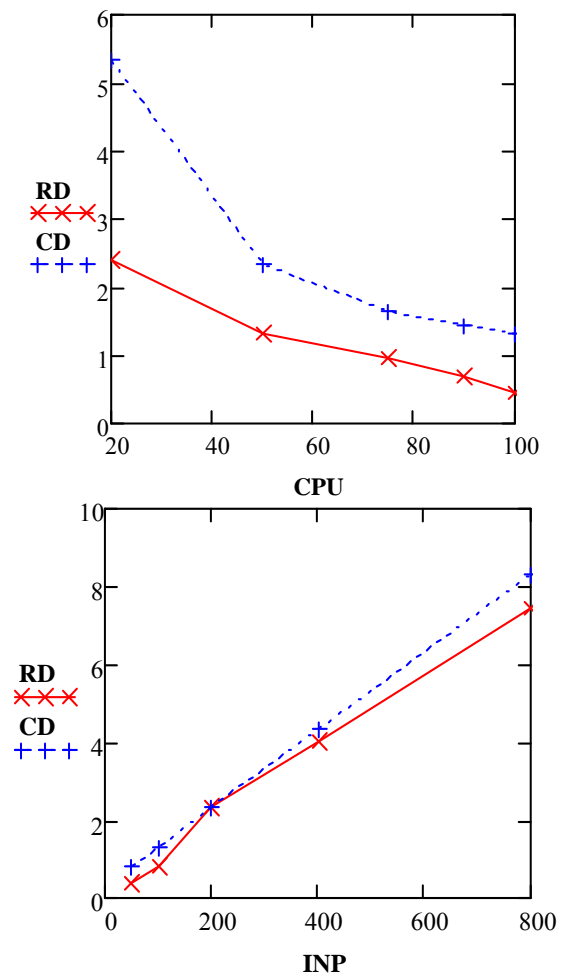


Figure 3 (a, b): To the upper (a), dependence of the value of the maximal delay (sec) of server capacity (% of maximally possible). CD – model data, RD – measured data. To the lower (b), dependence of the maximal delay (sec) of the parameter b_{G5} (Kbit). CD – model data, RD – measured data.

6 CONCLUSIONS

Within works on creating a prospective NPP APCS of the new generation, based on the method “network calculus” it was developed and verified a method of calculating time characteristics of the system under worst combination of input conditions. Such a solution will enable one to decrease NPP APCS creation time, to decrease expenses and time of validation of time characteristics of the system.

By use of the “network calculus” apparatus, a model of a NPP APCS segment has been developed, a delay of signal passage from a source to a receiver within the segment has been calculated, formulae for the maximal and minimal service functions for the NPP APCS segment have been derived. Parameters of the computer system for control signals may be symmetrically derived by analogous judgments.

REFERENCES

- Chen Peng, Dong Yue, Engang Tian, Zhou Gu., A delay distribution based stability analysis and synthesis approach for networked control systems, *Journal of the Franklin Institute*, volume 346, issue 4, pp. 349-365, May 2009.
- Le Boudec, J.-Y., Thiran P. Network Calculus: A Theory of Deterministic Queuing Systems for the Internet. Springer-Verlag, 300 pp., 2001.
- Byvaikov, E. M., Zharko, E. F., Mengazetdinov, N. E., Poletykin, A. G., Prangishvili, I. V., Promyslov, V. G. Experience from design and application of the top-level system of the process control system of nuclear power-plant, *Automation and Remote Control*, volume 67, number 5, pp. 735-747, 2006.
- Witsch, D., Vogel-Heuser, B., Faure, J.-M., Marsal G. Performance Analysis of industrial Ethernet networks by means of timed model checking, In *Proc IFAC INCOM '06 Symposium*, 2006.
- Vantanski, N. et. al. Compensating the transmission delay in networked control systems, In *14th Nordic process control workshop, NPCW '07*, Espoo, Finland, 2007.
- Hwangnam, Kim, Hou J.C. Network calculus based simulation for TCP congestion control: theorems, implementation and evaluation, In *INFOCOM '2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*. Publication Date: 7-11 March 2004, volume 4, pp. 2844-2855. ISSN 0743-166X.
- Cruz, R. L. A Calculus for Network Delay. Part I: Network Elements in Isolation, *IEEE Transactions on Information Theory*, volume IT-37, pp. 114-131, January 1991.
- Cruz, R. L. A calculus for network delay. II. Network analysis Information Theory, *IEEE Transactions on Information Theory*, volume IT-37, pp. 132-141, January 1991.
- Turner, J. New Directions in Communications (or which way to the information age?), *IEEE Communications*, volume 24, number 10, pp. 8-16, .October 1986.
- Zdarsky, N., Martinovic, F. A., Schmitt Ivan, Jens B. The DISCO Network Calculator, In *Proceedings: MMB '14th GI/ITG Conference – Measurement, Modelling and Evaluation of Computer and Communication Systems*, Dortmund 2008.

3D PATH PLANNING FOR UNMANNED AERIAL VEHICLES USING VISIBILITY LINE BASED METHOD

Rosli Omar^{1,2} and Dawei Gu²

¹*Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia*

²*Department of Engineering, University of Leicester, LE1 7RH, Leicester, U.K.*

{ro32, dag}@le.ac.uk

Keywords: 3D path planning, Visibility lines, Base line oriented visibility line, Optimization.

Abstract: In path planning, visibility graph (or visibility line (VL)) method is capable of producing shortest path from a starting point to a target point in an environment with polygonal obstacles. However, the run time increases exponentially as the number of obstacles grows, causing this method ineffective for real-time path planning. A 2D path planning framework based on VL has recently been introduced to find a 2D path in an obstacle-rich environment with low run time. In this paper we propose 3D path planning algorithms based on the 2D framework. Several steps are used in the algorithms to find a 3D path. First, a local plane is generated from a local starting point to a target point. The plane is then rotated at several pre-defined angles. At each rotation, a shortest path is calculated using 2D algorithms. After rotations at all angles have been done, the shortest one is selected. Simulation results show that the proposed 3D algorithms are capable in finding paths in 3D environments and computationally efficient, thus suitable for real time application.

1 INTRODUCTION

Uninhabited Aerial Vehicles (UAVs) are becoming more popular in accomplishing tasks in adverse environments. For example UAVs have been used for military purpose such as reconnaissance and combat as well as to perform civil tasks such as weather forecasting, environmental research, search and rescue missions and traffic control.

The advantage of UAVs in avoiding human loss brings "less" intelligence of the vehicle. In order to make UAVs more practically useful, it is important to raise the autonomy level of UAVs. Autonomy means the capability of UAVs to make its own decisions based on the available information captured by sensors, and potentially covers the whole range of the vehicle operations without human intervention (Frampton, 2008). However, autonomy technology is still in its early stage and fairly underdeveloped (<http://www.theuav.com>). It is the bottleneck for UAVs development in the future. Hence the problem of autonomy has to be addressed before the fully autonomous UAVs can be advanced. As path planning is one of the crucial factors in enhancing the autonomy level in UAVs, this paper focuses on this topic.

Researches on path planning among polygonal ob-

stacles have been around probably since the beginning of mobile robot. They have produced many methods and algorithms under several categories. Among them are geometric-based (Omar and Gu, 2009; Coleman and Wunderlich, 2008; Tian et al., 2007; Bortoff, 2000; Nilsson, 1984), grid-based (Chen et al., 1995; Lingelbach, 2004) and potential field (Garibotto and Masciangelo, 1991; Barraquand et al., 1992), to name but three. One of the popular methods is geometric-based category under which there is an approach called visibility lines (VL).

VL was first proposed by Lozano-Perez and Wesley (Lozano-Perez and Wesley, 1979) for path planning in the environments with polyhedral obstacles. Since then several researchers (Nilsson, 1984; Huang and Chung, 2004; Bygi and Ghodsi, 2006) used the method with some variants. However one major disadvantage of VL is the computational effort grows exponentially as the number of obstacles increases. To overcome such a problem, Huang and Chung (Huang and Chung, 2004) introduced Dynamic VG (DVG) which used local region to plan a path to speed up the run time. The local region was determined by the nodes that have maximum distance from a line drawn from starting point to target point called *S-G* line. In (Omar and Gu, 2009), we proposed 2D path planning algorithm which was based on VL called Base-

Line Oriented Visibility Lines(BLOVL). Also there is a sub-algorithm of BLOVL named *Core*. *Core*'s main purpose is to find a path from a two-stage process. First, a group of obstacles that lie on base line (BL) and their extension are identified. BL is similar to *S-G* line in (Huang and Chung, 2004). Second, a path is calculated using Dijkstra's algorithm. As the path planned by *Core* might not collision-free, BLOVL is used to further plan it. In addition BLOVL is designed to be used for real-time path planning.

On the other hand, 3D path planning problems have been studied extensively for many years. There were several different approaches available including Evolutionary Algorithms (EA) (Hasircioglu et al., 2008; Mittal and Deb, 2007), VL (Jiang et al., 1993) and Dubin circles (Ambrosino et al., 2006), to name but three. In (Hasircioglu et al., 2008) EA and B-spline curves for off-line 3D path planning were used. To increase the performance of the path, the number of generations had to be increased hence increased the run-time. Like (Hasircioglu et al., 2008), Mittal and Deb (Mittal and Deb, 2007) presented an off-line path planner with multi-objective EA and B-spline. The results of their work were several optimal 3D paths. (Ambrosino et al., 2006) used Dubin circles to first obtain estimate of a 3D path. Then the path was divided into three sub-paths. However, (Ambrosino et al., 2006) assumed that no obstacle to be avoided during the path generation.

In this paper, we propose a 3D path planning algorithm, $BLOVL_{3D}$ which consists of several sub-algorithms namely *BasePlane*, *Rotate_{3D}* as well as BLOVL. Basically $BLOVL_{3D}$ find the 3D path from a series of rotations of local planes. This algorithm and its sub-algorithms have been realized into a Matlab's graphical user interface (GUI) environment for simulation purpose to evaluate its effectiveness. The rest of this paper is organized as follows. Section 2 reviews the 2D path planning using *Core* and BLOVL algorithms. Section 3 explains our proposed 3D path planning algorithm in details. Section 4 shows an example of results from the proposed algorithms. In Section 5 we demonstrate the simulation results in term of run time. Section 6 concludes the paper.

2 2D PATH PLANNING

In order to make path calculation faster by visibility lines (VL) means, the number of obstacles used in the calculation has to be minimized. Thus *Core* has been designed to perform this task. Figure 1 illustrates the process of *Core* while **Algorithm 1** shows the steps of it.

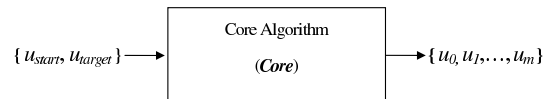


Figure 1: Core algorithm.

Algorithm 1: Core.

- 1: Create a base line (BL) from starting point, u_{start} to target point, u_{target}
 - 2: Construct a set of nodes, N_S , from the four corners of each obstacle that lies on the base line and their extensions, including u_{start} and u_{target}
 - 3: Create a cost matrix, C_M from N_S
 - 4: Find local path, $U(u_0, \dots, u_m)$ from C_M using Dijkstra's algorithm where $u_0 = u_{start}$ and $u_m = u_{target}$
-

Core begins with creating a base line (BL) from a local starting point, u_{start} to a target point, u_{target} . The obstacles that lie on BL and their extension, O_{BL} will be used for path calculation. The idea on how O_{BL} is identified is illustrated in Figure 2. In the figure, the obstacles that intersect with BL are numbered as 1 and 2 while the extended obstacles are 3 and 4. Obstacle 5 is ignored as it is neither on BL nor an extension of the obstacles on BL. As a result O_{BL} contains obstacles with the number of 1, 2, 3 and 4.

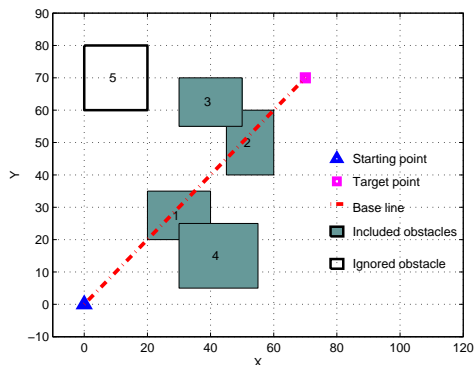


Figure 2: Obstacles identification using *Core*.

In Step 2 of *Core*, O_{BL} is used to build up a set of nodes stored in N_S . Then each pair of mutually visible nodes in N_S is connected by a line segment and given a cost based on its Euclidean distance. On the contrary, two mutually-invisible nodes are given infinity costs and are thus ignored. Based on N_S , in the next step of *Core*, a cost matrix, C_M is created. C_M stores the indexes of paired nodes and the lines segment Euclidean distances (costs). If all pairs of mutually-visible nodes are connected together by line segments, they will form a plane with zero altitude as shown in Figure 3.

Using C_M , Dijkstra's algorithm will then be ap-

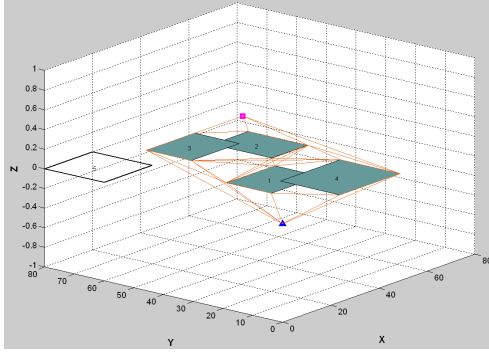


Figure 3: A plane formed by visibility lines with zero altitude.

plied in Step 4 of *Core* to find a local optimal path, U by minimizing the total path length from u_{start} to u_{target} . To ensure that the path is collision-free and capable to be used for real-time path planning, *Core* has to be associated with BLOVL as shown in **Algorithm 2**. Figure 4 shows the BLOVL process.

Algorithm 2: BLOVL.

- 1: Set $j = 0$ and $w_j = p_{start}$
 - 2: **while** $w_j \neq p_{target}$ **do**
 - 3: set $u_{start} = w_j$ and $u_{target} = p_{target}$
 - 4: call *Core*
 - 5: **if** $m = 1$ **then**
 - 6: set $w_{j+1} = u_1$
 - 7: **else**
 - 8: set $u_{target} = u_1$
 - 9: goto line 4
 - 10: **end if**
 - 11: set $j = j + 1$
 - 12: **end while**
-

The first step of BLOVL is setting the current starting point/waypoint, w_j to p_{start} where j is initialized to 0. Then w_j is compared with the p_{target} . If w_j is or at the vicinity of p_{target} , then the process is stopped. Otherwise w_j and p_{target} will be defined as u_{start} and u_{target} respectively. With u_{start} and u_{target} being the input, *Core* is called to find a local shortest path, U . *Core* will be called again if the number of elements in U is greater than 2 as this shows that there are obstacles between u_{start} and u_{target} . Note that the element number in U is indicated by m . If $m = 1$, it means that U has 2 elements and the resulted path is unobstructed from u_{start} and u_{target} , and the next waypoint w_{j+1} will be set to u_1 . This process will be repeated until w_j is or near to p_{target} . Notice that the resulted path from BLOVL consists of a set of global waypoints, $W = w_0, \dots, w_{n-1}$.

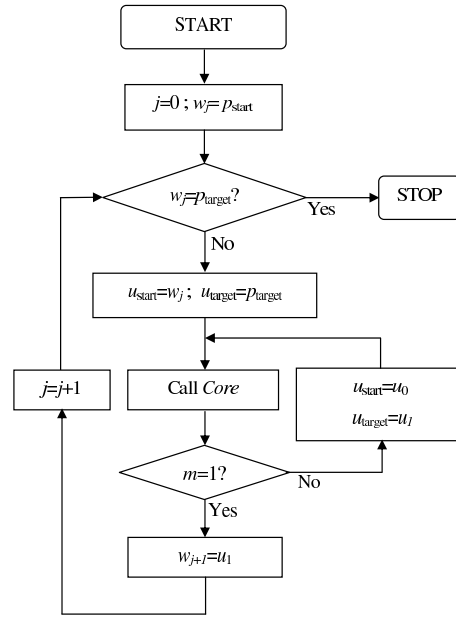


Figure 4: BLOVL process.

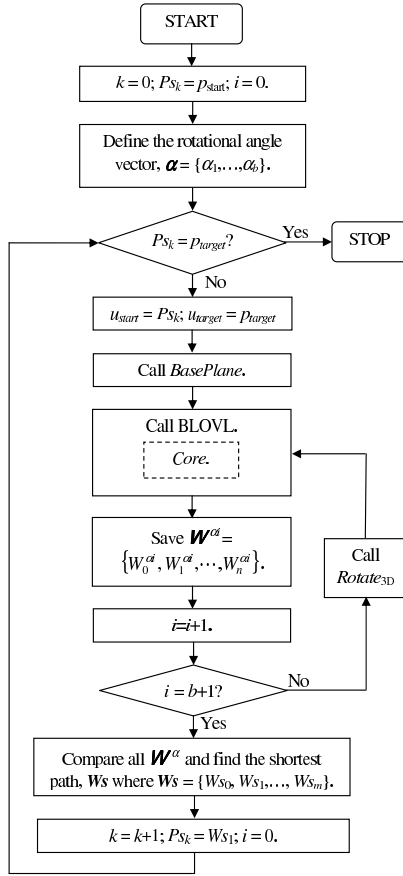
3 ALGORITHM FOR 3D PATH PLANNING

In practice UAV flies in 3D environments. To ensure that the UAV's paths are free from collision in such environments, path in 3D has to be planned. For such a purpose, we have developed 3D path planning algorithm, BLOVL_{3D} which consists of several sub-algorithms i.e. *BasePlane*, *Rotate3D* and BLOVL. BLOVL_{3D} uses rotational planes to find 3D paths. Figure 5 illustrates the process of BLOVL_{3D} and **Algorithm 3** shows its steps.

BLOVL_{3D} starts with initializing several necessary parameters i.e. $k = 0$, $Ps_k = p_{start}$ and $i = 0$. Notice that Ps is the global path with several waypoints that will be built-up along the process. The final value of k will determine the number of waypoints in Ps . i represents the index of the rotational angles.

In the next step of BLOVL_{3D}, the vector of rotational angle, α is defined. It consists of b number of angles. α will determine at which angle the plane will be rotated. Next Ps_k is compared with p_{target} . If Ps_k is p_{target} or at the vicinity of p_{target} the process will be stopped. Otherwise u_{start} will be set to Ps_k while u_{target} is p_{target} . Note that u_{start} and u_{target} are necessary for *Core* of BLOVL in the later stage of the algorithm.

In case of Ps_k is not the p_{target} or not at the vicinity of it, *BasePlane* which is shown in **Algorithm 4** is then called. *BasePlane* generates a local plane, $Px'y'_{u_{start}}$. As shown in **Algorithm 4**, the local


 Figure 5: BLOVL_{3D} process.

plane generated by *BasePlane* always change with the change in u_{start} hence it is named $Px'y'_{u_{start}}$.

To generate $Px'y'_{u_{start}}$ using *BasePlane*, first a base line, BL_{3D} is drawn from u_{start} to u_{target} . As u_{start} and u_{target} have different altitude, the angle β between BL_{3D} and the horizontal global plane, $Pxy_{u_{start}}$ can be calculated. BL_{3D}^\perp that intersects u_{start} and orthogonal to BL_{3D} is then defined. With BL_{3D} and BL_{3D}^\perp as the x - and y -axis respectively, $Px'y'_{u_{start}}$ that lies at β degree from $Pxy_{u_{start}}$ is defined. Then the coordinate of obstacles lying on $Px'y'_{u_{start}}$ is projected accordingly. As $Px'y'_{u_{start}}$ has been defined, next is to rotate the plane by α_i degree to find a local optimal path, W^{α_i} from u_{start} to u_{target} . Rotating this plane is performed by *Rotate3D* while finding W^{α_i} is accomplished by BLOVL and *Core*. **Algorithm 5** shows *Rotate3D*.

While $i < b + 1$, i is increased by 1 and α_i is updated accordingly and *Rotate3D* and BLOVL are kept called to find W^{α_i} . When $i = b + 1$, all paths (and their costs) that have been stored in W^α are compared with each other and path with the lowest cost, Ws is then selected. Notice that Ws consists of $\{Ws_0, \dots, Ws_{n-1}\}$ and the waypoints in Ws are ac-

ording to the global coordinate system.

In the next step, the index of global waypoints, k is increased by 1 and the next global waypoint, Ps_k is updated to be the second waypoint of the shortest path i.e. Ws_1 . i then is initialized back to 0 and the process as described above are repeated until Ps_k is or at the vicinity of p_{target} .

Algorithm 3: BLOVL_{3D}.

- 1: Set $k = 0$, $Ps_k = p_{start}$ and $i = 0$.
 - 2: Define the rotational angle vector, α .
 - 3: **while** $Ps_k \neq p_{target}$ **do**
 - 4: $u_{start} = Ps_k$; $u_{target} = p_{target}$.
 - 5: call *BasePlane* to generate local plane, $Px'y'_{u_{start}}$.
 - 6: **while** $i \neq b + 1$ **do**
 - 7: call BLOVL.
 - 8: Save waypoints, W^{α_i} generated by BLOVL.
 - 9: Increase i by 1.
 - 10: Rotate $Px'y'_{u_{start}}$ by α_i degree.
 - 11: **end while**
 - 12: Compare all paths in W^α and find the shortest, Ws . $Ws = \{Ws_0, \dots, Ws_o\}$.
 - 13: Increase k by 1 and update $Ps_k = Ws_1$. Initialise i to .
 - 14: **end while**
-

Algorithm 4: BasePlane.

- 1: Draw a base line, BL_{3D} connecting u_{start} and u_{target} . Find out the angle β between BL_{3D} and the horizontal plane $Pxy_{u_{start}}$, which contains u_{start} .
 - 2: Define a local plane, $Px'y'_{u_{start}}$, formed by BL_{3D} and the straight line, BL_{3D}^\perp which passes u_{start} , lies on $Pxy_{u_{start}}$ and is orthogonal to BL_{3D} .
 - 3: Define a local coordinate system on $Px'y'_{u_{start}}$, with u_{start} as the origin, BL_{3D}^\perp as x -axis and BL_{3D} as y -axis. Establish the coordinate transformation between this local coordinate system and the global one.
 - 4: Project the obstacles on $Px'y'_{u_{start}}$.
-

Algorithm 5: Rotate3D.

- 1: Rotate the plane $Px'y'_{u_{start}}$ by α_i degree.
 - 2: Find out the coordinate transformation between the new local system and the global one.
 - 3: Project the obstacles on the new $Px'y'_{u_{start}}$ plane.
-

4 EXAMPLE

A random scenario with 150 obstacles was generated as shown in Figure 6. Each obstacle was

numbered and given a random height. Starting point's altitude was set to 20 while the target point's altitude was 150. The planes were rotated at 0, 10, 30, 45, 60, 120, 135, 150, 170. The resulted path had the waypoints as shown in Table 1.

Table 1: The waypoints.

Number	X	Y	Z
1	26.48	18.29	20.00
2	143.76	171.24	41.05
3	394.24	378.76	69.72
4	576.76	581.24	98.77
5	817.76	834.25	133.93
6	959.76	960.92	150.00

Figure 7 and Figure 8 show the top and 3D views of the resulted path respectively.

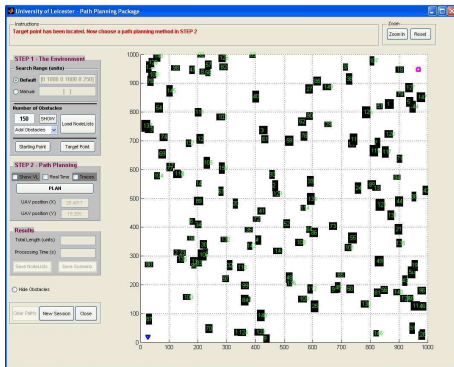


Figure 6: A random scenario with obstacles shown in black. The blue rectangle is the starting point and the square magenta is the target point.

5 SIMULATION RESULTS

Scenarios with several numbers of obstacles were simulated randomly to evaluate the performance of the proposed 3D path planning algorithms. The number of obstacles used were 50, 75, 100, 125, 150, 175 and 200. To increase the reliability of the results, ten different random scenarios were generated from each number of obstacles. The simulations were performed on Intel's 2.4Ghz Core 2 Duo processor with 2GB DDR2 RAM. As no data for other 3D algorithms available in the literature using the same scenarios as we used here, no comparison was done. Thus we compared the proposed 3D path planning algorithm performance with that of 2D that was introduced by (Omar and Gu, 2009) as both algorithms were designed for such scenarios. The results of the simulation were recorded in Table 2.

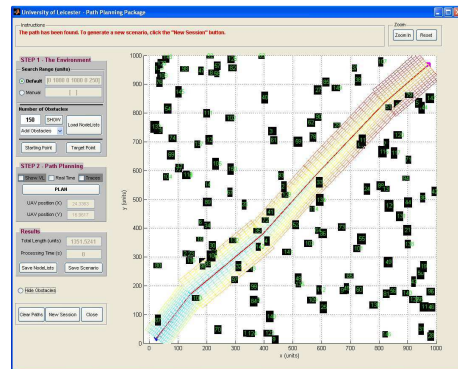


Figure 7: Top view of the resulted path (in red) with the rotated planes.

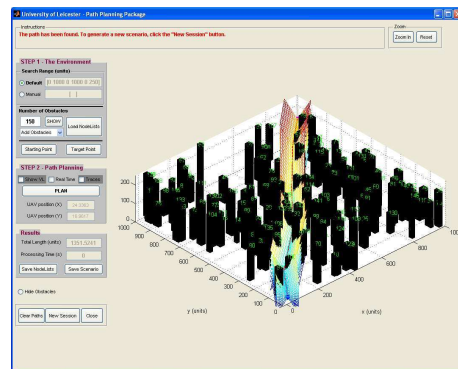


Figure 8: 3D view of the resulted path with the rotated planes.

From Table 2, using 2D algorithms no substantial growth in run time as the number of obstacles were increased while by 3D algorithms the calculation time increased quite significantly. This is due to (i) increased number of obstacles on the base line which results in increased number of waypoints. (ii) number of angles used to generate the rotational planes. If the resulted path has k waypoints and the rotational angle vector, α consists of b number of angles, the processing time might be $(b - 1) \times k$ longer than that of 2D path planning algorithms.

Table 2: Comparison of 2D and 3D path planning algorithms performance (in sec).

Number of obstacles	2D Algorithms			3D Algorithms		
	Min	Max	Ave	Min	Max	Ave
50	0.01	0.25	0.13	0.65	1.84	1.27
75	0.02	0.20	0.15	1.03	6.93	2.63
100	0.04	0.28	0.21	1.97	8.14	4.08
125	0.06	0.39	0.27	2.59	17.54	8.77
150	0.08	0.43	0.35	5.18	22.16	11.63
175	0.15	0.83	0.50	6.82	35.79	21.13
200	0.19	0.98	0.62	7.16	48.77	26.42

6 CONCLUSIONS

As visibility line (VL) method is effective in producing path with shortest length, it has been used to develop a three-dimensional (3D) path planning algorithm, BLOVL_{3D}. BLOVL_{3D} governs *BasePlane*, *Rotate*_{3D} as well as Base Line Oriented Visibility Line (BLOVL) algorithms to find a 3D path. *BasePlane* algorithm is used to establish a local plane. Next *Rotate*_{3D} algorithm rotates the plane. At each rotation of the plane, a path with lowest cost is calculated by BLOVL and recorded. After the local plane has been rotated at all angles, the resulted paths are compared to each other and the shortest one will be selected. The process continues with a new starting point which is the second waypoint of the previous shortest path. The process is stopped if the target point has been reached. Simulations results show that BLOVL_{3D} and its sub-algorithms are capable to effectively find sub-optimal paths in term of path length in 3D environments and is very promising to be applied in real time 3D path planning.

REFERENCES

- Ambrosino, G., Ariola, M., Ciniglio, U., Corraro, F., Pironti, A., and Virgilio, M. (2006). Algorithms for 3d uav path generation and tracking. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 5275–5280. IEEE.
- Barraquand, J., Langlois, B., and Latombe, J.-C. (1992). Numerical potential field techniques for robot path planning. In *Proceedings of the IEEE Transactions on Systems, Man and Cybernetics*, pages 224–240. IEEE.
- Bortoff, S. (2000). Path planning for uavs. In *Proceedings of the American Control Conference Chicago Illinois*, pages 364–368. ACC.
- Bygi, M. and Ghodsi, M. (2006). 3d visibility graph. In *12th CSI Computer Conference (CSICC 2006)*. Computer Society of Iran.
- Chen, D., Szczerba, R., and Uhran, J. (1995). Planning conditional shortest paths through an unknown environment: A framed-quadtree approach. In *Proceedings of the International Conference on Intelligent Robots and Systems*, pages 33–38. IEEE.
- Coleman, D. and Wunderlich, J. (2008). O3: An optimal and opportunistic path planner (with obstacle avoidance) using voronoi polygons. In *International Workshop on Advanced Motion Control*, pages 371–376. IEEE.
- Frampton, R. (2008). Uav autonomy. *The Journal for Defence Engineering & Science*, 1(Summer 2008):28–31.
- Garibotto, G. and Masciangelo, S. (1991). Path planning using the potential field approach for navigation. In *Fifth International Conference on Advanced Robotics*. IEEE.
- Hasircioglu, I., Topcuoglu, H., and Ermis, M. (2008). 3-d path planning for the navigation of unmanned aerial vehicles by using evolutionary algorithms. In *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, pages 1499–1506. ACM.
- Huang, H. and Chung, S. (2004). Dynamic visibility graph for path planning. In *Proceeding of International Conference on Intelligent Robots and Systems*. IEEE.
- Jiang, K., Seneviratne, L., and Earles, S. (1993). Finding the 3d shortest path with visibility graph and minimum potential energy. In *Proceeding of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 679–684. IEEE.
- LingelBach, F. (2004). Path planning using probabilistic cell decomposition. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 467–472. IEEE.
- Lozano-Perez, T. and Wesley, M. (1979). An algorithm for planning collision-free paths among polyhedral obstacles. In *Continuum ACM*, pages 560–570. ACM.
- Mittal, S. and Deb, K. (2007). Three-dimensional off-line path planning for uavs using multiobjective evolutionary algorithms. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 3195–3202. IEEE.
- Nilsson, N. (1984). Shakey the robot. Technical Report JReport 323, 333 Ravenswood Avenue, Menlo Park, CA 94025-3493.
- Omar, R. and Gu, D.-W. (2009). Visibility line based methods for uav path planning. In *ICROS-SICE International Joint Conference*. ICROS-SICE.
- Tian, Y., Yan, L., Park, G., S.Yang, Kim, Y., Lee, S., and Lee, C. (2007). Application of rrt-based local path planning algorithm in unknown environment. In *Proceeding of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 456–460. IEEE.

NEW PROPOSAL FOR A MULTI-OBJECTIVE TECHNIQUE USING TRIBES AND TABU SEARCH

Nadia Smairi, Sadok Bouamama

*National School of Computer Sciences, University of Manouba, Manouba 2010, Tunisia
nadia.smairi@gmail.com, Bouamama_sadok@yahoo.fr*

Khaled Ghedira, Patrick Siarry

*High Institute of Management, University of Tunis, Tunisia
University of Paris 12 (LiSSi, E.A. 3956), France
khaled.ghedira@isg.rnu.tn, siarry@univ-paris12.fr*

Keywords: Particle Swarm Optimization, Tribes, Tabu Search, Multi-objective Optimization.

Abstract: The aim of this paper is to present a new multi-objective technique which consists on a hybridization between a particle swarm optimization approach (Tribes) and tabu search technique. The main idea of the approach is to combine the high convergence rate of Tribes with a local search technique based on Tabu Search. Besides, in our study, we proposed different places to apply local search: the archive, the best particle among each tribe and each particle of the swarm. As a result of our study, we present three versions of our hybridized algorithm. The mechanisms proposed are validated using twelve different functions from specialized literature of multi-objective optimization. The obtained results show that using this kind of hybridization is justified as it is able to improve the quality of the solutions in the majority of cases.

1 INTRODUCTION

One of many drawbacks of evolutionary algorithms is that each one of them has many parameters to be tuned each time we want to solve a different problem. Tribes, an adaptative Particle Swarm Optimization (PSO) technique, has the advantage to be designed as a black box; the user defining only the search space, the function to minimize, the required accuracy and a maximum number of evaluations. At the beginning, it was designed to treat mono-objective problems. The aim of this work is to design a competitive multi-objective algorithm free from parameters based on Tribes. However, it has become evident that the concentration on a sole metaheuristic is restrictive. A skilled combination of Tribes with other optimization techniques can provide a more efficient behaviour and higher flexibility when dealing with the real-world problems. Therefore, in this paper, we propose a new multi-objective technique based on Tribes and Tabu Search (TS). In fact, TS is used to cover widely the solution space and to avoid the risk of trapping in non Pareto solutions and Tribes is used to accelerate

the convergence. In our study, we use twelve well-known multi-objective test functions in order to find the best one from the proposed techniques and to justify the use of the local search.

In section 2 of this paper we present the existing multi-objective PSO techniques. In section 3, we consider Tribes. In addition, in section 4, we present our proposed approach. Then comparative results are described in section 5, from which conclusions are drawn in section 6.

2 STATE OF ART

In the last few years, several PSO algorithms have been proposed to tackle the multi-objective optimization problem. Here we briefly review the most relevant of them.

Parsopoulos and Vrahatis (2002) propose three different types of aggregation: a classic linear aggregation, for which the weights are fixed, a dynamic aggregation where the weights are gradually modified during the treatment and an

aggregation the weights of which are brutally modified during the treatment.

Hu, Eberhart and Shi (2003) propose an algorithm optimizing each time one single objective using a lexicographical order.

The VEPSO strategy was introduced by Parsopoulos, Tasoulis and Vrahatis (2004). It presents an adaptation of VEGA to the particle swarm optimization.

Moore and Chapman (1999) propose an algorithm based on the Pareto dominance and a PSO algorithm with a circular topology of the neighbourhood. In this approach, the choice of the personal guide, for every particle, is arbitrarily made from a list containing the not dominated positions that are found.

Ray and Liew (2002) propose a PSO algorithm using the Pareto dominance. They combine evolutionary techniques with those of the OEP. They also use an operator of density on the neighbourhood to promote the density in the swarm.

This approach, proposed by Coello and Lechuga (2002), is based on having an external archive to store the not dominated positions. Furthermore, the updates of the archive are executed considering a geographical system which decomposes the space of the objectives to a set of hypercubes. The archive is also used to identify a leader which will drive the search.

The authors propose a multi-objective PSO algorithm, called DOPS in which several techniques are integrated for the selection of the leaders and the update of archive (Bartz-Beielstein, Limbourg, Parsopoulos, Vrahatis, Mehnen and Shmitt, 2003).

Quintero, Santiago and Coello (2008) suggest a hybridization of a PSO algorithm with local search techniques such as scatter search and rough sets theory.

The proposed algorithm (Sierra and Coello, 2005) is based on the dominance of Pareto: every not dominated position presents a potential candidate to be selected as a leader. A crowd function is also used to filter all the leaders. This approach (Sierra and Coello, 2007) also integrates the concept of the ϵ -dominance to fix the size of the archive.

The author has developed a multi-objective version of Tribes. In fact, Mo-Tribes use an approach based on the Pareto dominance. The not dominated particles are stored in an external archive which size and updates are automatically defined. Furthermore, the variety is maintained thanks to a criterion of maximization of the crowd distance and also thanks to the multiple restarts of the swarm. The results of Mo-Tribes are very encouraging (Cooren, 2008).

3 TRIBES

Tribes is a PSO algorithm that works in an autonomous way. Indeed, it is enough to describe the problem to be resolved and the way of making it at the beginning of the execution. Then, it is the role of the program to choose the strategies to be adopted (Clerc, 2006).

At the beginning, we start with a single particle forming a tribe. After the first iteration, the first adaptation takes place and we generate a new particle which is going to form a new tribe, while keeping in touch with the generative tribe. In the following iteration, if the situation of both particles does not improve, then every tribe creates two new particles: we form a new tribe containing four particles. In this way, if the situation deteriorates, then the size of the swarm grows (creation of new particles). However, if we are close to an optimal solution, the process is reversed and we begin to eliminate particles, even tribes. In fact, the removal or the generation of a particle are not arbitrary. The removal of a particle consists in eliminating a particle without risking the missing of the optimal solution. For that purpose, only the good tribes are capable of eliminating their worst elements. The creation of a particle is made for bad tribes as they need new information to improve their situations.

4 OUR APPROACH

4.1 Preliminary Study

The adaptation of Tribes to the multi-objective optimization consists in using the Pareto dominance to respect the completeness of every objective and to add an external archive to save the found not dominated solutions. Furthermore, as the PSO algorithm, Tribes can be considered neither a global optimization algorithm nor a local optimization one (Bergh, 2002). Therefore, the hybridization between Tribes and a local search algorithm can be considered as a competitive approach to handle difficult problems of multi-objective optimization. In order to improve the capacity of exploitation of Tribes, we apply a local search technique: TS. In fact, the local search is not going to be inevitably applied in a canonical way that is on all the particles of the swarm: we also propose two other manners, the first one consists in applying the local search only among the best particle of every tribe. The second one consists in applying it among the particles of the archive. We shall have then three versions of the algorithm.

The first one consists in applying the TS only to the particles of the archive which are situated in the least crowded zones. Let us note that, in this case, the local search is not applied unless the archive is full so that some time is allowed to the information to propagate in the swarm.

```

Begin
  Swarm initialization
  Swarm evaluation
  Archive initialization
  While  $f < f_{max}$ 
    For each tribe
      For each particle  $i$ 
        Determination of the state of the particle
        Choice of the strategy of movement
        Choice of the informer
        Update of the position
        Evaluation
        Update of  $p_i$  (best position visited by  $i$ )
        Update the best particle of the tribe
        Update the archive
      EndFor
    EndFor
    If criterion of adaptation verified
      Determination of the quality of the tribe
      Adaptation of the swarm
      Update of the adaptation criterion
    EndIf
    For each particle of the archive situated in
    the least crowded zones
      TS (stopping criterion)
    EndFor
  EndWhile
End
    
```

Figure 1: TS-TribesV1 pseudo-code.

The second version of the algorithm consists in applying the TS only to the best particles of the tribes. In fact, we consider that those particles are situated in promising zones and probably they need further intensification to find out other solutions.

The third version consists in applying the TS to all the particles of the swarm. It is made at the moment of the swarm adaptation.

The detailed description of TS-TribesV2 and TS-TribesV3 was omitted due to space restrictions.

4.2 Updating the External Archive

The update of the archive consists in adding all the not dominated particles to the archive and deleting the already present dominated ones. If the number of particles in the archives exceeds a fixed number, we apply a crowd function to reduce the size of the archive and to maintain its variety. Indeed, Crowd divides the objective space into a set of hypercube.

4.3 Choosing the Particle Informer

The choice of the particle informer or guide is similar to the case of mono-objective Tribes. Indeed, if we take a particle which is not the best of its tribe, his guide is then the best particle of the tribe. If we consider, on the other hand, the best particle of a given tribe, the informer is then some random particle from the archive.

4.4 Hybridizing Tribes with TS

The TS is introduced by Glover. It consists in the examination of a neighbourhood of a current solution x and retains the best neighbour x_0 even if x_0 is worse than x . However, this strategy can pull cycles. To prevent this kind of situation from appearing, we store the k last visited configurations in a short-term memory and we forbid to hold any other configuration which is already a part of it.

However, TS is essentially intended for the resolution of the combinatorial problems. Few works considered its adaptation for the continuous optimization. Among whom we can mention the approach of Chelouah and Siarry (2000). In that case, this method is similar to the classic TS. The difference lies essentially in the generation of the neighbourhood. It is necessary to define first of all a way to discretize the search space. In fact, the neighbourhood is defined by using the concept of "ball". A ball $B(x, r)$ centered on x (current solution) with radius r . To obtain a homogeneous exploration of the space, we consider a set of balls centered on the current solution x with radius $r_0, r_1, r_2, \dots, r_n$. Hence the space is partitioned into concentric crowns. The n neighbours of x are obtained by random selection of a point which does not belong to the tabu list inside each crown C_i , for i varying from 1 to n . Finally, we select the best neighbour x' even if it is worse than x and we insert it in the tabu list.

5 EXPERIMENTATIONS AND RESULTS

5.1 Test Functions

In order to compare the proposed techniques, we perform a study using twelve well-known test functions taken from the specialized literature on evolutionary algorithms. The detailed description of these functions was omitted due to space restrictions. However, all of them are unconstrained, minimization and have between 3 and 30 decision

variables. Indeed, we fix the maximal size of the archive to 100 for the two-objective functions and to 150 to the three-objective ones. We also varied the size of the neighbourhood for the TS algorithm: 5, 10 and 20. Moreover, we fix the maximal number of evaluations in the experimentations to $5e+4$.

Table 1: Properties of the test functions.

Test functions	Objective	Modality	Geometry
Oka2	f_1	Uni-modal	Concave
	f_2	Multi-modal	
Sympart	$f_{1,2}$	Multi-modal	Concave
S_ZDT1	$f_{1,2}$	Uni-modal	Convex
S_ZDT2	$f_{1,2}$	Uni-modal	Concave
S_ZDT4	f_1	Uni-modal	Convex
	f_2	Multi-modal	
R_ZDT4	$f_{1,2}$	Multi-modal	Convex
S_ZDT6	$f_{1,2}$	Multi-modal	Concave
S_DTLZ2	$f_{1,3}$	Uni-modal	Concave
S_DTLZ3	$f_{1,3}$	Multi-modal	Concave
WFG1	$f_{1,3}$	Uni-modal	Convex
WFG8	$f_{1,3}$	Uni-modal	Concave
WFG9	$f_{1,3}$	Multi-modal	Concave

5.2 Metrics of Comparison

For assessing the performance of the algorithms, there are many existent unary and binary indicators measuring quality, diversity and convergence. In the literature, there are many proposed combination in order to perform a convenient study and comparison. We choose the combination of two binary indicators that was proposed in (Knowles, Thiele and Zitler, 2006): R indicator and hypervolume indicator.

5.2.1 R indicator (I_{R2})

It computes the difference between the maximum value of the augmented Tchebycheff utility function of the reference set and the obtained solutions from the procedure.

5.2.2 Hypervolume Indicator (I_H)

The hypervolume indicator measures the hypervolume of that portion of the objective space that is weakly dominated by an approximation set A, and is to be maximized. Here we consider the hypervolume difference to a reference set R; where smaller values correspond to higher quality.

5.2.3 Results

The binary indicators used to make the comparison measure both convergence and diversity. The results

regarding the R indicator are given in tables 2, 3 and 4 (R can take values between -1 and 1 where smaller values correspond to better results). The hypervolume difference is given for all test functions in table 5, 6 and 7. Again, smaller values mean better quality of the results because the difference to a reference set is measured.

For both indicators, we present the summary of the results obtained. In each case, we present the average of I_{R2} and hypervolume measures over 10 independent runs. These values are given for the different sizes of neighbourhood. According to these tables, we notice that:

- The found fronts for test functions S_ZDT1, S_ZDT2 and S_DTLZ2 are very close to the reference set (for all the versions). Moreover, the found fronts for test functions OKA2, WFG8 and WFG9 are better than the proposed reference fronts (for all the versions).
- Bad performance behaviour is noticed for S_ZDT4 and R_ZDT4 for all the versions except TS-TribesV3. We note that bad convergence behaviour is detected also with another PSO algorithm for ZDT4 in (Hu, Eberhart and Shi, 2003).
- TS-TribesV1 outperforms generally the other versions except for test functions S_ZDT4 and R_ZDT4 where TS-TribesV3 gives the best results.
- The neighbourhood size has no big effect on the performances of the considered algorithms. In fact, they keep the same tendency with the neighbourhood size variation.

Finally, we recapitulate that TS-Tribes is very competitive as it supports both intensification and diversification. In fact, the choice of particle's informer is done in order to accelerate the swarm's convergence towards the search space zones where are situated the archive's particles. This can be considered as an intensification process. Moreover, the archive's updating is done thanks to the Crowd function that maintains the archive's diversity. This can be considered as a diversification process. Indeed, TS supports both intensification and diversification. The good neighbourhood exploration intensifies the search towards specific zones in the search space. Besides, the TS mechanisms such as tabu list allow avoiding the risk of trapping in non Pareto solutions.

Table 2: Results for R indicator (neighbourhood size = 5).

Test Functions	TS-TribesV1	TS-TribesV2	TS-TribesV3
OKA2	-1.23e-3	-1.22e-3	-1.21e-3
Sympart	6.74e-5	2.91e-5	8.38e-5
S_ZDT1	7.21e-4	1.26e-3	1.05e-3
S_ZDT2	4.01e-5	1.48e-3	3.27e-5
S_ZDT4	2.84e-3	4.84e-3	4.10e-3
R_ZDT4	8.21e-3	2.24e-3	1.46e-2
S_ZDT6	4.50e-3	7.78e-3	2.19e-3
S_DTLZ2	2.52e-4	2.19e-4	2.70e-4
S_DTLZ3	4.24e-4	2.99e-4	7.68e-4
WFG1	2.44e-2	3.93e-2	4.94e-2
WFG8	-2.01e-2	-1.18e-2	-2.25e-3
WFG9	-6.73e-3	-6.10e-3	-2.63e-3

Table 3: Results for R indicator (neighbourhood size = 10).

Test Functions	TS-TribesV1	TS-TribesV2	TS-TribesV3
OKA2	-1.15e-3	-1.03e-3	-1.02e-3
Sympart	2.99e-5	3.20e-5	4.68e-5
S_ZDT1	5.17e-4	1.19e-3	1.21e-3
S_ZDT2	3.72e-5	1.02e-3	1.23e-4
S_ZDT4	2.82e-3	8.78e-3	1.68e-4
R_ZDT4	4.24e-3	3.35e-3	2.38e-3
S_ZDT6	3.05e-3	8.79e-3	2.42e-3
S_DTLZ2	1.69e-4	2.32e-4	2.13e-4
S_DTLZ3	2.08e-4	3.37e-4	4.72e-4
WFG1	2.49e-2	4.39e-2	4.89e-2
WFG8	-1.69e-2	-1.22e-2	-2.26e-3
WFG9	-9.21e-3	-4.93e-3	-8.44e-3

Table 4: Results for R indicator (neighbourhood size = 20).

Test Functions	TS-TribesV1	TS-TribesV2	TS-TribesV3
OKA2	-1.01e-3	-1.01e-3	-1.03e-3
Sympart	4.03e-5	4.84e-5	5.40e-5
S_ZDT1	6.26e-4	1.26e-3	1.26e-3
S_ZDT2	3.93e-5	1.35e-3	3.95e-5
S_ZDT4	2.31e-3	9.67e-3	2.53e-6
R_ZDT4	8.30e-3	2.78e-3	1.08e-4
S_ZDT6	3.37e-3	6.02e-3	4.32e-3
S_DTLZ2	1.52e-4	1.71e-4	2.41e-4
S_DTLZ3	1.43e-4	2.96e-4	7.36e-4
WFG1	2.88e-2	4.33e-2	3.02e-2
WFG8	-1.96e-2	-1.32e-2	-8.68e-3
WFG9	-1.18e-2	-7.59e-3	-8.26e-4

Table 5: Results for $I_{\bar{H}}$ (neighbourhood size = 5).

Test Functions	TS-TribesV1	TS-TribesV2	TS-TribesV3
OKA2	-1.23e-3	-1.22e-3	-1.21e-3
Sympart	2.01e-4	8.80e-5	2.49e-4
S_ZDT1	5.81e-4	5.13e-3	4.59e-3
S_ZDT2	3.40e-4	3.87e-3	3.08e-4
S_ZDT4	7.89e-3	1.38e-2	1.15e-2
R_ZDT4	1.47e-2	6.85e-3	4.30e-2
S_ZDT6	6.51e-3	1.65e-2	4.67e-3
S_DTLZ2	1.67e-3	8.78e-4	1.81e-3
S_DTLZ3	5.62e-3	8.30e-4	2.12e-2
WFG1	1.65e-1	2.08e-1	2.58e-1
WFG8	-1.25e-1	-7.21e-2	-1.42e-2
WFG9	-4.06e-2	-3.23e-2	-3.86e-3

Table 6: Results for $I_{\bar{H}}$ (neighbourhood size = 10).

Test Functions	TS-TribesV1	TS-TribesV2	TS-TribesV3
OKA2	-1.20e-3	-1.20e-3	-1.20e-3
Sympart	8.95e-5	9.47e-5	1.41e-4
S_ZDT1	2.45e-3	5.16e-3	5.11e-3
S_ZDT2	3.51e-4	2.74e-3	5.28e-4
S_ZDT4	7.84e-3	2.52e-2	4.57e-3
R_ZDT4	1.52e-2	7.07e-3	1.04e-3
S_ZDT6	6.38e-3	1.93e-2	5.21e-3
S_DTLZ2	8.09e-4	8.78e-4	1.81e-3
S_DTLZ3	6.10e-4	4.88e-3	1.07e-2
WFG1	1.70e-1	2.56e-1	2.55e-1
WFG8	-1.09e-1	-7.03e-2	-1.30e-2
WFG9	-2.29e-2	-3.01e-2	-5.43e-3

Table 7: Results for $I_{\bar{H}}$ (neighbourhood size = 20).

Test Functions	TS-TribesV1	TS-TribesV2	TS-TribesV3
OKA2	-1.21e-3	-1.18e-3	-1.20e-3
Sympart	1.20e-4	1.44e-4	1.61e-4
S_ZDT1	1.50e-3	1.70e-3	5.24e-3
S_ZDT2	3.29e-4	8.65e-4	5.14e-4
S_ZDT4	6.52e-3	2.78e-2	1.52e-5
R_ZDT4	2.46e-2	8.55e-3	3.22e-4
S_ZDT6	9.59e-3	2.19e-2	2.92e-2
S_DTLZ2	1.30e-4	5.93e-4	1.94e-3
S_DTLZ3	2.98e-4	3.40e-3	1.74e-2
WFG1	1.63e-1	2.17e-1	1.70e-1
WFG8	-1.28e-1	-8.96e-2	-5.74e-2
WFG9	-7.22e-2	-2.49e-2	-1.05e-2

6 CONCLUSIONS

We have introduced a new hybrid multi-objective evolutionary algorithm based on Tribes and TS. This hybrid aims to combine the high convergence rate of Tribes with the good neighbourhood exploration performed by the TS algorithm. Therefore, we have studied the impact of the place where we apply TS technique on the performance of the algorithm. The proposed version TS-TribesV1 gave the best results almost for all the test functions except for S-ZDT4 and R-ZDT4 for which the TS-TribesV3 gave the best results.

The results showed that the hybridization is a very promising approach to multi-objective optimization. As part of our ongoing work we are going to compare the proposed algorithms with other techniques that are representative of the state of art of the multi-objective optimization. Moreover, we are going to study other hybridization between Tribes and other local search techniques.

REFERENCES

- Bartz-Beielstein, T., Limbourg, P., Parsopoulos, K.E., Vrahatis, M.N., Mehnen, J., and Shmitt, K. (2003, December). Particle Swarm Optimizers for Pareto Optimization with Enhanced Archiving Techniques. *In congress on Evolutionary Computation Canberra, Australia*, IEEE Press, Vol. 3, 1780-1787.
- Bergh, F. (2002). *An Analysis of Particle Swarm Optimizers*. PhD thesis, Departement of Computer Science, University of Pretoria, Pretoria, South Africa.
- Carlos, A. and Coello, C.A.C. (2000, June). An Updated Survey of GA-Based Multiobjective Optimization Techniques. *ACM Computing Surveys*, Vol. 32, No. 2.
- Chelouah, R. and Siarry, P. (2000). Tabu Search applied to global optimization. *European Journal of Operational Research* 123, 256-270.
- Clerc, M. (2006). *Particle Swarm Optimization*. International Scientific and Technical Encyclopaedia, John Wiley & sons.
- Coello, C.A.C and Lechuga, M.S. (2002, May). MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization. *Congress on Evolutionary Computation (CEC'2002)*, IEEE Service Center, Piscataway, New Jersey, Vol. 2, 1051-1056.
- Cooren, Y. (2008). *Perfectionnement d'un algorithme adaptatif d'optimisation par essaim particulaire. Applications en génie médicale et en électronique*. PhD thesis, Université Paris 12.
- Coello, C.A.C., Pulido, G.T. and Lechuga, M.S. (2004, June). Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3), 256-279.
- Hu, X., Eberhart, R. and Shi, Y. (2003). Particle swarm with Extended Memory for multi-objective Optimization. *In IEEE Swarm Intelligence Symposium*.
- Knowles, J., Thiele, L. and Zitzler, E. (2006, February). A tutorial on the Performance Assessment of Stochastic Multi-objective Optimizers. *Tik-Report No-214*, Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland.
- Moore, J. and Chapman, R. (1999). *Application of particle swarm to multiobjective optimization*. Departement of Computer Science and Software Engineering, Auburn University.
- Parsopoulos, K.E., Tasoulis, D.K. and Vrahatis, M.N. (2004, February). Multiobjective optimization using parallel vector evaluated particle swarm optimization. *In Proceedings of the IASTED International Conference on Artificial Intelligence and Applications (AIA 2004)*, Innsbruck, Austria, ACTA Press, Vol. 2, 823-828.
- Parsopoulos, K.E. and Vrahatis, M.N. (2002). Particle Swarm Optimization Method in Multi-objective Problems. *Proceedings of the ACM 2002 Symposium on Applied Computing (SAC'2002)*, 603-607.
- Quintero, L.V.S., Santiago, N.R. and Coello, C.A.C. (2008). Towards a More efficient Multi-objective Particle Swarm Optimizer. *Multi-objective Optimization in computational intelligence: Theory and practice*, Information Science Reference, Hershey, USA, In Lam Thu Bui and Sameer Alam (editors), 76-105.
- Ray, T. and Liew, K.M. (2002, March). A swarm metaphor for multiobjective design optimization. *Engineering Optimization*, 34(2), 142-153.
- Sierra, M.R. and Coello, C.A.C. (2005). Improving PSO-based multi-objective optimization using crowding, mutation and ϵ -dominance. *In third International Conference on Evolutionary Multi-Criterion Optimization*, Guanajuata, Mexico, LNCS 3410, Springer-verlag, 505-519.
- Sierra, M.R. and Coello, C.A.C. (2007). A study of techniques to improve the efficiency of a multi-objective particle swarm optimizer. *Evolutionary Computation in Dynamic and Uncertain Environments*, Springer, 269-296.
- Zitzler, E. and Deb, K. (2007, July). Tutorial on Evolutionary Multiobjective Optimization. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'07)*, London, United Kingdom.

ADAPTIVE FUZZY SLIDING MODE CONTROL FOR UNCERTAIN NONLINEAR SYSTEMS AGAINST ACTUATOR FAULTS

Meriem Benbrahim, Najib Essounbouli, Abdelaziz Hamzaoui
CReSTIC, Reims University, 9 rue de Quebec B.P. 396, F-10026 Troyes Cedex, France
{meriem.benbrahim, najib.essounbouli, abdelaziz.hamzaoui}@univ-reims.fr

Ammar Betta

Batna University, 5 rue Chahid boukhloouf, 05000 Batna, Algeria

Keywords: Fuzzy control, Fault tolerant control, Sliding mode control.

Abstract: In this paper, we propose to combine the fuzzy sliding mode control to tolerate actuator faults of unknown nonlinear systems subject to external disturbances. In particular, the idea of using adaptive fuzzy system to tolerate actuator faults of unknown nonlinear systems by approximating the system functions and the effects caused by actuator faults are avoided by the control structure. On the basis of Lyapunov stability theory it is shown that the resulting adaptive closed loop system can be guaranteed to be asymptotically stable in the presence of faults on actuators and disturbances.

1 INTRODUCTION

In most practical control systems, components failures may occur at uncertain time and the size of a fault is also unknown. The faults may lead to performance deterioration or even instability of the system. Therefore, the study of designing fault-tolerant control (FTC) systems, which let the systems operate in safe conditions and with proper performances whenever components are healthy or faulted, has received considerable attention over the past two decades (Veillette, 1995) (Yang et al., 2001) (Wang.R et al., 2007) (Liao.F et al., 2002) (Wu and Zhang, 2006) (Zhang et al., 2008). The existing fault-tolerant design approaches can be broadly classified into two groups, namely passive approaches (Zhao and Jiang, 1998) and active approaches (Mao and Jiang, 2007). In the passive approaches, robust control techniques are utilized to design a fixed controller for maintaining the acceptable system stability and performances throughout normal or faulty cases.

Recently, adaptive control has been widely used to deal with actuator faults in various systems. In (Tao et al., 2004) (Boskovic et al., 1998), actuator lock-in-place (stuck at some unknown place) failures were accommodated by adaptive redundant control structure for linear systems. (Tao et al., 2004) also contains corresponding studies on some systems with known nonlinearities. (Tang et al., 2007) extended the re-

sults to MIMO parametric-strict-feedback nonlinear systems. Loss of actuator effectiveness is considered in (Ye and Yang, 2006) (Yang and Ye, 2006) for linear systems in the framework of linear matrix inequality (LMI) to guarantee not only the stability, but also the robust performance of the failed system. The common advantage of these adaptive control approaches against actuator fault is that they are independent of fault detection and diagnosis (FDD).

(Boskovic et al., 2005) (Boskovic and Mehra, 2006) developed adaptive flight control based on multiple model, switching and tuning. However, the methods mentioned above require that the controlled system is known or only contains some linear unknown parameters when there is on fault.

Since it was proved that adaptive fuzzy systems are universal approximators (Wang and Mendel, 1992), and stable adaptive fuzzy control design was showed in (Wang, 1994), fuzzy logic systems (FLS) and neural network (NN) have been used to nonlinear systems, and also FTC systems. In (Polycarpou and Helmicki, 1995), a general framework for constructing automated fault diagnosis and accommodation architectures was presented using on-line approximators and adaptive schemes, (Polycarpou et al., 2004) (Zhang et al., 2004) (Zhang et al., 2006) (Mao et al., 2006) (Xue and Jiang, 2006) provided several FTC methods based on fuzzy logic systems (FLSs) or/and NNs. (Diao and Passino, 2001) and (Rong et al.,

2006) applied FTC to practical systems for a turbine engine and aircraft autoland respectively. Most of the existing works on fuzzy or neural networks FTC is to detection and diagnosis/isolation faults with FLS or NN. Thus, good fault detection and diagnosis (FDD) is very important since if there are false or omitted alarms of the faults, the overall system may even become unstable. In (Ping and Yang, 2008), the authors developed an adaptive FTC approach without resorting to FDD mechanism to accommodate both total and partial loss of effectiveness of actuators in unknown affine nonlinear systems. The main idea is to introduce adaptive fuzzy systems to tolerate actuator faults of unknown nonlinear systems by approximating the system functions and the effects caused by actuator faults are avoided by the control structure. However, using the projection algorithm need some knowledge on the system behavior which represents a restrictive assumption and increases the computation time. Furthermore, only the free external disturbance case is treated.

In this paper, we propose an adaptive fuzzy sliding mode controller to tolerate actuator faults of unknown nonlinear systems with external disturbances. In The opposite case of the approaches developed in the literature, only one fuzzy system is used to approximate the unknown dynamics, which allows avoiding perfectly the controller singularity problem. Top deal with the external disturbances and the approximation errors, sliding mode technique is adopted. Hence, the used sliding surface has been modified such that the approaching phase is removed to overcome the knowledge of the upper bound of disturbances to guarantee the sliding condition and to efficiently eliminate the chattering phenomenon.

This paper is organized as follows: Section 2 describes the problem statement. Section 3 is dedicated to the synthesis of the proposed approach. In section4, a simulation example demonstrates the effectiveness of the propose scheme. Finally, section 5 concludes the paper.

2 PROBLEM STATEMENT

Consider the following nonlinear system with m inputs:

$$\begin{cases} \dot{x}_i = x_{i+1} & 1 \leq i \leq n-1 \\ \dot{x}_n = f(\underline{x}) + \underline{g}^T(\underline{x})\underline{u} + d(t) \\ y = x_1 \end{cases} \quad (1)$$

where $\underline{x} = [x_1, x_2, \dots, x_n]^T$ represents the state vector, $\underline{u} = [u_1, \dots, u_m]^T \in \mathfrak{R}^m$ is the input vector whose the component may fail during the system operation,

$y \in \mathfrak{R}$ is the output system, $\underline{g}^T(\underline{x}) = [g_1, \dots, g_m] \in \mathfrak{R}^m$ and $f(\underline{x})$ are unknown continuous nonlinear functions. d is the bounded external disturbance. The states x_i ($i = 1, \dots, n$) are measurable and the reference output y_m is bounded and sufficiently derivable. this is a multiple input single output system with all the inputs contributed to a common control object like stabilizing the closed loop system, tracking a reference signal with satisfactory performance of both. There are many such systems in our real life. The provided approach is also effective for multi input multi output systems. We only consider a simple case to simplify the presentation. The actuator faults considered in this paper is the loss of effectiveness which is modeled as follows:

$$\begin{cases} u_i(t) = \rho_i v_i(t) & 1 \leq i \leq m \\ \rho_i \in [0, 1] \end{cases} \quad (2)$$

ρ_i is the still effective proportion of the i^{th} actuator after losing some effectiveness. When $\rho_i = 1$, the corresponding actuator is normal (without fault). With the actuator fault (2), the input vector can be rewritten as:

$$\underline{u}(t) = \underline{\rho}\underline{v}(t) \quad (3)$$

where $\underline{v}(t) = [v_1, \dots, v_m]^T$ is the applied control vector and $\underline{\rho} = \text{diag}(\rho_1, \dots, \rho_m)$.

The control objective is to design a robust adaptive fuzzy sliding mode control law for the system (1) with the actuator fault (2) to ensure that all signals are bounded in the closed loop and the output $y(t)$ can track the given reference signal $y_r(t)$ as closely as possible despite the presence of uncertainties, external disturbances and actuator faults. From the fault model (2), it is reasonable that there is at least one actuator still active for the control purpose. In this case, we propose to use a proportional actuation structure as follows (Ping and Yang, 2008):

$$\underline{v}(t) = \underline{b} v_0(t) \quad (4)$$

where $\underline{b} = [b_1, \dots, b_m]^T$ represents the matrix of proportional actuation and $v_0(t)$ the proposed robust adaptive fuzzy sliding mode control law. Using equations (3) and (4), the system (1) will be described by:

$$\begin{cases} \dot{x}_i = x_{i+1} & 1 \leq i \leq n-1 \\ \dot{x}_n = f(\underline{x}) + \underline{g}^T(\underline{x})\underline{\rho}\underline{b}v_0(t) + d(t) \\ y = x_1 \end{cases} \quad (5)$$

For this, the following assumptions are needed:

Assumption 1. System (1) is constructed such that despite the loss of actuator effectiveness according to (2), the system still be forced.

Assumption 2. The external disturbance $d(t)$ is assumed to be bounded, i.e., there exists a positive unknown constant χ such that: $|d| < \chi$.

The proposed control scheme combines fuzzy logic for approximation and sliding mode for robustness to attain the control objectives.

3 PROPOSED APPROACH

3.1 Fuzzy Logic System

An fuzzy logic system (FLS) consists of four parts: the knowledge base, the fuzzifier, the fuzzy inference engine manipulating fuzzy rules, and the defuzzifier (Wang, 1994). The knowledge base for the FLS comprises a collection of fuzzy IF-THEN rules. The fuzzifier maps a real point in the input space (measurement of the systems state) to a fuzzy set. In general there are two possible choices of this mapping, namely singleton or non-singleton. In this paper, we use the singleton fuzzifier mapping. The fuzzy inference engine performs a mapping from fuzzy sets of the input to fuzzy sets in the output space, based on the fuzzy IF-THEN rules (in the fuzzy rule base) and the compositional rule of inference. The defuzzifier maps fuzzy sets in the output space to a crisp point in this space; in this study we use the centre-average defuzzifier mapping (Wang, 1994).

The output of a multi-input single-output FLS with centre-average defuzzifier, product inference, and singleton fuzzifier are of the following form:

$$y(\underline{x}) = \frac{\sum_{i=1}^m y_i \cdot \left(\prod_{j=1}^n \mu_i(x_j) \right)}{\sum_{i=1}^m \prod_{j=1}^n \mu_i(x_j)} \quad (6)$$

where $\mu_i(x_j)$ represents the membership degree of the input x_j , y_i the conclusion constant corresponding to the i 'th rule and m the number of used fuzzy rules.

The output of the FLS can be rewritten on the following vectorial form (Wang, 1994):

$$y(\underline{x}) = \underline{\Psi}^T \underline{\Phi}(\underline{x}) \quad (7)$$

where $\underline{\Psi} = [y_1, \dots, y_m]^T$ represents the vector of the adjustable parameters and $\underline{\Phi}(\underline{x}) =$

$\left[\frac{\prod_{j=1}^n \mu_1(x_j)}{\sum_{i=1}^m \prod_{j=1}^n \mu_i(x_j)}, \dots, \frac{\prod_{j=1}^n \mu_m(x_j)}{\sum_{i=1}^m \prod_{j=1}^n \mu_i(x_j)} \right]^T$ the regressor vector.

According to the universal approximation theorem (Wang, 1994), there exists an optimal fuzzy system

in the form (5) such it approximates uniformly an unknown continuous function $h(\underline{x})$ on a compact set for any approximation accuracy:

$$h(\underline{x}) = \underline{\Psi}^{*T} \underline{\Phi}(\underline{x}) + \varepsilon \quad (8)$$

where ε is a very small positive constant.

3.2 Sliding Mode Control

To attain the desired objectives, we propose to use a sliding mode control. This choice is motivated by the fact that sliding mode allows to maintain the tracking performances in presence of both structural uncertainties and external disturbances (Slotine and Li, 1991). For this, we consider the following sliding surface:

$$S(t) = e^{(n-1)}(t) + \sum_{i=1}^{n-1} \lambda_{i-1} e^{(i-1)}(t) \quad (9)$$

where $e(t) = y_r(t) - y(t)$ denotes the tracking error and $e^{(i)}(t)$ its i 'th time derivative. The constants λ_i are chosen such the corresponding polynomial roots are stable (Slotine and Li, 1991). Using the sliding surface $S(t)$ in this actual form presents two major drawbacks: (i) during the reaching phase, the system is sensitive to uncertainties and external disturbances, which provokes chattering phenomenon in the neighborhood of the sliding surface. (ii) Choosing big values of the slops which allows reducing the reaching phase but requires an important starting energy, and small values give a slow response. So, it is necessary to find a trade-off between the starting energy and the time response (Hussain et al., 2010). To overcome this problem, we propose to use a modified the sliding surface allowing to suppress the reaching phase, and hence the system will be at $t = 0$ on the surface ($S(t) = 0$). In this case, the sliding surface will be defined as follows:

$$\begin{aligned} S(t) &= e^{(n-1)}(t) + \sum_{i=1}^{n-1} \lambda_{i-1} e^{(i-1)}(t) \\ &- \frac{2}{\pi} \left[\frac{\pi}{2} - \arctan g(t) \right] \left(e^{(n-1)} + \sum_{i=1}^{n-1} \lambda_{i-1} e^{(i-1)} \right) \quad (10) \\ &= e^{(n-1)}(t) + \sum_{i=1}^{n-1} \lambda_{i-1} e^{(i-1)}(t) + S_0 \end{aligned}$$

If $f(\underline{x})$ and $[\underline{g}^T(\underline{x})\rho\underline{b}]$ are well known, the control law can be given as:

$$\begin{aligned} v_0 &= [\underline{g}^T(\underline{x})\rho\underline{b}]^{-1} \left[-f(\underline{x}) + y_r^{(n)} + \sum_{i=1}^{n-1} \lambda_{i-1} e^{(i)}(t) \right] \\ &+ [\underline{g}^T(\underline{x})\rho\underline{b}]^{-1} [k_d \text{sign}(S(t))] \quad (11) \end{aligned}$$

Where k_d is a positive constant chosen such that: $S(t) \cdot \dot{S}(t) < 0$.

However, the dynamics of the system studied in our paper are unknown which makes the use of this control law impossible. To resolve this problem, one can use direct adaptive fuzzy controller or an indirect adaptive fuzzy controller. In the direct scheme, the control law is approximated by a fuzzy system. However, the control gain must be constant or satisfying some restrictive assumptions. The indirect scheme consists in approximating the unknown dynamics by two fuzzy systems to synthesize the control law. Nevertheless, the used adaptation laws are very complicated to avoid the singularity problem. In this work, we propose to approximate the unknown terms using only one fuzzy system under the constraint that the robustness of the closed loop system is guaranteed and the number of the involved parameters in the control design is reduced.

3.3 Control Law Synthesis

This section is dedicated to the synthesis of the proposed approach.

Using (5), the time derivative of the sliding surface (10) is given by:

$$\begin{aligned} S\dot{(t)} &= e^{(n)} + \sum_{i=1}^{n-1} \lambda_{i-1} e^{(i)}(t) \\ S\dot{(t)} &= y_r^{(n)} - f(\underline{x}) - [\underline{g}^T(\underline{x})\rho\underline{b}]v_0 - d + \sum_{i=1}^{n-1} \lambda_{i-1} e^{(i)}(t) \end{aligned} \quad (12)$$

If we muster all the unknown parameters in one, the above expression can be rewritten as:

$$S\dot{(t)} = y_r^{(n)} - f_d(\underline{x}) - [\underline{g}^T(\underline{x})\rho\underline{b}]v_0 \quad (13)$$

$$\text{where } f_d(\underline{x}) = f(\underline{x}) + d - \sum_{i=1}^{n-1} \lambda_{i-1} e^{(i)}(t).$$

According to assumption 1, we have $[\underline{g}^T(\underline{x})\rho\underline{b}] \neq 0$. So, it can be positive or negative. We assume in this work, that there exists a positive constant g_0 such that: $[\underline{g}^T(\underline{x})\rho\underline{b}] > g_0 > 0$. Furthermore, the function $f_d(\underline{x})$ is unknown. To attain the control objectives, we propose to use a fuzzy system $\underline{\Psi}^T \underline{\Phi}(\underline{x})$ to approximate. We define a new variable α such that: $\alpha = g_0^{-1} \|\underline{\Psi}\|^2$. According to (8), and the approximation error as: $\tilde{\alpha} = \alpha - \hat{\alpha}$ whose time derivative is given by: $\dot{\tilde{\alpha}} = -\dot{\hat{\alpha}}$

Proposition *The control law*

$$v_0 = M_0 \cdot S(t) + \frac{\hat{\alpha}}{2\beta^2} \underline{\Phi}(\underline{x})^T \underline{\Phi}(\underline{x}) S(t) \quad (14)$$

with

$$\dot{\hat{\alpha}} = \frac{\gamma}{2 \cdot \beta^2} \underline{\Phi}(\underline{x})^T \underline{\Phi}(\underline{x}) S(t) \quad (15)$$

guarantees the stability and the robustness of the closed loop system in presence of actuators faults.

It ensures also the boundedness of all the involved signals.

Proof.

According to (8), (13) and (14), using the fact that the reference signal $y_r^{(n)}$ yields to:

$$\begin{aligned} S(t)S\dot{(t)} &\leq \frac{g_0\alpha}{2\beta^2} \underline{\Phi}(\underline{x})^T \underline{\Phi}(\underline{x}) S^2(t) + \frac{\beta^2}{2} + \frac{g_0 \cdot S^2(t)}{\eta^2} \\ &\quad + \frac{\eta^2(\varepsilon^2 + \chi^2)}{2g_0} - [\underline{g}^T(\underline{x})\rho\underline{b}]v_0 \end{aligned} \quad (16)$$

Where η and χ two positive constants.

To prove the stability, we consider the following Lyapunov function:

$$V_L = \frac{1}{2} S^2(t) + \frac{g_0}{2\beta} \tilde{\alpha}^2 \quad (17)$$

Using equations (15) and (16), the time derivative of (17) becomes:

$$\begin{aligned} \dot{V}_L &\leq \frac{1}{2} \left[\beta^2 + \frac{\eta^2(\varepsilon^2 + \chi^2)}{g_0} - g_0\sigma\alpha^2 \right] \\ &\quad + \left[-2g_0M_0 \frac{S^2(t)}{2} - g_0\eta\sigma \frac{\tilde{\alpha}^2}{2} \right] \end{aligned} \quad (18)$$

Let $a_0 = \frac{1}{2} \left[\beta^2 + \frac{\eta^2(\varepsilon^2 + \chi^2)}{g_0} - g_0\sigma\alpha^2 \right]$ and $b_0 = \min(2g_0M_0, \eta\sigma)$. Then, the time derivative of V_L is given by:

$$\dot{V}_L \leq a_0 + b_0 V_L(0) \quad (19)$$

which implies

$$V_L(t) \leq V_L(0) \exp(a_0 t) + \frac{b_0}{a_0} \quad \forall t \geq 0 \quad (20)$$

Hence, the Lyapunov function converges toward a bounded value $\frac{b_0}{a_0}$. This implies that all the involved signals are bounded. Furthermore, we can have $\lim_{t \rightarrow \infty} S(t) \leq 2\frac{b_0}{a_0}$, which ensures the convergence of the tracking error to zero (Wang, 1994).

4 SIMULATION AND RESULTS

In this section, the presented adaptive fuzzy fault tolerant controller is applied to a nonlinear system with the actuator faults described as (2).

Example: We consider that after transformation, the nonlinear system can be written as the following form which has a redundancy actuation structure.

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{5\sin x_1 - 0.02x_2^2 \cos(x_1) \sin(x_1)}{3 - 0.2\cos^2 x_1} + \frac{\cos^2 x_1}{3 - 0.2\cos^2 x_1} u_1 \\ &\quad + \frac{2\cos^2 x_1}{3 - 0.2\cos^2 x_1} u_2 + d \end{aligned} \quad (21)$$

Where the actuators of u_1 and u_2 are the control inputs, $d = 0.1\sin(2t)$ represents the external disturbance. The evolution of the actuators effectiveness $\rho = \text{diag}(\rho_1, \rho_2)$ is given by figure (1). In order to control system (21), the proposed control law is applied with the following simulation parameters: $m = 5 \times 5 = 25$ rules for the fuzzy logic system, with $\rho = 0.01$, $\gamma = 4.250$ and initial values $\alpha = 0_{25}$. Figures 2 and 3 give the simulation results for regulation problem and 4-5 those of tracking of a sinusoidal reference signal. We can see the convergence of the states to their respective reference signals despite the presence of both effectiveness loss (figure 1) and external disturbances.

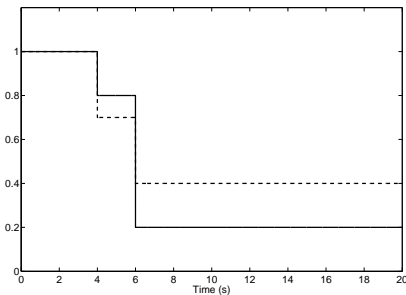


Figure 1: Evolution of the effectiveness: (—): ρ_1 , (- -): ρ_2 .

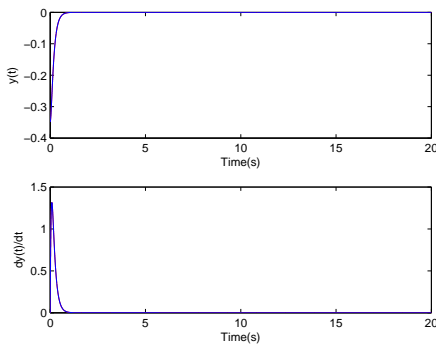


Figure 2: Evolution of the state variables.

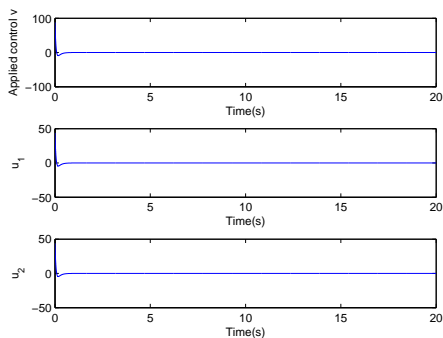


Figure 3: Control signals.

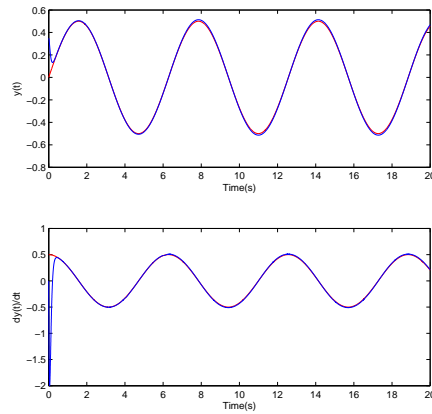


Figure 4: Evolution of the state variables and their reference signals.

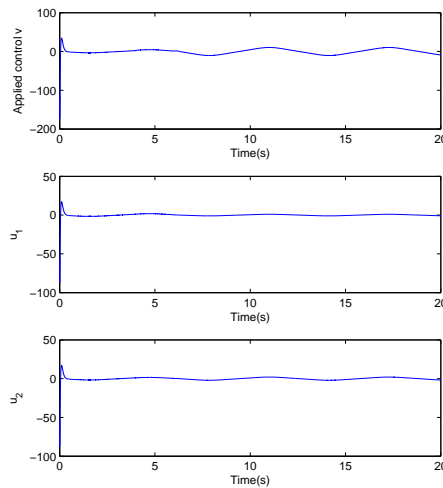


Figure 5: Control signals: v , u_1 and u_2 .

5 CONCLUSIONS

In this paper, a fuzzy sliding mode approach of fault tolerant control problem for an uncertain perturbed nonlinear system is studied. To overcome the problem of unknown dynamics, only one adaptive fuzzy system has been used. Furthermore, the sliding surface has been modified to suppress the reaching phase and hence improve the robustness of the closed loop system. The global stability has been established in the sense of Lyapunov. Many simulations have presented to show the good performances despite the presence of actuator failures. As future work, the case of actuator lock-in-place will be also treated and the extension of this approach to multi-input multi-output will be studied.

ACKNOWLEDGEMENTS

This work is developed for the project CPER-MOSYP and supported by both the Champagne-Adrenne Region and the European found FEDER.

REFERENCES

- Boskovic, J., Bergstrom, S., and Mehra, R. (2005). Adaptive accommodation of failures in second-order flight control actuators with measurable rates. In *American Control Conference*.
- Boskovic, J. and Mehra, R. (2006). A multiple model-based decentralized system for accommodation of failures in second-order flight control actuators. In *American Control Conference*.
- Boskovic, J., Yu, S., and Mehra, R. (1998). A stable scheme for automatic control reconfiguration in the presence of actuator failures. In *American Control Conference*. New York:IEEE Press.
- Diao, Y. and Passino, K. (2001). Stable fault-tolerant adaptive fuzzy/neural control for a turbine engine. *IEEE Transactions on Control Systems Technology*.
- Hussain, A., Essounbouli, N., and Hamzaoui, A. (2010). A robust adaptive fuzzy wavelet network based controller for a class of nonlinear systems. *American Control Conference Westin Seattle Hotel, Seattle, Washington, USA*.
- Liao, F., Wang, J., and Yang, G. (2002). Reliable robust flight tracking control: an lmi approach. *IEEE Transactions on Control Systems Technology*.
- Mao, Z. and Jiang, B. (2007). Fault estimation and accommodation for networked control systems with transfer delay. *Acta Automatica Sinica*.
- Mao, Z., Jiang, B., and Chowdhury, F. (2006). Fault accommodation for a class of nonlinear flight control systems. *International Symposium on Systems and Control in Aerospace and Astronautics*.
- Ping, L. and Yang, G.-H. (2008). Fault tolerant control for unknown nonlinear systems with actuator failures: An adaptive fuzzy approach. *American Control Conference Westin Seattle Hotel, Seattle, Washington, USA*.
- Polycarpou, M. and Helmicki, A. (1995). Automated fault detection and accommodation: a learning systems approach. *IEEE Transactions on Systems, Man, and Cybernetics*.
- Polycarpou, M., Zhang, X., Xu, R., Yang, Y., and Kwan, C. (2004). A neural network based approach to adaptive fault tolerant flight control. In *TEMPLATE'06, 1st International Conference on Template Production*.
- Rong, H. J., Huang, G. B., Sundararajan, N., and Saratchandran, P. (2006). Fuzzy fault tolerant controller for actuator failures during aircraft autoland. *IEEE International Conference on fuzzy systems*.
- Slotine, J. and Li, W. (1991). *Applied Nonlinear Control*. Englewood Cliffs Prentice Hall, New Jersey.
- Tang, X., Tao, G., and Joshi, S. (2007). Adaptive actuator failure compensation for nonlinear mimo systems with an aircraft control application. *Automatica*.
- Tao, G., Chen, S., Tang, X., and al (2004). *Adaptive Control of Systems with Actuator Failures*. Springer-Verlag, Berlin.
- Veillette, R. (1995). Reliable linear-quadratic state-feedback control. *Automatica*.
- Wang, L. (1994). *Adaptive fuzzy systems and control*. Englewood Cliffs NJ: Prentice Hall, New Jersey.
- Wang, L. and Mendel, J. (1992). Fuzzy basis functions, universal approximation, and orthogonal least-squares learning. *IEEE Transactions on Neural Networks*.
- Wang, R., Jin, G., and Zhao, J. (2007). Robust fault-tolerant control for a class of switched nonlinear systems in lower triangular form. *Asian Journal of Control*.
- Wu, H. and Zhang, H. (2006). Reliable hinf fuzzy control for continuous-time nonlinear systems with actuator failures. *IEEE Transactions on Fuzzy Systems*.
- Xue, H. and Jiang, J. (2006). Fault detection and accommodation for nonlinear systems using fuzzy neural networks. *IEEE 5th International Power Electronics and Motion Control Conference*.
- Yang, G., Wang, J., and Soh, Y. (2001). Reliable hinf controller design for linear systems. *Automatica*.
- Yang, G. and Ye, D. (2006). Adaptive fault-tolerant hinf control via state feedback for linear systems against actuator faults. In *45th IEEE Conference on Decision and Control*.
- Ye, D. and Yang, G. (2006). Adaptive fault-tolerant tracking control against actuator faults with application to flight control. *IEEE Transactions on Control Systems Technology*.
- Zhang, D., Wang, Z., and Hu, S. (2008). Robust satisfactory fault tolerant control of uncertain linear discrete-time systems: an lmi approach. *International Journal of Systems Science*.
- Zhang, X., Liu, Y., Rysdyk, R., Kwan, C., and Xu, R. (2006). An intelligent hierarchical approach to actuator fault diagnosis and accommodation. *IEEE Aerospace Conference*.
- Zhang, X. D., Parisini, T., and Polycarpou, M. M. (2004). Adaptive fault-tolerant control of nonlinear uncertain systems: an information-based diagnostic approach. *IEEE Transactions on Automatic Control*.
- Zhao, Q. and Jiang, J. (1998). Reliable state feedback control system design against actuator failures. *Automatica*.

A CONTEXT-AWARE ADAPTATION SYSTEM FOR SPATIAL AUGMENTED REALITY PROJECTIONS

Anne Wegerich, Julian Adenauer

Research Training Group prometei, Technische Universität Berlin, Germany
{anne.wegerich, julian.adenauer}@zmms.tu-berlin.de

Jeronimo Dzaack, Matthias Roetting

Department of Human-Machine-Systems, Technische Universität Berlin, Germany
{jdz, roetting}@mms.tu-berlin.de

Keywords: Spatial Augmented Reality, Projection systems, Context-aware user adaptation, User interfaces.

Abstract: To cover three-dimensional information spaces stationary or spatial Augmented Reality (sAR) systems involve installed projection systems, Head-Up- and other displays. Therefore, information presentation techniques for sAR contain three basic problems assigned to the questions which form, which screen position and which physical location the information should have in 3D space. This paper introduces an approach and presents the details of a corresponding system that concentrates on the location problem and the appropriate visualization adaptation. It manages the information presentation for physical occlusions and difficult light conditions of sAR floor projections with a light sensor matrix and a connected software for low and high-level context integration. With changing the size, position, and orientation of the projection area and the content of the presented information it implements a context-aware adaptation system for sAR.

1 INTRODUCTION

Augmented Reality (AR), the augmentation of the environment with virtual, computer generated information, is used in many different areas. If the user is required to move in large areas, he or she needs to carry a mobile presentation device or wear a head-mounted display (HMD). But whenever the user's movements are restricted to a small, controlled area, this burden can be loosened by realizing spatial or stationary AR (sAR).

Whereas in mobile AR the information is typically presented on only one display device, sAR systems use many displays distributed in a small 3D space. This bears the danger that the user is confronted with incoherent and/or redundant information because many displays or segregated (parts of) displays are used for showing one information. To avoid this, a central system is required to automate the selection and optimization of the information that is presented to the user.

Beside the display problems where on the screen and in which form the sAR information should be presented (view and presentation management) the main question especially for sAR is: Where (i.e. on which

display or projection area) should the information be presented physically in a room (display management)? Furthermore, the definition of AR by (Azuma et al., 2001) requires that the virtual information has to be combined with the real world in real time and it has to have a content-related connection to the point where it is presented (spatial registration).

Altogether information presentation with (s)AR display technologies needs to attend to physical problems like edges, gaps, over-lappings of two or more projections, and occlusions without losing spatial registration as a content-related requirement (Fig. 1).

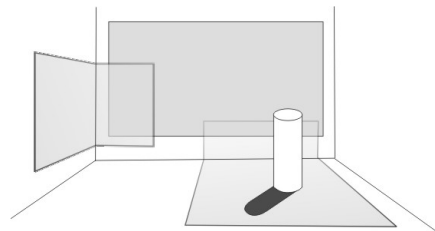


Figure 1: Gaps and overlappings of three projections (gray) in a room with particular occlusion (shadow and white column).

In this paper we address occlusions and we present

a sAR system that handles the presence of occluding objects and related light conditions (shadows or reflections) with regard to the AR definition. More precisely the system manages the context-aware recalculation of location, size, orientation, and visualization of a projected information if a physical object (e.g. the user himself) occludes it.

After showing some related work concerning the display management and the measurement of environmental context information we present the proposed system in section 3 and 4. Afterwards we explain the purpose of the software and how it will be integrated in the further development of a sAR smart home system.

2 DISPLAY MANAGEMENT AND MEASUREMENT OF ENVIRONMENTAL CONTEXT

Several solutions exist for recognizing over-lappings and closing gaps in projected sAR scenarios. Therefore, a movable mirror combined with a projector is presented by (Pinhanez, 2001). So the presentation follows the user. Other rotatable projectors are suggested by (Ehnes et al., 2004; Ehnes and Hirose, 2006). These projectors offer only one screen that moves and does not produce any gap or overlapping. To solve the problem of roaming between different immovable projection systems these authors also designed an architecture that handles overlappings with selecting the best possible projection system (Ehnes et al., 2005). However, they did not make a suggestion for physical occlusions.

For the integration of high-level context a lot of AR applications are based on image processing systems which are able to recognize states of the environment or the user. This is usually used to provide automated user support in communication, work, or information processing. In AR applications this recognition is very important but often limited to special areas like city navigation or guidance in museums. However, an approach for a user-adaptation with high-level context integration is missing. Especially for 3D information spaces new research results give evidence that information visualization must be adapted for different 3D presentation depths and optimal information perception of proposed virtual distances and perspectives (Drascic and Milgram, 1996; Jurgens et al., 2006; Herbon and Roetting, 2007).

Separated from high-level context, environmental (or low-level) context registration is a field of research that is well investigated. Therefore, the cur-

rent research is focused on new applications and combinations of them to facilitate the development of new technologies especially in the scope of Human-Machine-Interaction (HCI). Hence, there is a huge amount of applications in which environmental context data is collected with sensors. Especially for popular social applications a lot of psychophysiological data is measured to enable the automated recognition of emotional states of the user. Such systems concentrate for example on areas like e-learning platforms (Karamouzis and Vrettos, 2007), indirect or direct interaction in multimedia applications like web pages, virtual communities, and games (Ward and Marsden, 2003; Kim et al., 2008; Mahmud et al., 2007). Particularly in the scope of mixed reality games the use of low-level-context is a popular approach because it connects the real and the virtual world for the player (Romero et al., 2004).

For these and other upcoming technologies analyzing context information and integrating it to develop adaptable systems has led to context ontology models which are generic or domain specific and allow the standardized use of context information and the development of associated system or software architectures (Chaari et al., 2007).

With the presented approach we combine these context models with user-centred adaptation techniques for sAR information visualization. With this we want to optimize sAR systems, make them more useful, and establish generic models for context integration.

3 SAR SYSTEM SETUP

The system is part of the development of a sAR smart home environment which involves a number of different sAR devices. These are projection systems, video-see-through and head-up displays (Bimber and Raskar, 2005).

The proposed system is mainly based on an array of light sensors on the floor to control the position, size, and orientation of a projection. The sensors measure light conditions in the environment (low-level context information). A connected software module manages the context-related adaptation of the information visualization based on the feedback of the recalculated presentation and sends the resulting image to a projector. So the proposed low-level change of the projection will be recalculated again if the new projection got ambiguous in the current context. The support for a user searching an object for example could be showing a map with a target marker. If the projection area (location) for map presentation has to

be changed because of difficult light conditions also the size and orientation or the whole visualization type of the new map have to be adapted. The first change could have made it incomprehensible for the user e.g. because of his current perspective.

The complexity of contentual changes increases if high-level context information is taken into account. If a projection area has to be changed, the content adaptation software perhaps has to present the same information with another visualization (content) to facilitate its perception. This assumption is based on a lot of requirements determined by the abilities of the user (background knowledge, cognitive capacity, experience, etc.) and the context. In the map example this perhaps means to change the visualization from a map to an arrow because the user currently is distracted and only can process simple visualizations in his peripheral vision. So the presented system relates to a complex network of decisions. Altogether the software has to automate having the right information presentation technique at the right position for the conditions the context provides.

The system we introduce in this paper is the first step towards a full home automation and support with sAR displays. The hardware of the presented system is a proof of concept and therefore is limited to a solution that includes one projector and a small projection area. But the framework is extendable to larger projection areas and sensor arrays.

4 CONTEXT-AWARE PRESENTATION SYSTEM

The task of the system we developed is to analyze lighting conditions on the floor of a room to make this surface usable for an sAR output of a projector that is installed above it. Thus, it solves a part of the problem to add information everywhere in a three-dimensional space without losing the relevant content of it and important parts of its formal representation. Furthermore, the system provides the possibility to adapt the visualization in terms of HCI criteria and the upcoming research of three-dimensional perception.

The system consists of a light sensor matrix that is integrated in a PVC floor coating, a connected microcontroller board which is connected to a PC, and a Java-based software. The size of the PVC floor coating and the sensor array is a proof of concept and could be extended for larger rooms and projections. The presented system is able to change the size and position of an at least possible projection area. This is the first step to adapt the presented content which

depends on the resulting distance of the information to the user and additionally on the properties of other sAR devices which are in a similar distance, the type of device, its orientation, and of course the type of information which has to be presented and a lot of other context requirements.

After collecting information about the physical context (occlusions or lighting conditions) the software first evaluates possible projection areas and selects one. Secondly it has to access the properties of the selected area and analyzes further high-level context information to change the visualization of the content if needed.

4.1 Installation

The system consists of a sensor array for brightness measuring in a certain physical space and a microcontroller. It converts the analogue sensor data and sends it to a connected PC which controls the output of a projector (see Figure 2).

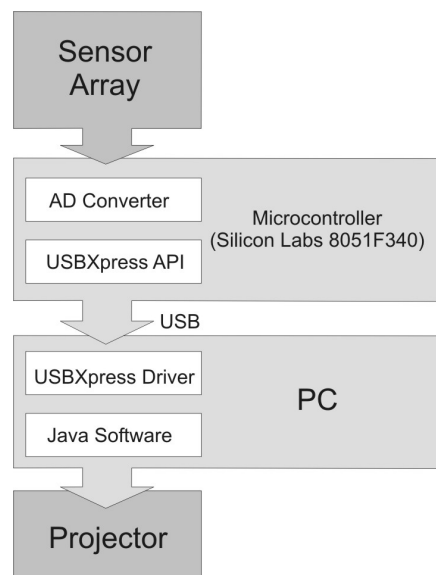


Figure 2: Schema of the context-aware presentation system.

A $0.75m \times 1m$ PVC floor coating is divided into twelve squares and a projector is placed $2.5m$ above it. The projector is connected to a PC and projects corresponding squares on the floor. In the middle of each square, a light sensor of Type AMS104Y from Panasonic's NaPiCa-series is set. The sensor's feature is a linear output and a built-in optical filter for spectral response similar to that of the human eye.

Each sensor is connected to an input of a microcontroller. The voltage over the resistor depends on

the sensor’s photocurrent and is therefore directly dependent on the amount of light on the sensor. This voltage is being measured by a 10 bit analogue-to-digital converter (ADC) which is part of the microcontroller (see Figure 3 for more detail on the sensor matrix).

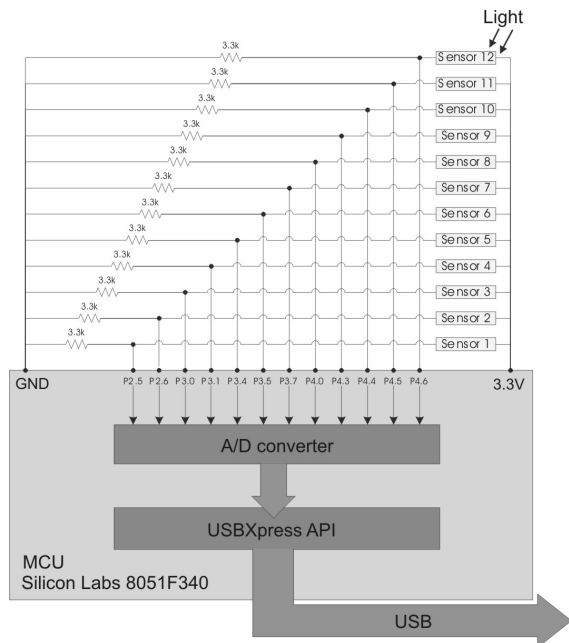


Figure 3: Diagram of the components and connections of the sensor matrix.

The microcontroller we used is an 8051F340 from Silicon Labs. It features a built-in analogue multiplexer for the ADC, enabling the controller to convert analogue voltages from 20 input pins. Moreover, it has an onboard USB controller which can be used with Silicon Labs’ USBXpress API for easy USB implementation on client and host side. Via this USB interface the controller board sends the digital sensor values to a PC.

The PC is running a software that reads the converted sensor data from the USB port. These values are compared to predefined minimum and maximum values. Value below minimum means, the sensor is covered, so nothing should be projected here. A value higher than the maximum, on the other hand means, there is too much light on the area and the decreased contrast averts a projection. Therefore, only squares (one square per sensor) with light values in the range between minimum and maximum are taken into consideration, where information should be displayed.

From these squares that are in range the program selects the ones that build up the largest coherent quadrangular area according to an implemented hier-

archy. In the resulting connection of these squares information could be displayed.

4.2 Functionality

The overall goal of the software component is the adaptation of the projected superimposition with regard to low-level and high-level data integration. Therefore, it incorporates two steps of projection recalculation: low-level and high-level data related adaptation (currently only the first software part is solved). The data is processed in real time. So the projected information is always in a visible position accepting a very small delay from the sensor data request.

In a first step the software calculates the most appropriate size, position, and orientation for a projection that can be used to present text, icons, or (rendered) images. It is made up of directly connected projection squares whose sensors are not in a shadow or in a direct light reflection. Then the new proposed projection is adapted to the unchanged orientation of the target (a text reading person or target position of a pointing projection, etc.). After this step the information has the correct orientation and is displayed in the best possible projection area for the target (e.g. the user). In Figure 4 this first adaptation step of the system is demonstrated with an arrow pointing at a designed target (red). This arrow is only one possible usage of a projection area.

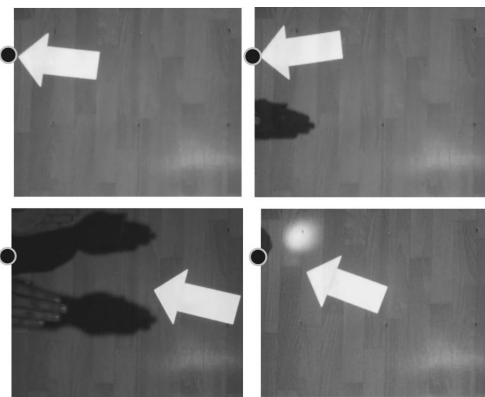


Figure 4: Examples for resulting projections (low-level recalculation) when sensors are covered or bright light falls on them, arrow always points in a target direction (represented by black dot).

The second part of the software uses the feedback signals about the proposed projection area and the current information visualization. It analyzes high-level data (context information about the user, his distance to the projection, current state of the needed in-

formation, etc.) to recalculate whether the visualization is still optimal for the user or not. If it is not, the visualization changes to an alternative image or text for a better understanding. Figure 5 shows an example for the proposed reaction of the system. The resulting visualization for a navigational hint in this case is a map instead of an arrow. We based this approach on findings from an indoor navigation experiment we made (Wegerich et al., 2009). In this experiment projected maps were rated higher and caused better performance when the target position is not visible for the user.

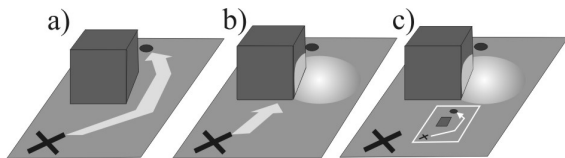


Figure 5: Example for an adaptation scenario and 2 steps of visualization recalculation a) shows the target and the starting point of the user, b) first step of adaptation where the projection area, size and orientation changes to the at least possible squares with no light reflection and the correct direction (low-level data integration), c) second step of adaptation; the visualization is changed to a map which shows the unambiguous target position.

5 DISCUSSION

The proposed system component for the adaptation of floor projections is a proof of concept. We presented a solution for the problem of occlusions in physical issues of information presentation with sAR. The sensor matrix is not limited to the presented size and amount of sensors. The usage of a higher resolution could be achieved by integrating more light sensors. Furthermore, this makes it possible to change the projection area not only to other rectangular forms but to a more adaptive shape of the presented information which we work on in the smart home scenario.

A higher resolution is also achieved with the usage of cameras which are very common in AR applications. In most cases they are integrated in the ceiling or higher edges. So image processing software is needed to solve perspectives and occlusions only the camera sees, but not the user. When a user lifts an arm and still can see the same area on the floor a camera system would change the projection area because it has difficulties to decide where the occlusion is in its distance to the floor. So the advantage of floor-based sensor matrices is that they measure the conditions at the point where the projection will be with less effort and as fast as a camera based system.

The presented sAR system component handles the formal information representation and the display management but at this point not the optimization of the content because of its complexity. Other high-level context information is needed to decide which form the information should have. This will be the result of the integration in the intended smart home system. One of the applications of high-level context integration is e.g. to pay attention to possible fast moves of a user or to distances the area adaptation causes. An arrow could be usable in this case but perhaps a map of the room (and its cabinets) is better if more detailed location information is needed.

6 FUTURE WORK

With the introduced system the foundation is made for using context information and properties of three-dimensional perception. It adapts the information visualization and furthermore solves parts of the display management problem of sAR systems. The next step is the development and evaluation of a more complex context model that follows specified guidelines for 3D information presentation and integrates different sAR devices. The aim is the enhancement of the software functionality on the basis of this model.

Afterwards, it will be combined with other context-aware system components which together form a user-adapted sAR presentation system. The resulting system will present information in larger 3D spaces where the form, position, additional interaction parameters, and especially the content is selected for 3D perception adaptation to the abilities of the user.

The presented system also solves a technical part of the Ubiquitous Spatial Augmented Reality (UAR) requirement to make information available everywhere in a larger 3D space with using context. This is an essential aspect of the underlying definition because of the so achieved connection between the concepts of Ubiquitous Computing and Augmented Reality. In this manner the presentation technology in a 3D information space needs to be context-aware in a more continuous way to adapt the presentation for any location in the room. Furthermore the underlying context model has to integrate location-based adaptation because of a possible dynamically changing environment. For example a projective sAR system used for superimpositions on a working surface should handle changing objects or tools and its positions on the surface.

Finally, we want to develop an ontology that describes and generalizes the automated decision of the

adaptation system. Based on this ontology we will develop an expert system to manage the rules of perception and cognitive processing of 3D information presentation in UAR environments and to solve the high-level context integration. This will make the system scalable and adaptable to different use cases and also for mobile AR display applications.

ACKNOWLEDGEMENTS

We want to thank the DFG, the Department of Human-Machine Systems Berlin, and prometei for their support.

REFERENCES

- Azuma, R., Bailiot, Y., Behringer, R., Feiner, S., Julier, S., and MacIntyre, B. (2001). Recent advances in augmented reality. *IEEE Comput. Graph. Appl.*, 21(6):34–47.
- Bimber, O. and Raskar, R. (2005). *Spatial Augmented Reality: Merging Real and Virtual Worlds*. A K Peters, Ltd.
- Chaari, T., Ejigu, D., Laforest, F., and Scuturici, V.-M. (2007). A comprehensive approach to model and use context for adapting applications in pervasive environments. *J. Syst. Softw.*, 80(12):1973–1992.
- Drascic, D. and Milgram, P. (1996). Perceptual issues in augmented reality. In *Stereoscopic Displays and Virtual Reality Systems III, Proceedings of SPIE*, number 2653, pages 123–134.
- Ehnes, J. and Hirose, M. (2006). Projected reality - content delivery right onto objects of daily life. In *ICAT*, pages 262–271.
- Ehnes, J., Hirota, K., and Hirose, M. (2004). Projected augmentation - augmented reality using rotatable video projectors. In *ISMAR '04: Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 26–35, Washington, DC, USA.
- Ehnes, J., Hirota, K., and Hirose, M. (2005). Projected augmentation ii — a scalable architecture for multi projector based ar-systems based on 'projected applications'. In *ISMAR '05: Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 190–191, Washington, DC, USA.
- Herbon, A. and Roetting, M. (2007). Detection and processing of visual information in three-dimensional space. *MMI Interactive Journal*, 12:18–26.
- Jurgens, V., Cockburn, A., and Billingham, M. (2006). Depth cues for augmented reality stakeout. In *CHINZ '06: Proceedings of the 7th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction*, pages 117–124, New York, NY, USA. ACM.
- Karamouzis, S. T. and Vrettos, A. (2007). A biocybernetic approach for intelligent tutoring systems. In *AIAP'07: Proceedings of the 25th conference on Proceedings of the 25th IASTED International Multi-Conference*, pages 267–271, Anaheim, CA, USA. ACTA Press.
- Kim, Y. Y., Kim, E. N., Park, M. J., Park, K. S., Ko, H. D., and Kim, H. T. (2008). The application of biosignal feedback for reducing cybersickness from exposure to a virtual environment. *Presence: Teleoper. Virtual Environ.*, 17(1):1–16.
- Mahmud, A. A., Mubin, O., Octavia, J. R., Shahid, S., Yeo, L., Markopoulos, P., and Martens, J.-B. (2007). amazed: designing an affective social game for children. In *IDC '07: Proceedings of the 6th international conference on Interaction design and children*, pages 53–56, New York, NY, USA. ACM.
- Pinhanez, C. S. (2001). The everywhere displays projector: A device to create ubiquitous graphical interfaces. In *UbiComp '01: Proceedings of the 3rd international conference on Ubiquitous Computing*, pages 315–331, London, UK. Springer-Verlag.
- Romero, L., SANTIAGO, J., and CORREIA, N. (2004). Contextual information access and storytelling in mixed reality using hypermedia. *Comput. Entertain.*, 2(3):12–12.
- Ward, R. D. and Marsden, P. H. (2003). Physiological responses to different web page designs. *Int. J. Hum.-Comput. Stud.*, 59(1-2):199–212.
- Wegerich, A., Dzaack, J., and Roetting, M. (2009). Optimizing virtual superimpositions: User centered design for a UAR supported smart home system. submitted.

EFFICIENT ORGANIZATION OF TRUCK PLATOONS BY MEANS OF DATA MINING

Application of the Data Mining Technique for the Planning and Organization of Electronically Coupled Trucks

Ralph Kunze, Richard Ramakers, Klaus Henning and Sabina Jeschke

*Center of Learning and Knowledge Management and Institute of Information Management in Mechanical Engineering
RWTH Aachen University, Dennywartstrasse 27, Aachen, Germany
{kunze, ramakers, henning, sabina.jeschke}@zlw-ima.rwth-aachen.de*

Keywords: Truck platoons, Platoon driving, Electronic coupling, Road capacity, Data mining, Sequential pattern mining.

Abstract: One opportunity to manage the increase of freight transportation and to optimize utilization of motorway capacities is the concept of truck platoons. With the aid of Advanced Driver Assistance Systems, trucks are electronically coupled keeping very short gaps (approx. 10 meters) to form truck platoons on motorways. This contributes to a couple of advantages, such as a reduction of the inter-vehicle distance, which leads to an improved vehicle occupancy and gained road space, an optimization of traffic flow, the reduction of fuel consumption advantaged by slipstream driving, a relief for professional drivers and finally an increase in safety due to the longitudinal and lateral guidance. In this paper, a brief introduction into these truck platoons is given, which have been realized within the project KONVOI. The paper focuses on the calculations for an efficient planning and organizing of truck platoons, which could have been realized by means of the application of data-mining technique. Therefore, the efficiency criterion of electronically coupled truck platoons is discussed as well as the framework of the data-mining technique and the algorithm for the calculation of efficient truck platoons are presented. Finally, the experimental results of the application of the data-mining technique and their relevance for an efficient operation of truck platoons is presented.

1 INTRODUCTION

1.1 Initial Situation

The integration of the new European member countries is a challenging component for national traffic planning in the near future. Especially countries with a central geographic position within Europe, such as Germany, have to shoulder the majority of the future traffic emergence. Additionally, this traffic will predominantly encumber the road. Due to the increase of global freight transportation, the maximum road capacity in several countries worldwide is nearly reached (Economic and Social Commission for Asia and the Pacific, 2008). A modern national economy needs an efficient traffic system to face such a challenge successfully. Otherwise, today's even worse traffic situation will be pre-assigned to collapse.

The importance of this fact was emphasized by studies from the European Commission in 2006 (Commission of the European Communities, 2006). Between 1995 and 2004 a growth of 35% in road

freight transport was detected. Furthermore, between the years 2000 and 2020 an increase of 55% in road transportation is expected (Commission of the European Communities, 2006). In the year 2003, the European Commission stated that every day 7,500 kilometers of the European road system are being blocked by traffic jams (Commission of the European Communities, 2003). Environmental pollution, safety risks and a loss in efficiency for the economy are only some of the effects that result from these factors. Similar problems are known and discussed worldwide.

One possibility to face the rising traffic volume on the roads is the modal shift to other types of transportation (e.g. rail, shipping). Further potential has the optimization of the road-side traffic flow by driving assistance systems. Since the 90s, Advanced Driver Assistance Systems (ADAS) for trucks have been on offer, including pre-adjustment of speed and distance to the front vehicle. This is exerted automatically via computerized engine- and brake-management in connection with an automated transmission. The combination of an Adaptive Cruise

Control (ACC) together with an Automatic Guidance (AG) leads to autonomous driving. The difference between platooning and autonomous driving makes the necessity of a leading vehicle. Following trucks can go far distances without any manual engagement by the driver as long as another ahead-driving vehicle exists. Nevertheless, each truck must be assigned with a truck driver at all times due to legal rules and regulations. Due to the platoons, smaller distances between the vehicles (up to 10 meters) can be realized. These truck platoons contribute to an optimization of traffic flow up to 9% and a reduction of fuel consumption due to slipstream driving (Savelsberg, 2005).

The development and evaluation of the practical use of truck platoons is the objective of the project KONVOI, which was funded by German's Federal Ministry of Economics and Technology. The Project KONVOI is an interdisciplinary research project with partners of RWTH Aachen University, industry and public institutions, which ended after a duration of 49 months with test runs on German highways at the end of May 2009 (Figure 1). With the assistance of virtual and practical driving tests by using experimental vehicles and a truck driving simulator, the consequences and effects on the human-, the organization- and the technology-dimension have been analyzed (Henning et al., 2007).



Figure 1: Test Run on German Motorways (March 2009).

1.2 Problem Definition

For planning and organization of such truck platoons, as well as evaluation and simulation purposes, it is necessary to implement an algorithm which searches for economic truck platoons within the planned or current driven routes of all participating trucks. In any case of organized platoon driving, it is essential to search for and group possible participants. In principle, the "common route" is one general search criterion, whereas other criteria (e.g. profit, waiting-time,

properties of the participating vehicles) can differ because of the focus.

In this paper, a data-mining technique is presented to solve the mentioned problem of searching for possible platoons which satisfy given specific criteria (in the case of truck platoons: economy). The scenario of driver organized truck platoons, on which KONVOI is based, is presented in section 2. Section 3 explains the operational and organizational structure of platoon systems. Before introducing the so called TPSpan-Algorithm and related work as well as the possibility to use this algorithm as a solution for the problem of planning and organizing truck platoons in section 5, the criterion of a efficient usage of electronically coupled trucks as well as the calculation basis for the efficiency of truck platoons are defined in section 4. Finally, in section 6 the experimental results of the application of the data-mining technique is presented.

2 THE SCENARIO "DRIVER ORGANIZED TRUCK PLATOONS"

The project KONVOI is based on the scenario "Driver Organized Truck Platoons" (Figure 2) which was developed in the project "Operation-Scenarios for Advanced Driving Assistance Systems in Freight Transportation and Their Validation" (EFAS) (Henning and Preuschoff, 2003). In the scenario "Driver Organized Truck Platoons", the platoons can operate on today's existing motorways without extending the infrastructure and the driver has the permanent control of the autonomous driving procedures (Henning and Preuschoff, 2003). The creation of a platoon depends on the initiating driver who delivers the necessary data about time and place of meeting, the destination, as well as the required truck telemetric data (loading weight, engine power etc.) with the help of a Driver Information System (DIS). The high flexibility of truck transportation is not lost, because scheduling, like in rail traffic, is dispensable. After activating the ADAS, a selection of the best matching platoons is automatically shown. The ADAS informs the driver and prepares the participation to the selected platoon. The DIS acts as a human machine interface of the platoon system and helps the truck driver to plan the route and guides the driver to the meeting point (Friedrichs et al., 2008).

The driver has to initialize and respectively confirm all of the platoon maneuvers in order to build and to dissolve the platoon. As soon as the final position in the platoon is reached, an automated longi-

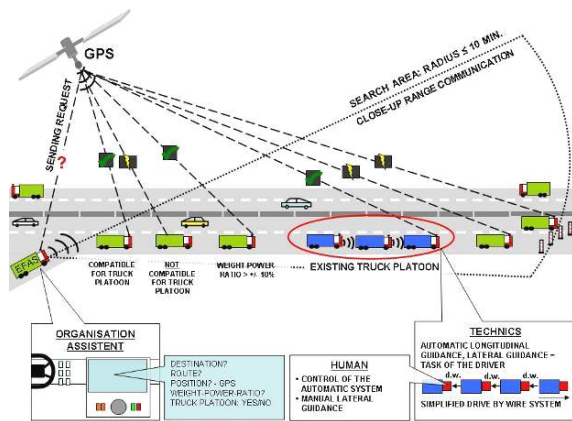


Figure 2: Scenario 1 - "Driver Organized Truck Platoons" (Henning and Preuschoff, 2003).

longitudinal guidance with a target distance of 10 meters between the trucks and a lateral guidance is possible. On one hand, this target distance was chosen because the short distance prevents most drivers from driving between the platoons. On the other hand, the short distance causes slipstream effects, which can lead to a reduced fuel consumption. Since road markings are needed for the lateral guidance, the platoon system is exclusively developed for the use on motorways. Because of a limitation for most trucks at approx. 50 mph, the speed of the trucks on motorways differs only slightly. Therefore, the truck platoons are operated at a speed between 37 and 50 mph. This speed can be managed safely at 10 meters distance by the KONVOI-System.

3 THE PLATOON SYSTEM

In order to realize different platoon sizes, four experimental vehicles have been equipped with the required automation-, information- and automotive-technology (Figure 3). The main components for the implementation of the system architecture in the experimental vehicles are the actuators (steering and power train), the sensors (object registration in close-up and far range, recognition of lane), the vehicle-vehicle-communication (WLAN), the automation unit (coordination of the different vehicle states), the control unit (adaptive cruise control and automatic guidance) and the driver information system (human-machine interface, organization assistant, GPS and 3G) (Henning et al., 2007).

The transversal guidance of the ADAS is based on the transversal offset to the leading vehicle and the recording of the own track position with a Comple-

mentary Metal Oxide Semiconductor (CMOS) image processing system as well as the analysis of the data flow from the vehicle-vehicle-communication. The necessary steering moment for the automated guidance of the trucks is realized via a steering actuator on the base of an electric motor in the vehicle, which is build as a dual circuit with detached energy supply.

The longitudinal guidance of the ADAS is based on a LIDAR (Light Detection and Ranging) distance sensor, a CMOS-Camera and a RADAR-sensor. The distance sensors are used to determine the distance in longitudinal direction and the lateral offset to the leading vehicle. The vehicle-vehicle-communication transfers necessary vehicle data from all platoon members, which are required for the ACC to realize the target distance of 10 meters. In all trucks, a target acceleration interface is implemented, which automatically calculates the drive-train and the management of the different brakes in the vehicles. The acceleration is either calculated autonomously for each vehicle or deduced from the data which is transferred via the vehicle-to-vehicle-communication.

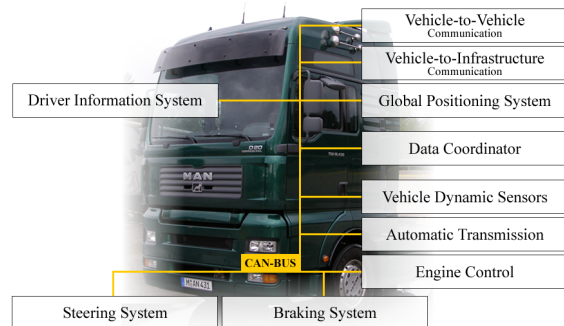


Figure 3: Automation-, Information- and Automotive-Technology of an Experimental Vehicle (Kunze et al., 2009).

Every experimental vehicle is attached with cameras which are able to identify the traffic lane, thus determining the position of every truck within the traffic lane. An electronically accessible steering system has been integrated additionally. A steering actuator on the base of an electric motor delivers the necessary steering moment for the automated guidance of the trucks (Friedrichs et al., 2008). With the help of the Driver Information System, the truck driver plans his route, selects economic platoon participants as well as initializes and respectively confirms the platoon manoeuvres in order to build and to dissolve the platoon. The platoon organization is realized on a central server with a data-mining-algorithm under consideration of economic aspects, which is part of this paper. For this task, the DIS has to send the time schedule, route plan and GPS position of the truck with

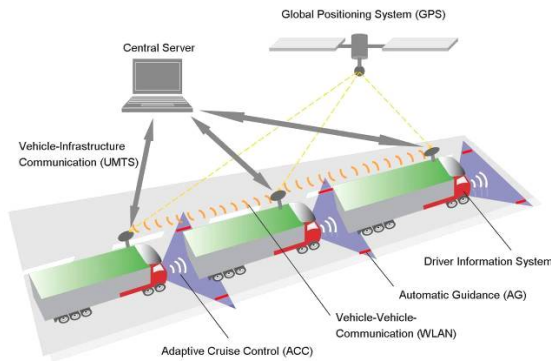


Figure 4: The Platoon System (Friedrichs, 2008).

a vehicle-infrastructure-communication via G3 to the central server (Figure 4).

4 EFFICIENCY OF TRUCK PLATOONS

4.1 Overview

The efficiency of truck platoons depends on several criteria. One important criterion for the efficiency of truck platoons is the profit which can be achieved by driving as a platoon. This profit can be simply calculated by the proceeds of less fuel consumption which depends on the length of the frequent sub-route (i.e. in kilometers) and the number of participants minus the extra personnel costs which have to be paid while others wait for the rest of the platoon participants.

4.2 Definitions

The term route is used in everyday language as a list of waypoints starting at a given point and leading to a specific destination. Before we define the term route for this paper, we want to define the distance between two waypoints as a sector, whereas the route, and thus the list of waypoints, is generated by a route planner based on a digital map (i.e. a graph of a chosen infrastructure) (Figure 5).

4.2.1 Definition "Section"

A *section* s is a non-divisible piece of a road, typically given by the underlying digital map. $K_A(s)$ identifies the start and $K_E(s)$ the end point of the section (e.g. as GPS-Coordinates). The distance of the section (e.g. in kilometers) is indicated by $Length(s)$. Furthermore, we define $Meet(s)$ as a meeting opportunity. $Meet(s)$ equals 1, if and only if there is a meet-

ing point (e.g. rest area) within the section s , otherwise $Meet(s)$ is equal to 0.

4.2.2 Definition "Route"

A *route* r is a sequence of *sections*, denoted as $r = (s_1, \dots, s_n)$, whereby for all $n > 0$ and $0 < i < n$ $K_E(s_i) = K_A(s_{i+1})$. A *part of a route* $r = (s_1, \dots, s_n)$ is a cut-out of r , i.e. for $i, j > 0$ and $i, j \leq n$ $Part(r)_i^j = \begin{cases} (s_i, \dots, s_j) & i \leq j \\ () & \text{else} \end{cases}$. A route $r' = (s'_1, \dots, s'_m)$ is called a *sub-route* of a route $r = (s_1, \dots, s_n)$, denoted as $r' \triangleright r$, if and only if $i > 0$ exists, so that $r' = Part(r)_i^{i+m-1}$. A route $r = (s_1, \dots, s_n)$ is *leading*, if for all $i, j \leq n$ and $i \neq j, s_i \neq s_j$ applies.

4.2.3 Definition "Platoon"

Let $R = (r_1, \dots, r_m)$ be a set of routes as defined above. A platoon is a route $l = (s_1, \dots, s_n)$ with $n > 0$, $Meet(s_1) = 1$ and $\exists w, v \in R, w \neq v \ l \triangleright w \cap l \triangleright v$. The *size of a platoon* l is defined as the number of possible platoon participants, i.e. $Size(l) = |\{w \in R \mid l \triangleright w\}|$. The *length of a platoon* l is defined as the distance which can be driven together as a platoon, i.e. $Length(l) = \sum_{i=1}^n Length(s_i)$.

4.3 Development of the Efficiency Criterion

For the organization of truck platoons with the aid of the central server and the associated search for platoon opportunities an efficient generation of offers is essential for the acceptance of the platoon system (section 3). The server provides all necessary functionalities to calculate the efficiency of truck platoons. For this purpose the following additional information for every section of a route s are important:

- $Length(s)$ is the length of the sections in kilometers,
- $STime_i(s)$ is the estimated arrival time of the truck i on this section,
- $ATime_i(s)$ is the estimated point of time, on which the truck i drives onto the section and
- $Type(s)$ is the well-defined type of street of the section s .

Because solely sections of highways are used for the formation of truck platoons, the type of street $Type(s)$ is needed for the filtering of the individual sections. For the efficient usage of truck platoons only three platoon-specific factors are taken into consideration. These are the driven distance of a platoon, the

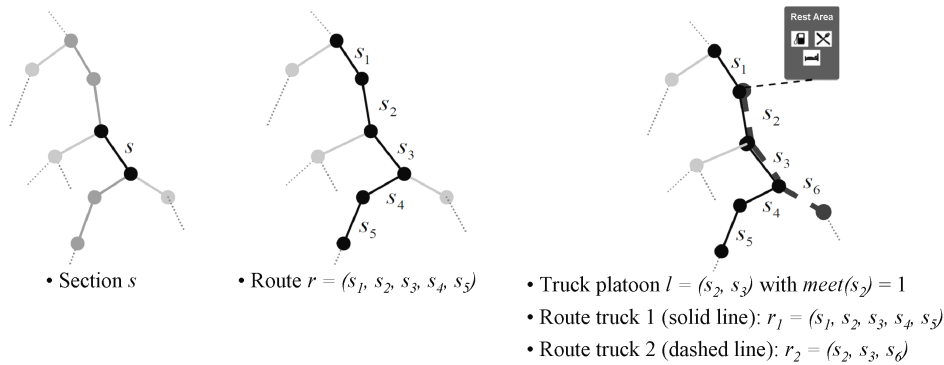


Figure 5: Examples for Sections (left), Routes (middle) and a Truck Platoon (right) (Friedrichs, 2008).

utmost waiting time and the utmost quantity of vehicles of a platoon. The platoon size $Size(l)$ is defined accordingly definition "Platoon" in section 4.3. For calculations the length of a platoon $Length(l)$ and the waiting time $Time_i(l)$ need route-specific data, which are provided by the server.

For the calculation of the route of a truck platoon $l = (s_1, \dots, s_n)$ applies:

$$Length(l) = \sum_{i=1}^n Length(s_i) \quad (1)$$

For the calculation of the waiting time for the participation on a truck platoon l of a truck i on a section (s_i) (independently of the section) applies:

$$Time_{i,s_j}(l) = \max_{1 \leq k \leq Size(l)} \{STime_k(s_j)\} - STime_i(s_j) \quad (2)$$

The efficiency of a truck platoon arises from the difference of the economic benefit (reduction of fuel consumption due to slip stream driving) and additional personnel costs (waiting time for platoon participants) stating a minimal common route within a platoon, maximal waiting time and a minimal revenue. The calculation of efficient truck platoons is based on this definition. A truck platoon l is efficient, if the additional personnel costs are smaller than the estimated revenue. Additionally a truck platoon l is efficient, if and only if the truck platoon is efficient for each individual platoon participant (Friedrichs, 2008).

$$Efficiency_i(l) = \min_{1 \leq k \leq Size(l)} \{Revenues_i(l) - Costs_i(l)\} \quad (3)$$

4.4 Calculation Basis for the Efficiency of Truck Platoons

4.4.1 Calculation of the Revenues

Schmitz (2004) calculates under consideration of the entire route of a truck platoon $Length(l)$, the quantity

of the participating vehicles and the fuel consumption u_i (in liter per 100km) the $Revenues_i(l)$ of a truck i , which can be realized due to the participation in a truck platoon l . The costs for one liter fuel are designated with k_{KS} . The reduction of the fuel consumption is indicated with $\Delta B_{e,Size(l)}$ (Schmitz, 2004).

$$Revenues_i(l) = u_i \cdot \frac{Length(l)}{100} \cdot \Delta B_{e,Size(l)} \cdot k_{KS} \quad (4)$$

The fuel consumption, which is induced by slip-stream driving in a platoon, can be theoretically calculated and practically measured. Several simulations and measurements have shown that the theoretical results cannot be achieved in practice (Table 1).

On account of this further research work took place. With the aid of traffic flow simulations the fuel reduction of the leading and the following vehicles have been analyzed. Under conservative assumptions the fuel consumption for truck platoons is implied with 2% for the 1st truck, 11% for the 2nd and 13% for every other participant. Because of the reason that the profit of driving in a platoon on 1st, 2nd or another position differs, it was furthermore assumed that every truck drives on average the same distance on each position of a platoon. Under those assumptions, the following equation 5 shows the fuel consumption of truck platoons with the size n .

$$\Delta B_{e,n} = \frac{2\% + 11\% + (n-2) \cdot 13\%}{n} \quad (5)$$

Table 2 shows the reduction of the fuel consumption of electronically coupled truck platoons based on equation 5.

With the equations 4 and 5 it is possible to calculate the revenues of truck i achieved by driving within the platoon l .

4.4.2 Calculation of the Costs

The costs for participating in a truck platoon are caused by several criteria. First of all, the participants have to wait for other participants at an arranged

Table 1: Reduction of Fuel Consumption based on Theory, Simulation and Test (Bonnet and Fritz, 2002).

	THEORETICAL	SIMULATION DAIMLER	MEASUREMENT DAIMLER
Fuel Consumption (1st vehicle)	2,17% (14t); 1,64% (28t)	2% (28t)	6% (14t)
Fuel Consumption (2nd vehicle)	38,06% (14t); 28,76% (28t)	19% (28t)	21% (28t)

Table 2: Reduction of Fuel Consumption due to Electronically Coupled Platoon Driving.

PLATOON STRUCTURE	FUEL REDUCTION
1st Platoon Vehicle (Leading Vehicle)	2%
2nd Platoon Vehicle (Following Vehicle)	11%
3rd Platoon Vehicle (Following Vehicle)	13%
Average Fuel Reduction, 2 Vehicles	6.50%
Average Fuel Reduction, 3 Vehicles	8.67%
Average Fuel Reduction, 4 Vehicles	9.75%
Average Fuel Reduction, 5 Vehicles	10.40%

meeting point or reduce speed until they got together. Another criterion for extra personnel costs is the platoon driving itself and the associated platoon maneuvers (e.g. a platoon has to be dissolved at a working area). It is also possible that a truck, which participates in a platoon, has to drive with reduced speed because of another, slower participant (e.g. uphill).

Due to the fact that the extra personnel costs caused by platoon maneuvers or speed reduction because of slow participants are unascertainable without accurate road and traffic information, therefore those influencing criteria will be ignored. The only criteria which take account of extra personnel costs will be the waiting time $Time_i(l)$ for each participating truck i and the wage rate k_{PK} .

$$Costs_i(l) = k_{PK} \cdot Time_i(l) \quad (6)$$

4.4.3 Boundaries of the Efficiency of Truck Platoons

To get a statement on the maximum of waiting time, equation 3 has to be transformed and complemented by equations 4 and 6. The efficient criterion of electronically coupled truck platoons is originated in equation 7.

$$u_i \cdot \frac{Length(l)}{100} \cdot \Delta B_{e,Size(l)} \cdot \frac{k_{KS}}{k_{PK}} > Time_i(l) \quad (7)$$

With this efficient criterion it can be easily proofed, if the benefit of the operation of electronically coupled truck platoons is bigger than their costs.

5 DATA MINING TECHNIQUE

5.1 Related Work

Discovering patterns in sequences is an important knowledge discovery and data-mining research area. There are many different interesting fields like discovering rules in so called events (Mannila et al., 1997) or finding text subsequences (Califano and Rigoutsos, 1993) that match a regular expression. In 1995, Agrawal and Srikant introduced the problem of mining sequential patterns (Agrawal and Srikant, 1995): "A database with customer transactions is given. Each transaction consists of the fields; customer-id, transaction-time and a set of items purchased in this transaction. Quantities of items purchased in a transaction are not considered. The problem of mining sequential patterns is to find all sequences that have a certain user-specified minimal support." This difficulty deals with the questions; "which items are bought in which order" or "which item will be bought next". Agrawal and Srikant presented three algorithms for solving the sequential pattern problem and many other algorithms have been proposed to speed up the mining process (e.g. (Zaki, 2001); (Pei et al., 2004)). In general, those algorithms are based on two different methods for mining sequential patterns: apriori-based methods and pattern-growth methods (Antunes and Oliveira, 2004).

The apriori-based methods are, as the name states, based on the so called apriori condition. This condition is used to generate larger candidates based on smaller frequent sequences in the so called join-step. Each generated candidate is then validated in the following so called prune-step. These two steps are repeated until there are no more frequent sequences found. The pattern-growth methods deduce from a smaller frequent sequence, a longer sequence. Instead of generating candidates, these methods directly look for frequent sequence.

In 2003, Yan et al. introduced the problem of mining closed sequential patterns which means that in-

stead of finding all frequent sequences, only the "most supported" sequences are searched for (Yan et al., 2003). They presented a new algorithm to find closed sequential pattern, called CloSpan, which is based on a pattern-growth method introduced in 2001 by Pei et al. Figure 6 illustrates the main idea of the approach. The items are nodes in a lexicographical sequence tree. Each path of the tree is a frequent sequence found in the database. So called projected databases are stored for each node of the tree. If the projected databases of two different paths are equal to each other, the mining for more frequent sequences can be eventually stopped. In figure 6, the mining for further frequent sequences can be stopped at node *B* in the path *B* because the projected database of this node is equal to the projected database of node *B* in the path *A, B*.

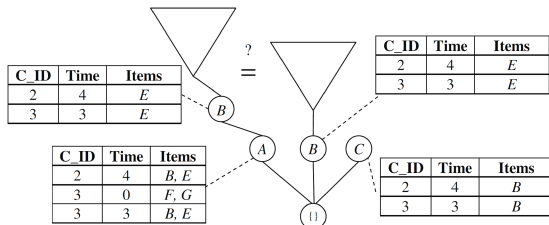


Figure 6: Illustration of the Algorithm to Find Closed Sequential Pattern.

The problem of mining frequent sub-routes or economic truck platoons cannot be solved easily by any of those algorithms. Due to the small support of two and the given sequences of hundreds of sections, those algorithms collapse. In the case of platoons the support has to be two, because two is the smallest number of participants to form a platoon. However, these algorithms are good approaches in finding possible solutions for solving this problem.

5.2 The Truck Platoon Sequential Pattern Algorithm

The so called TPSpan-Algorithm (Mining Truck Platoon Sequential pattern) had been already presented in detail in 2008 (Meisen et al., 2008). In this section, we will give a brief introduction into the TPSpan-Algorithm which searches for possible platoons in a route database, especially optimized to find economic truck platoons. The algorithm is split in four phases and illustrated in Figure 7.

In the first phase, the exclusion criteria are used to eliminate trucks or routes which do not meet those criteria. Within the second phase, the grouping criteria are applied to secure the forming of platoons within similar conditions. In the third phase, the data-mining

technique checks each group for possible economic truck platoons regarding the assessment criteria. Finally, in the fourth and last phase, the economic truck platoons are grouped by truck-id. The grouped information is then transmitted to each truck (e.g. meeting points, profit).

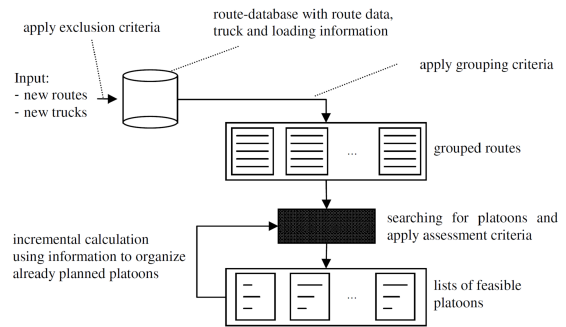


Figure 7: Illustration of the Algorithm to Find Truck Platoon Sequential Pattern.

The used data-mining technique in the third phase is based on the projected pattern-growth idea presented by Pei et al. (2001) and has been enhanced for solving the problem of mining frequent sub-routes. It generates a lexicographic tree as shown in Figure 6. Instead of starting with each frequent section, the tree starts with those frequent sections which have a meeting opportunity ($Meet(s_1) = 1$). This means that the first section has to be e.g. a rest area. Due to this, the width of the tree is reduced and the condition for a platoon (starting with a meeting opportunity) is ensured. Another important improvement is the so called node-compress-method which is used to reduce the validation of the assessment criteria. Instead of validating the criteria for every platoon, the common distance of a platoon will be increased as long as the number of participants does not change. It is ensured that the algorithm will still search for the most economic platoons: A platoon $l_1 = (s_1, \dots, s_n)$ with $Size(l_1) = k$ is always more economical than a platoon $l_2 = (s_1, \dots, s_n)$ with $Size(l_2) = k$, whereas $Length(l_1) > Length(l_2)$.

Figure 8 shows the framework of the data mining technique. The algorithm is working from the root of the tree to the leaves. In each node, the algorithm calculates possible platoons (line 4) and determines new children (line 6). If the projected database of the node does not support any more platoons (line 1) the recursion terminates. The framework also shows the already mentioned node-compress-method (line 3) which returns a true value if the compression terminated with a split. If this is the case, further children are possible, other wise the routes of the projected database end and no more children can be determined (line 5).

```

Node.calculate()
1. if (Node.Depth == 1 && ProjRDB.Support < MinKonvoiSize)
2.   Return
3. bSplit = ProjRDB.compress()
4. ProjRDB.calculateKonvois()
5. if (bSplit)
6.   Node.createChilds()
7.   for each NodeChild in Node
8.     NodeChild.calculate()

```

Figure 8: Illustration of the Algorithm to Find Truck Platoon Sequential Pattern.

As shown in figure 7 the algorithm also uses information that has already been calculated to organize planned platoons by calculating the tree incrementally. Comparing the differences between the trees enables the organization of platoons, e.g. delay of participants at the meeting point or canceling of planned platoons. The main goal of the incremental calculation is to reach a comparison of root-nodes. If a root-node, i.e. a node on the first level of the tree, has been changed, the sub-tree is recalculated and existing changes are recorded, analyzed and submitted.

5.3 Consideration of the Efficiency of Truck Platoons within the Data-mining Technique

To find efficient truck platoons these calculations are executed by the TPSSpan-Algorithm in row 4 *ProjRDB.calculateKonvois()*. Figure 9 shows the algorithm for the calculation of efficient and economic truck platoons for a projected data base.

In the first row the algorithm proofs if the length of the projected data base is bigger than or as big as the minimal route of a truck platoon. The length matches the sum of the distances of the individual atomic routes. In the following row a maximal window is calculated, which specifies the maximum waiting time for a truck to meet potential platoons.

```

ProjRDB.calculateKonvois()
1. if (ProjRDB.Distance < clMinLength) return
2. lTimeFrame = maximalTimeToWait()
3. if (lTimeFrame < 0) return
4. PossKonvois = Get2Sets(ProjRDB, lTimeFrame)
5. newPossKonvois = PossKonvois
6. while newPossKonvois != empty || clMaxKonvoiSize reached
7.   newPossKonvois = AprioriCandidateGen(newPossKonvois)
8.   PossKonvois = newPossKonvois
9. for each Konvoi in PossKonvoi
10.  if (calcProfit(Konvoi) > 0)
11.    Konvois.add(Konvoi)

```

Figure 9: Algorithm for the Calculation of efficient truck platoons (Meisen, 2006).

The calculation of efficient truck platoons are based on the efficient criterion of electronically coupled truck platoons (former equation 7):

$$u_i \cdot \frac{Length(l)}{100} \cdot \Delta B_{e.Size(l)} \cdot \frac{k_{KS}}{k_{PK}} > Time_i(l) \quad (8)$$

Because the estimated fuel reduction is unknown at this point of time the maximum value has to be assumed. With this procedure the algorithm searches in row 4 with the function *Get2Sets(ProjRDB, lTimeFrame)* all two elementary sets of projected routes, for which the estimated waiting time is smaller than the calculated time window. Afterwards, by means of apriori-based methods further larger quantities are formed, whereas the parameter "maximal platoon size" serves as an upper bound (row 5 to 8).

6 EXPERIMENTAL RESULTS

For evaluation and simulation purposes synthetic data were generated by calculating random routes using a routing planner. To do so, two zip codes were randomly chosen from all postal codes in Germany. The departure time was also randomly picked within one day. The calculated routes were saved in a database. With this approach three different databases with 500, 2,000 and 5,000 routes were generated. Experimental evaluation has shown that our TPSSpan-Algorithm finds profitable truck platoons within generated databases. The experiments have also shown that the number of economic truck platoons increase exponentially with the number of routes and that scaling parameters are necessary to prune the amount of possible truck platoons (i.e. when there are more than 2,000 routes). Within the experimental evaluations, we identified the following scaling parameters: minimal common distance in kilometers, minimal and maximal truck platoon size, maximal waiting time in seconds and minimal profit in (Figures 10 to 12).

Figures 10 to 12 show individual diagrams about the efficiency, the flexibility and the reliability of truck platoons. The minimal and maximal truck platoon size was set to two and four. The maximal waiting time was 10,000 s, the minimal common distance was 1 km and the minimal profit was 1, if not chosen to be the value of the x-axis. We say that a platoon is efficient, if and only if the proceeds are above the costs. Furthermore, we say that a platoon is flexible, if alternatives to participate with a platoon are given. The reliability of truck platoons focus on the chance of finding a platoon to participate with. The experiments have shown that the best values for the known pruning parameters are a maximal waiting time of 3,000 s, a minimal common distance of 100 km and a minimal profit of 1. With those values for the pruning parameters, efficiency, flexibility and reliability are still assured (Friedrichs, 2008).

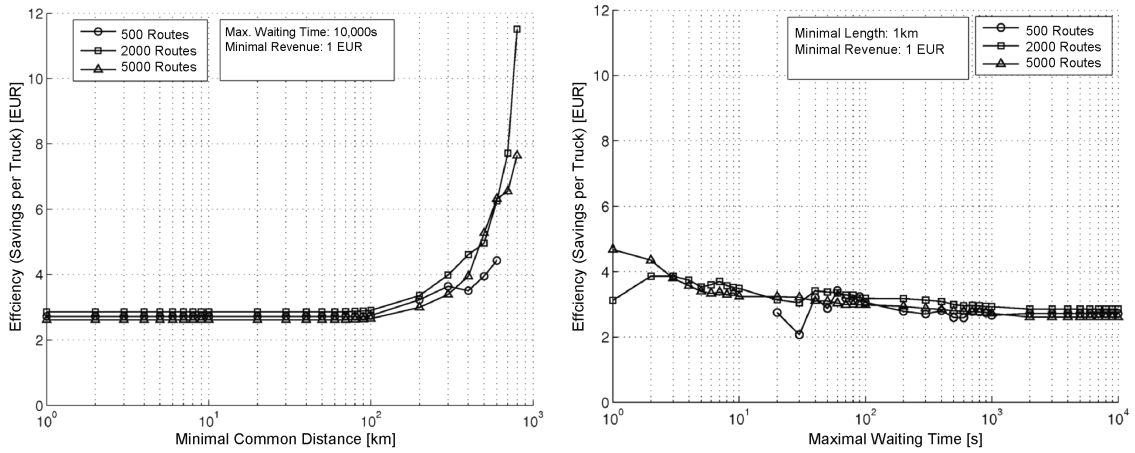


Figure 10: Efficiency depending on minimal common Distance of a Platoon (left) and maximal Waiting Time (right) (Friedrichs, 2008).

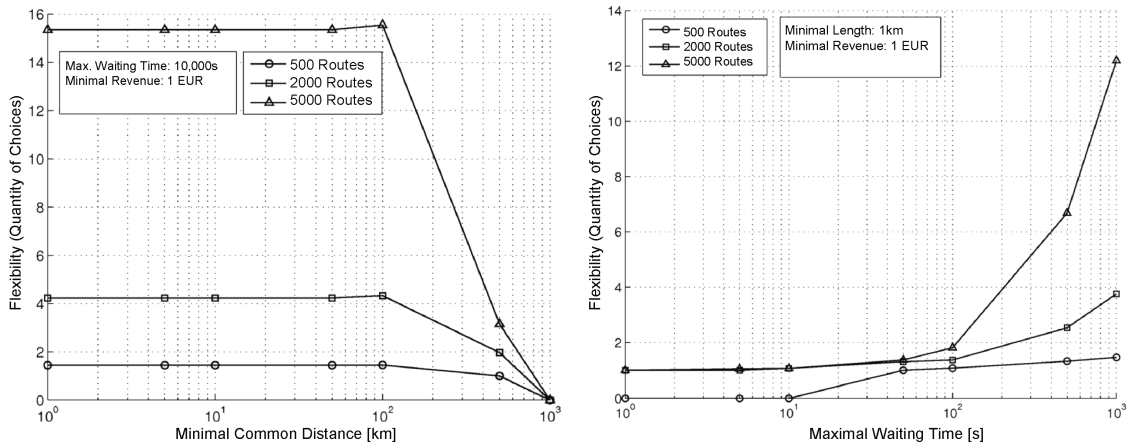


Figure 11: Flexibility depending on minimal common Distance of a Platoon (left) and maximal Waiting Time (right) (Friedrichs, 2008).

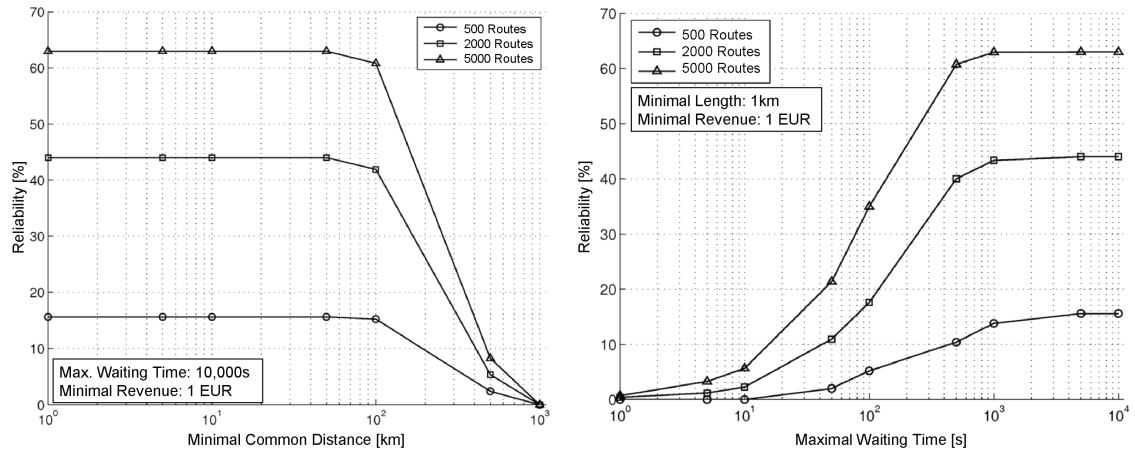


Figure 12: Reliability depending on minimal common Distance of a Platoon (left) and maximal Waiting Time (right) (Friedrichs, 2008).

7 CONCLUSIONS

In this paper, we have introduced a data mining technique to plan and organize platoons. Furthermore, we have introduced and presented experimental results for an application area "truck platoons: trains on road". For this case, we have examined that it is possible to find truck platoons and that the amount of platoons increase exponentially with the amount of routes (participating trucks). Due to this rise, pruning parameters are necessary. Further experiments have shown that it is possible to find truck platoons reliable, efficient and flexible, even if pruning parameters are used. We have given suggestions for these parameters to achieve the mentioned factors. Further work has to be done in the field of realization. We have presented the dilemma of different results of reducing fuel consumption. Additional effort has to be put into the goal closing the gap between results of fuel reduction based on theory, simulation and tests and thus provide a further substantial contribution to the efficiency of truck platoons.

REFERENCES

- Agrawal, R. and Srikant, R. (1995). Mining Sequential Patterns. In *Proc. of the 11th Intl Conf. on Data Engineering, Taipei (Taiwan)*.
- Antunes, C. and Oliveira, A. L. (2004). Sequential Pattern Mining Algorithms: Trade-offs between Speed and Memory. In *Proc. of the 2nd Intl Workshop on Mining Graphs, Trees and Sequences (ECML/PKDD)*.
- Bonnet, C. and Fritz, H. (2002). Fuel Consumption Reduction Experienced By Two PRO-MOTE-CHAUFFEUR Trucks. In *Electronic Towbar Operation*.
- Califano, A. and Rigoutsos, I. (1993). Flash: A fast look-up algorithm for string homology. In *Proc. of the 1st Intl Conf. on Intelligent Systems for Molecular Biology, Bethesda (USA)*.
- Commission of the European Communities, E. U. (2003). *Europe at a crossroad - The need for sustainable transport*. Brussels, Belgium.
- Commission of the European Communities, E. U. (2006). *Keep Europe moving Sustainable mobility for our continent*. Brussels, Belgium.
- Economic and Social Commission for Asia and the Pacific, U. N. (2008). *Statistical Yearbook for Asia and the Pacific*. United Nations Publication, Bangkok, Thailand.
- Friedrichs, A. (2008). *A Driver Information System for Truck Platoons*. VDI, Duesseldorf, Germany.
- Friedrichs, A., Meisen, P., and Henning, K. (19.-22.05.2008). *A Generic Software Architecture for a Driver Information System to Organize and Operate Truck Platoons*. International Conference on Heavy Vehicles (HHVT2008), Paris, France.
- Henning, K. and Preuschoff, E. (2003). *Einsatzszenarien fuer Fahrerassistenzsysteme im Strassengueterverkehr und deren Bewertung*. VDI, Duesseldorf, Germany.
- Henning, K., Wallentowitz, H., and Abel, D. (2007). Das Lkw-Konvoisystem aus den Perspektiven Informations-, Fahrzeug- und Automatisierungstechnik. In *Mechatronik 2007 - Innovative Produktentwicklung*.
- Kunze, R., Tummel, C., and Henning, K. (19.-20.12.2009). *Determination of the Order of Electronically Coupled Trucks on German Motorways*. 2nd Conference on Power Electronics and Intelligent Transportation System (PEITS 2009), Shenzhen, China.
- Mannila, H., Toivonen, H., and Verkamo, A. I. (1997). Discovery of frequent Episodes in Event Sequences. In *Data Mining and Knowledge Discovery*.
- Meisen, P. (2006). *Data-Mining Technique for Truck Platoons*. RWTH Aachen University, ZLW/IMA, Aachen, Germany.
- Meisen, P., Henning, K., and Seidl, T. (2008). A Data-Mining Technique for the Planning and Organization of Truck Platoons. In *Proceedings of the International Conference on Heavy Vehicles*.
- Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., and Hsu, M.-C. (2004). PrefixSpan: Mining Sequential Patterns Efficiently by PrefixProjected Pattern Growth. In *Proc. of the Intl. Conf. Data Engineering (ICDE'01)*.
- Savelsberg, E. (2005). *Lastenheft fuer elektronisch gekoppelte Lkw-Konvois*. VDI, Duesseldorf, Germany.
- Schmitz, J. (2004). *Energy-economic cost-benefit-analysis of electronically coupled truck convoys*. RWTH Aachen University, ZLW/IMA, Aachen, Germany.
- Yan, X., Han, J., and Afshar, R. (2003). CloSpan: Mining Closed Sequential Patterns in Large Datasets. In *SDM'03*.
- Zaki, M. J. (2001). SPADE: An Efficient Algorithm for Mining Frequent Sequences. In *Proc. of Machine Learning Journal, Special issue on Unsupervised Learning (Doug Fisher, ed.)*.

MORPHING WING REAL TIME OPTIMIZATION IN WIND TUNNEL TESTS

Andrei V. Popov, Lucian T. Grigorie, Ruxandra Botez
École de Technologie Supérieure, Montréal, Québec, H3C 1K3, Canada
Andrei-vladimir.popov.1@ens.etsmtl.ca, ltgrigorie@yahoo.com, Ruxandra@gpa.etsmtl.ca

Mahmoud Mamou, Youssef Mébarki
Institute for Aerospace Research, NRC, Ottawa, Ontario, K1A 0R6, Canada
{mahmood.mamou, youssef.mebarki}@nrc-cnrc.gc.ca

Keywords: Morphing Wing, Real Time Optimization, Wind Tunnel, Laminar-to-turbulence Transition.

Abstract: In this paper, wind tunnel results of a real time optimization of a morphing wing in wind tunnel for delaying the transition towards the trailing edge are presented. A morphing rectangular finite aspect ratio wing, having a wind tunnel experimental airfoil reference cross-section, was considered with its upper surface made of a flexible composite material and instrumented with Kulite pressure sensors, and two smart memory alloys actuators. Several wind tunnel tests runs for various Mach numbers, angles of attack and Reynolds numbers were performed in the 6'×9' wind tunnel at the Institute for Aerospace Research at the National Research Council Canada. Unsteady pressure signals were recorded and used as feed back in real time control while the morphing wing was requested to reproduce various optimized airfoils by changing automatically the two actuators strokes. The paper shows the optimization method implemented into the control software code that allows the morphing wing to adjust its shape to an optimum configuration under the wind tunnel airflow conditions.

1 INTRODUCTION

The Consortium for Research and Innovation in Aerospace in Quebec CRIAQ 7.1 project was a collaborative project between the teams from École de technologie supérieure (ETS), École Polytechnique, the Institute for Aerospace Research - National Research Canada (IAR-NRC), Bombardier Aerospace, Thales Avionics. In this project, the laminar flow past aerodynamically morphing wing was improved in order to obtain important drag reductions.

This collaboration called for both aerodynamic modeling as well as conceptual demonstration of the morphing principle on real models placed inside the wind tunnel. Drag reduction on a wing could be achieved by modifications of the airfoil shape which had an effect in the laminar to turbulent flow transition point position. The main objective of this concept was to promote large laminar regions on the wing surface by moving the transition point toward the trailing edge of the airfoil wing, thus reducing drag over an operating range of flow conditions

characterized by Mach numbers, airspeeds and angles of attack (Zingg, 2006).

To modify the laminar flow around a wing airfoil, it was necessary to change the airfoil shape; therefore one of the methods for airfoil changes was developed at Kentucky University, which consisted of deflecting the wing upper surface using adaptive actuators (Jacob and Munday, 1998, 1999, 2002).

It was shown that the actuators activated oscillatory motions of a certain frequency to the boundary layer flow over the upper surface. These actuators were made of piezo-electric materials which changed their shapes when connected to an electrical current differential voltage. The wind tunnel tests showed that the displacement of the transition point to the trailing edge resulted in the drag decrease and in the lift increase (Jacob, 1999).

The “chordwise air collection” method was designed in 1984, in fact, laminar flow control (LFC) over the upper surface of the wing was realized by the boundary layer suction, thereby moving the transition position at 85% of the chord (Allison, 1978).

A numerical algorithm was developed for optimizing the suction distribution, by maintaining the transition at a desired location (chord %) and maintaining the energy spent at a minimum (Hackenberg, 1995). Three steps were considered: 1. Boundary layer computation; 2. Transition prediction, and 3. Optimization of the suction distribution while maintaining the transition location at a certain desired percentage of the chord. In the third step, the gradient method was used.

Optimized wings were conceived by reducing the kinetic energy of the perturbation and drag values while, the lift and pitch moment coefficients were maintained at desired values (Pralits, 2003).

A controller was developed at Southampton University, with the aim of maintaining the desired turbulence level over a flat plate equipped with a suction porous panel. The pressure fluctuations were measured with microphones at the boundary layer over the flat plate, and the signal was conditioned and filtered to remove the background noise of the wind tunnel fan, then the turbulence level was estimated by computing the RMS (Root Mean Square) pressure signal. The controller used the error between the RMS values of the measured pressures and the desired RMS values at the spots where the microphones were installed, thus maintaining the transition on the specified area over the flat plate (Rioual, 1994).

Closed-loop control of the morphing platform (wing-shape control) and simultaneously enforced prescribed closed loop aircraft dynamics (flight control) were modeled (Gandhi, 2007). The N-MAS wing designed by NextGen Aeronautics was considered. The flight control law actively used the leading edge morphing wing sweep angle as an actuator to assist in manoeuvres while guaranteeing aircraft stability. The morph between the two modes of Loiter and Dash was modeled by a first-order transfer function, and hence the morph rate was governed by the transfer function time constant.

The empirical structural weights for various wing geometries were obtained by implementing two finite element-based structural optimization methods: 1) an aggregate and 2) a simultaneous analysis (Skillen, 2005). These methods were applied on a morphing wing with two degrees of freedom: the wing sweep and the root chord length. Two linear actuators were used: one positioned along and parallel to the forward spar and the other one positioned along and parallel to the wing root chord. These geometrical variations produced four configurations with changes in area, aspect ratio and sweep: the high lift configuration for the largest area and minimal sweep angle; the loiter configuration

for the maximum aspect ratio and minimum sweep angle; the dash/cruise configuration for the maximum sweep and minimal area; and the manoeuvre configuration for the maximum area at the maximum sweep.

A symmetric wing structure was created with two tapered graphite/epoxy composite plates and a steel body. Four pairs of SMA wires were attached to the wings' bottom surfaces in the chord-wise direction. Lift and drag forces were measured at various angles of attack. Dynamic vibration signals were measured by Fiber Bragg Grating FBG sensors at the wing root and were used to monitor aeroelastic unstable flutter phenomena, at various angles of attack (Yang, 2006).

A wing structure comprised of an optimized internal layout of cables and struts was able to change its shape. Cables were used as actuators, while struts provided rigidity to the wing. In addition to achieving continuous morphing by changing cable lengths, this structure had the advantages of being light weighted and having a distributed actuation. Topology optimization was used to optimally place cables and struts in a bay or in a wing section. The Non-dominated Sorting Genetic Algorithm II (NSGA II) was used for modeling the NASA HECS and the NextGen TSCh wings (Bharti, 2006).

Wings roll performances were achieved by use of articulated conformal control surfaces. Analysis results were compared to experimental results obtained for a 16% scale model of a fighter wing equipped with embedded smart materials used to deform a control surface. The control surface design was found suitable for low-rate applications such as takeoff and landing configurations (Sanders, 2003). NextGen developed an in-plane morphing geometry concept. Flexible elastomeric skins with out-of-plane stiffeners accommodated the wing motion while transmitting air pressure loads to the wing substructure. Wind tunnel testing of a full-scale wing for a 2400 lb vehicle and flight testing of a subscale unmanned aerial vehicle (UAV) were performed. The following issues were identified: the need to address multiple geometries and flight envelopes to account for morphing shape changes; the in-plane wing flexibility resulting from its mechanism restraint by linear actuators. Another half-span wind tunnel model was tested in the NASA Transonic Dynamics Tunnel for aeroelasticity studies (Andersen, 2007).

The Flight Test results of a Mission Adaptive Compliant Wing (MAC-Wing) variable geometry Trailing Edge Flap with a Natural Laminar Flow NLF airfoil have been described. The MAC-Wing technology provided light-weight, low power,

variable geometry reshaping of the upper and lower flap surface with no discontinuities. The airfoil-flap system was optimized to maximize the laminar boundary layer extent over a broad lift coefficient range for endurance aircraft applications. The expanded laminar bucket capability allowed the endurance aircraft to extend their range, by 15% or more, by optimizing the lift-to-drag ratio (L/D) throughout the mission. The wing was tested at full-scale dynamic pressure, Mach number, and reduced-scale Reynolds numbers on Scaled Composites' Knight Aircraft. Laminar flow regime occurred up to 60% chord of the wing during tests. Significant fuel and weight savings as well as high control authority were verified by tests and analyses. Fifteen Dantec Dynamics hot film sensors measured the boundary layer transition position (De Breuker, 2007).

A Defense Advances Research Projects Agency DARPA sponsored wind tunnel test model of a Lockheed Martin morphing concept was designed and tested. The wind tunnel model incorporated the key features representatives of a full scale vehicle model: out-of-plane morphing through a coordinated actuation system integrated with seamless skins and a composite support structure that encompassed the actuator system along the wing fold hinge lines, structural layout and materials featured in the full scale vehicle design, and a first-time thermopolymer actuator integral to a leading edge device for smooth contour between the inner wing and fuselage when fully morphed. The model was instrumented with strain gauges, accelerometers and pressure transducers; data was acquired and correlated with aircraft design and analysis methods (Love, 2007).

The airborne modification of an aircraft wing airfoil shape could be realized continuously to maintain laminar flow over the wing surface as flight conditions changed. To achieve such a full operating concept, a closed loop control system concept was developed to control the flow fluctuations over the wing surface with the deformation mechanisms (actuators) (Popov, 2008).

The wing model had a rectangular plan form of aspect ratio of 2 and was equipped with a flexible upper surface skin on which shape memory alloys actuators were installed (Coutu, 2009 and Georges, 2009). Two shape memory alloys (SMA) actuators created the displacement of the two control points on the flexible skin in order to realize the optimized airfoil shapes (Sainmont, 2009).

As reference airfoil, the laminar Wind Tunnel Experimental Airfoil WTEA was used because it was already optimized for laminar flow in the

transonic regime; its aerodynamic performance was investigated at IAR-NRC (Khalid, 1993). The optimized airfoils were previously calculated by modifying the reference airfoil for each airflow condition as combinations of angles of attack and Mach numbers such that the transition point position was found to be the nearest as possible to the airfoil trailing edge (Sainmont, 2009). Several optimized airfoils were found for the airflow cases combinations of Mach numbers and angles of attack. The optimized airfoils configurations were stored in the computer memory by means of a database and were selected as needed by the operator or computer in order to be realized by the morphing wing (Popov, 2009). But this strategy relied on the previously calculated aerodynamic characteristics of the airfoils which usually were determined by use of CFD codes and optimization algorithms.

The idea presented in this paper was to implement the same optimization algorithm into the computer controller that would search the optimal configuration with the real system, in real time and for real aerodynamic airflow conditions. The basic idea of optimization control is to by-pass the necessity of a previously calculated optimized airfoils database, and to generate in real time the optimized airfoil for the exact conditions of the wind flow. For such a task it was necessary to develop a subroutine that optimized the airfoil shape in the same way in which the optimized airfoils database was generated. The method of optimization used in this case was a mixed method between 'the gradient ascent' or 'hill climbing' method and the 'simulated annealing' which is a meta-heuristic search method.

The 'hill climbing' method is a local search optimization algorithm. It considers a random poor solution, which improves slowly by iterations. When solution cannot be further improved, it becomes the final solution and the algorithm ends. In our case, the searching domain is defined by the actuators displacements as variables, while the cost function needed to be maximized is the transition point position x_{tr} . Normally the function is defined analytically and the maximum is searched along the lines with the maximum local derivatives or gradients. Although 'hill climbing' would be very fast and simple to program, the solution found is not guaranteed to be the global maximum of the entire search domain (Hill climbing, from Wikipedia). Other local search algorithms such as 'stochastic hill climbing', 'random walks' and 'simulated annealing' would overcome this problem. The characteristic of these methods is that the algorithm searches random solutions within the search domain

in order to cover all the possible local maxima and to find the global maximum (Simulated annealing, from Wikipedia).

The reason why a mixed method was needed, was because the cost function for such complex problem (minimize the CD, maximize the CL/CD or maximize the transition point position x_{tr} for a morphing wing) was not defined analytically and the implementation of ‘gradient ascent’ method was not suitable. Also, due to time cost (very long time response of the SMA actuators due to heating but especially cooling time), a purely probabilistic meta-heuristic search algorithm such as ‘stochastic hill climbing’, ‘random walks’ and ‘simulated annealing’, was not suitable too.

The idea of the present algorithm was the mixture of the ‘hill climbing’ method with the ‘random walks’ or ‘simulated annealing’ and the search within the defined domain nine points, one being the center of a circle while the other eight being situated on the circle with a predefined radius. When the maximum is found within the nine points, the algorithm reset the next searching step by iterating with eight points situated on smaller circles until the global maximum is found. This mixed method was found to be the fastest i.e. it considered the least number of points evaluated for converging to the transition point position x_{tr} maximum.

2 EXPERIMENTAL SETUP DESCRIPTION

2.1 Mechanical and Electrical Control System

The concept of this morphing wing consisted in a rectangular wing model (chord $c = 0.5$ m and span $b = 2.1$ m) incorporating two parts. One fixed part was built in aluminum by the IAR-NRC team which sustained the resistance forces acting during wind tunnel tests. The other part consisted in a flexible skin installed on a metallic structure on the wing upper surface and was designed and manufactured at Ecole de Technologie Supérieure (Fig. 1). The flexible skin was required to change its shape through two action points in order to realize the optimized airfoil for the airflow conditions in which tests were performed.

The actuators were composed of two oblique cams sliding rods span-wise positioned that converted the horizontal movement along the span in vertical motion perpendicular to the chord (Fig. 2).

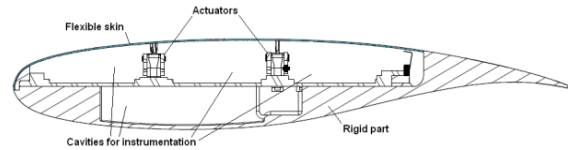


Figure 1: Cross section of the morphing wing model.

The position of each actuator was given by the mechanical equilibrium between the Ni-Ti alloy SMA wires that pulled the sliding rod in one direction and the gas springs that pulled the sliding rod in the reverse direction. The gas springs role was to counteract the pulling effect of aerodynamic forces acting in wind tunnel over the flexible skin when the SMA’s were inactive. Each sliding rod was actuated by means of three parallel SMA wires connected to a current controllable power supply which was the equivalent of six wires acting together. The pulling action of the gas spring retracted the flexible skin in the undeformed-reference airfoil position, while the pulling action of the SMA wires deployed the actuators in the load mode i.e. morphed airfoil in the optimized airfoil position (see Fig. 2). The gas springs used for these tests were charged with an initial load of 225 lbf (1000 N) and had a characteristic rigidity of 16.8 lbf / in (2.96 N / mm).

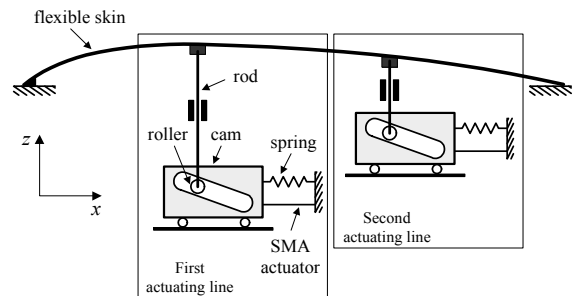


Figure 2: Schematics of the flexible skin mechanical actuation.

The mechanical SMA actuators system was controlled electrically through an ‘open loop’ control system. The architecture of the wing model open loop control system, SMA actuators and controller is shown in Figure 3. The two SMA actuators had six wires each, which were supplied with power by the two AMREL SPS power supplies, controlled through analog signals by the NI-DAQ USB 6229 data acquisition card. The NI-DAQ was connected to a laptop through an USB connection. A control program was implemented in Simulink which provided to the power supplies the needed SMA current values through an analog signal as

shown in Figure 3. The control signal of 2 V corresponded to an SMA supplied current of 33 A. The Simulink control program used as feedback three temperature signals coming from three thermocouples installed on each wire of the SMA actuator, and a position signal from a linear variable differential transducer (LVDT) connected to the oblique cam sliding rod of each actuator. The temperature signals served in the overheat protection system that disconnected the current supply to the SMA in case of wire temperature passed over the set limit of 120°C. The position signals served as feedback for the actuator desired position control. The oblique cam sliding rod had a horizontal versus vertical ratio 3:1; hence the maximum horizontal displacement of the sliding rod by 24 mm was converted into a maximum vertical displacement of the actuator and implicit of the flexible skin by 8 mm.

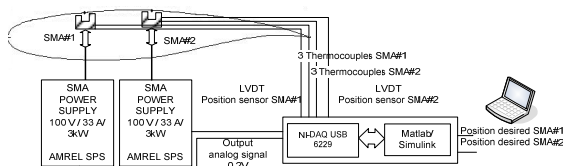


Figure 3: Architecture of the morphing wing model control system.

A user interface was implemented in Matlab/Simulink which allowed the user to choose the optimized airfoils shape from database stored on the computer hard disk and provided to the controller the vertical needed displacements in order to obtain the desired optimized airfoil shape. The controller activated the power supplies with the needed SMA current values through an analog signal as shown in Figure 3. In practice, the SMA wires were heated at an approximate temperature of 90°C with a current of 10 A. When the actuator reached the desired position the current was shut off and the SMA was cycled in endless heating/cooling cycles through the controller switching command on/off of the current in order to maintain the current position until another desired position or the entire system shut off was required.

In support of the discrete pressure instrumentation, infrared thermography (IR) visualization was performed to detect the transition location on the morphing wing upper surface and validate the pressure sensor analysis. The transition detection method using IR was based on the differences in laminar and turbulent convective heat transfer coefficient and was exacerbated by the artificial increase of model-air flow temperature

differences. In the resulting images, the sharp temperature gradient separating high temperature (white intensity in image) and low temperature (dark intensity) regions was an indication of the transition location. The infrared camera used was an Agema SC3000 camera, equipped with a 240×320 pixels Quantum Well Infrared Photodetector (QWIP detector), operating in the infrared wavelength region of 8-9 μm and cooled to 70°K to reduce thermal noise. The camera provided a resolution of 0.02°C and a maximum frame rate of 60 Hz. It was equipped with the default lens (FOV = 20°×15°), and was installed 1.5 m away from the model with an optical axis oriented in the horizontal plane at about 30° with respect to the wing surface mid-chord normal. Optical access was provided through an opening on the side wall of the test section opposite to the upper surface. More details about the methodology and processing are available in ref. (Mébarki, 2009).

2.2 Aerodynamic Detection System and Graphical User Interface

The morphing wing goal was the improvement of the laminar flow over the upper surface of the wing. In order to ensure that the improvement was real, we built a detection system that gave information about the flow characteristics. An array of twelve Kulite pressure sensors was installed on the flexible skin.

The pressure data acquisition was performed using a NI-DAQ USB 6210 card with 16 analog inputs, at a total sampling rate of 250 kilo samples/s. The input channels were connected directly to the IAR-NRC analog data acquisition system which was connected to the twelve Kulite sensors. The IAR-NRC served as an amplifier and conditioner of the signal at a sampling rate of 15 kilo samples/s. One extra channel was used for the wind tunnel dynamic pressure acquisition to calculate the pressure coefficients C_p 's from the pressure values measured by the twelve pressure sensors. The signal was acquired at sampling rate of 10 kilo samples/s in frames of 1024 points for each channel which allowed a boundary layer pressure fluctuations fast Fourier transform (FFT) spectral decomposition up to 5 kHz for all channels, at a rate of 9.77 samples/s using Matlab/Simulink software. The plot results were visualized in real time on the computer screen in dedicated windows (see Figure 4) at a rate of 1 sample/sec. Figure 4 shows an example of graphical user interface in which all the aerodynamic and morphing shape information were centralized together with the control buttons of the controlling

software. The window shows information about the Mach number, the angle of attack, the airfoil shape of the morphing wing, and the two actuators vertical displacements needed to obtain the desired airfoil shape. In the two plots, are shown the pressure coefficients distribution C_p 's of the twelve Kulite sensors, and the noise of the signal (RMS) of each pressure signal. Figure 4.a shows the wing un-morphed position, while the Figure 4.b shows the wing under its morphed position. The results obtained are qualitatively very similar to those obtained in previous studies (Nitsche, 1989 and Mangalam, 2004).

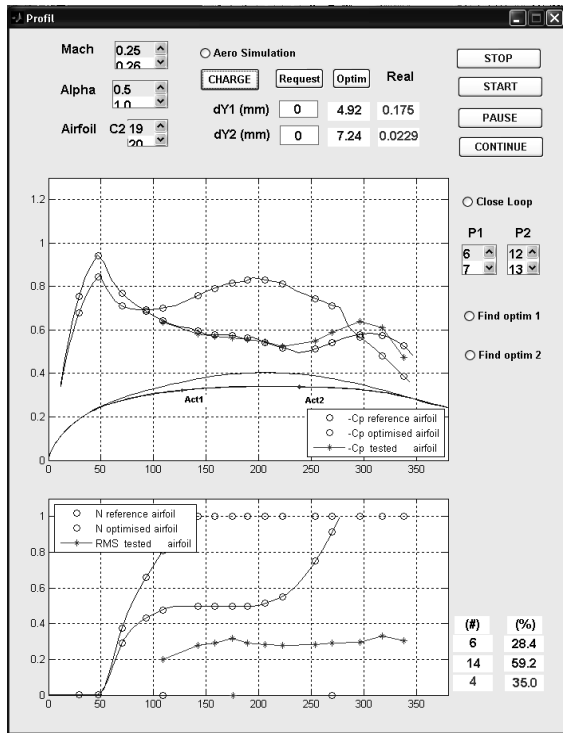


Figure 4.a: Un-morphed configuration.

The transition between laminar and turbulent flow was detected by means of each pressure signal's root mean square (RMS). The lower RMS plot given in Figure 4 shows the normalized quantity of the pressure signal noise from each Kulite sensor (star points curve). In the example shown in Figure 4, the RMS plot in the un-morphed configuration (Figure 4.a) the transition is shown in the fourth sensor due to the fact that it had the maximum RMS value.

In Figure 4.a, on the GUI was shown an un-morphed airfoil by use of a black color. The actuators reference positions correspond to $dY1 = 0$ mm and $dY2 = 0$ mm, the C_p distribution calculated by XFOIL for the reference airfoil (black curve), and

the C_p theoretical values of the sensors shown as black circles on the C_p distribution curve.

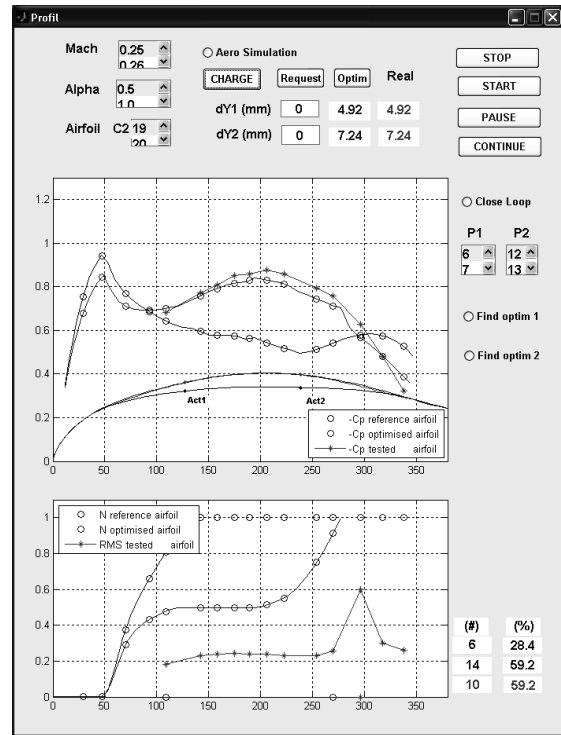


Figure 4.b: Morphed configuration. Graphical User Interface (GUI) with the control buttons of the software.

In the lower plot of Figure 4.a was shown the N factor used by XFOIL to predict transition for the reference airfoil (black curve). The critical value $N_{cr} = 7.34$ was used in the simulation to match the turbulence level $T = 0.14\%$ measured in the wind tunnel using Mack's correlation (1), and the plotted values on the figure are normalized (N/N_{cr}) (Mack, 1977).

$$N_{cr} = -8.43 - 2.4 \cdot \log(T) \quad (1)$$

In the case of an un-morphed configuration, the predicted transition position was found to be the 6th position of the sixteen available sensors positions. In the beginning of wind-tunnel tests, a number of sixteen sensors were installed, but due to their removal and re-installation during the next two wind tunnel tests, four of them were found defective, therefore a number of twelve sensors remained to be used during the last third wind tunnel tests so only twelve Kulite sensors were used for plotting the C_p distribution and RMS distribution (star plots).

Results predicted for the morphed airfoil were shown in circles. The morphed airfoil coordinates were shown as blue curves in the upper part of

Figure 4.b, the C_p distribution was calculated by XFOil for the optimized airfoil (circles curve), and the C_p theoretical values of the sensors were shown as blue circles on the C_p distribution curve. In the lower plot of Figure 4.b, the N factor used by XFOil to predict transition was shown for the optimized airfoil (circles curve). In this case of morphed configuration, the predicted position of transition was the 14th position of the sixteen available sensors positions.

These lower circles (un-morphed) and upper circles (morphed) curves served as theoretical validations of the measured value curves reflecting the aerodynamic parameters (C_p and RMS) provided by Kulite sensors in real time with a sampling rate of 1 S/sec. In Figure 4.b is shown the actuated airfoil in the morphed position ($dY1 = 4.92$ mm and $dY2 = 7.24$ mm). The transition position was given by the sensor location where the maximum RMS was found, which in this case is the 10th Kulite sensor out of 12 sensors. The instant visualization allows us to find the exact position predicted by XFOil.

2.3 Simulation and Experimental Results Obtained in the Wind Tunnel

The simulation of the system used as programming platform the Matlab/Simulink software. The simulation used the optimization subroutine exactly the same as in bench tests and wind tunnel tests, except that in computer simulation and bench test the aerodynamic pressures that acted upon the skin and which stimulated the sensors were simulated by use of XFOil software. As mathematical model of the flexible skin was used a B-spline with four flexion points. Two points were fixed where the skin is glued on the wing rigid structure and two points were mobile and were placed in the actuators coordinates on the wing structure. The B-spline shape that define the airfoil's flexible skin did not have the same coordinates as the flexible skin but was a good approximation for the purpose of designing an optimization subroutine in closed loop with a CFD code. Laser scanning during bench tests showed that the differences between the scanned airfoils and the theoretical airfoils were less than 0.5 mm (less than 6.25% of the maximum actuators deflection of 8 mm) (Popov, 2009). The optimization initialized the algorithm with the values $dY1 = 4$ mm and $dY2 = 4$ mm. Afterwards the algorithm evaluated the transition point position in eight points of coordinates ($dY1$, $dY2$) situated on a circle centered in the initial point with a radius of 4

mm within the search domain defined by the bi-dimensional space of actuators strokes $\{dY1 = [0, 8], dY2 = [0, 8]\}$. For each evaluation point, the x_{tr} value was evaluated by use of XFOil and stored in the memory. After the first round of evaluations the optimizer decided which evaluating point had the maximum value of x_{tr} , which will become the initial point for the next round of evaluations. The logic schematic of the optimization subroutine is shown in Figure 5.

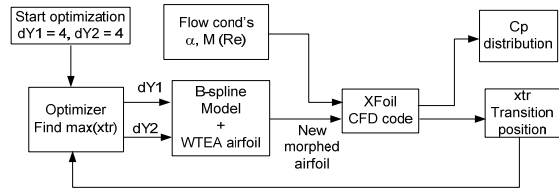


Figure 5: Optimization logic schematic.

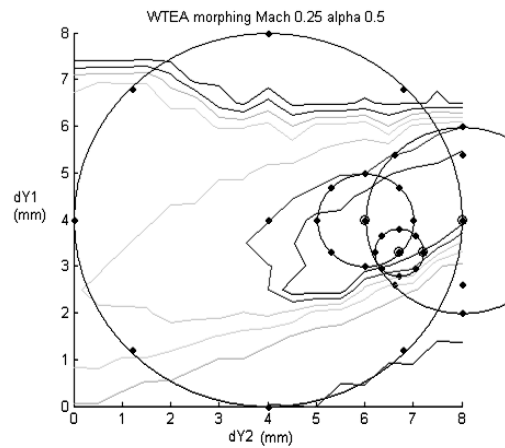


Figure 6: Optimization in simulation using XFOil code for the airflow condition $M = 0.25$ and $\alpha = 0.5^\circ$.

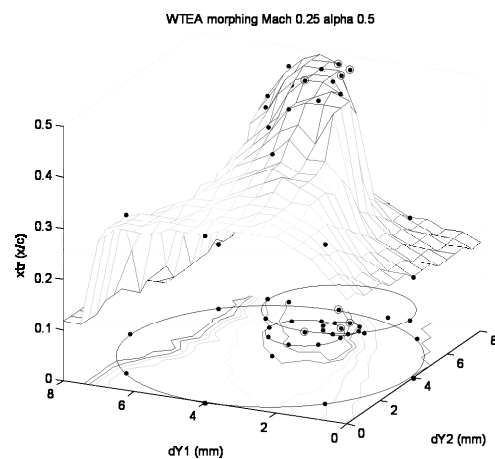


Figure 7: Optimization in simulation using XFOil code for the airflow condition $M = 0.25$ and $\alpha = 0.5^\circ$.

Figures 6, 7 and 10 show the result of WTEA airfoil optimization after four evaluation rounds, first evaluation with a radius of 4 mm, second evaluation with a radius of 2 mm, third evaluation with a radius of 1 mm and fourth and last evaluation with a radius of 0.5 mm. As seen in Figure 7 the last round of evaluation was almost unnecessary because the maximum x_{tr} was found inside a plateau of maximums with very small differences between them. Before doing the optimization it was performed a mapping of the search domain, i.e. for each combination of $dY1$ and $dY2$ in the interval [0 mm, 8 mm] with a step of 1 mm it was found the x_{tr} and was built the surface $x_{tr} = f(dY1, dY2)$ for the purpose of visualizing the form of the ‘hill’ and to validate the algorithm in the simulation. Figure 8 and 9 show the same optimization routine that run during the wind tunnel tests in the same airflow conditions as the ones simulated except that there was no map of the searched function. The result was slightly different because the airfoil shape of the real flexible skin under wind tunnel conditions was different than the airfoil shapes defined by use of B-splines. Still the result was similar, in terms of actuator strokes $dY1$ and $dY2$ as well as the position of transition. Similarly there could be observed in Figure 9 a plateau of evaluation points that had the transition occurrence on the 11th sensor.

Figure 10 shows the result of the airfoil shape optimization, C_p distribution and x_{tr} transition point position on the upper surface of the airfoil obtained through simulation using XFOil and B-splines model for the flexible skin. The values obtained for wind flow conditions of Mach = 0.25 and $\alpha = 0.5$ are $dY1 = 3.3$ mm and $dY2 = 7.2$ mm. Also in Figure 7 is shown the N factor distribution which was the parameter used by XFOil to calculate the transition point position. When N factor reached the N_{cr} critical value the transition was triggered. This parameter was used in wind tunnel to validate the transition position found through the RMS measuring of the Kulite pressure sensors.

Figure 11 shows the optimized airfoil shape, C_p distribution and x_{tr} transition point position on the upper surface of the airfoil in wind tunnel test (star plots) compared to the optimal airfoil plots (upper circles) and reference airfoil plots (lower circles) obtained through simulation. Also in the lower subplot of Figure 11 the N factor used by XFOil to detect the transition position was compared to the RMS of the Kulite sensors. Both the N factor and RMS were normalized and the purpose of the plots was to have a visual indicator of the transition position. The software considered the transition

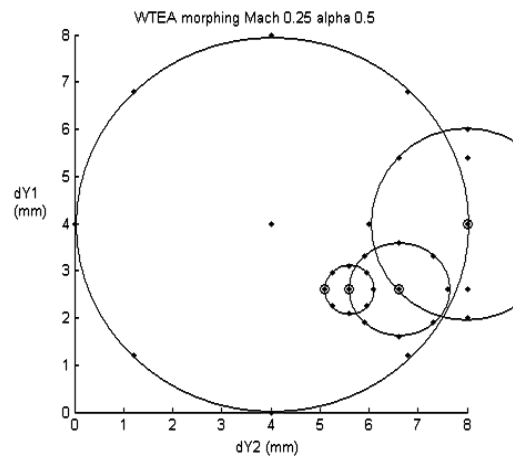


Figure 8: Optimization in real time during wind tunnel tests for the airflow condition $M = 0.25$ and $\alpha = 0.5^\circ$.

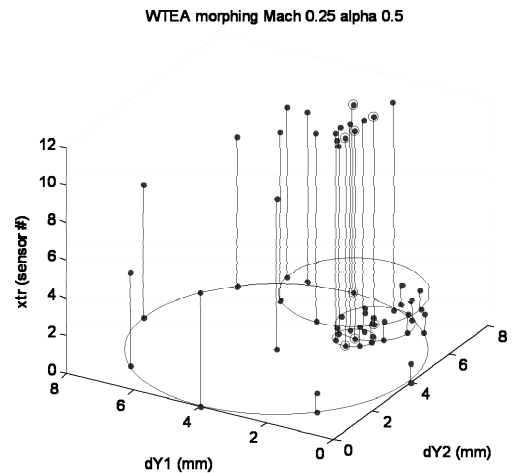


Figure 9: Optimization in real time during wind tunnel tests for the airflow conditions $M = 0.25$ and $\alpha = 0.5^\circ$.

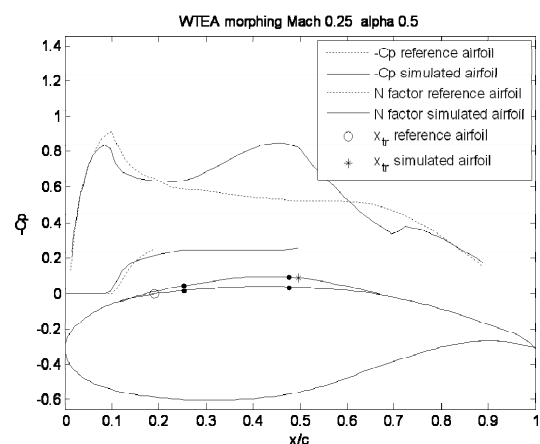


Figure 10: Optimization simulation result for $M = 0.25$ and $\alpha = 0.5^\circ$.

position in the coordinates of the sensor with the highest noise (RMS) as confirmed by previous studies (Rioual, 1994). The values obtained in wind tunnel for wind flow conditions of Mach = 0.25 and $\alpha = 0.5$ are $x_{tr}/c = 0.635$ ($x_{tr} = 317.5$ mm) for the actuator displacement values $dY1 = 2.6$ mm and $dY2 = 5.1$ mm.

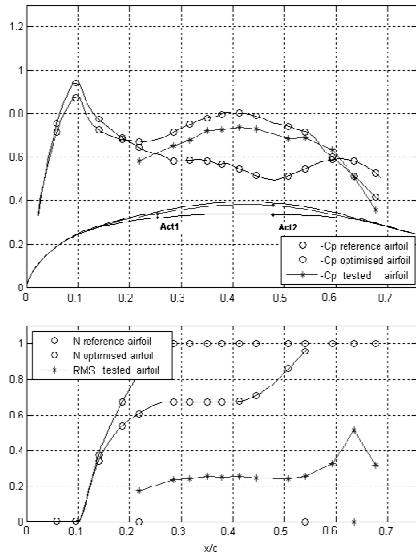


Figure 11: Optimization result during wind tunnel test for $M = 0.25$ and $\alpha = 0.5^\circ$.

Figure 12 shows the time history of the optimization process in wind tunnel. Due to the long response of the SMA actuators – the time of cooling from maximum displacement to zero was approx 2 min – the entire process of optimum search converged to the optimum values in approx 20 min. Also, there can be observed that the requested displacements of the actuators at the maximum displacement of 8 mm were not realized, due to the fatigue of the SMA’s accumulated in previous testes. The maximum deflection was in fact 7 mm for the first actuator and 6.5 mm for second actuator.

Figure 13 shows typical infrared results obtained at $M = 0.25$, $\alpha = 0.5^\circ$ for various configurations. Only the composite portion of the wing at $x/c \leq 0.7$ was shown. The white spots on the wing are the electronically heated Kulite pressure transducers. The two lines of SMA actuators, colder than the model surface, were also visible at quarter chord and near mid-chord. The locations of the transition in the images have been highlighted using a white dashed line: it corresponded to the location of a large surface temperature gradient, the laminar region being about 2-3°C hotter than the turbulent region.

The reference airfoil configuration (Figure 13)

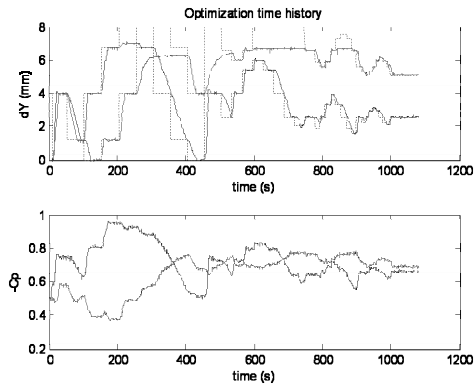


Figure 12: Optimization time history during wind tunnel test for $M = 0.25$ and $\alpha = 0.5^\circ$.

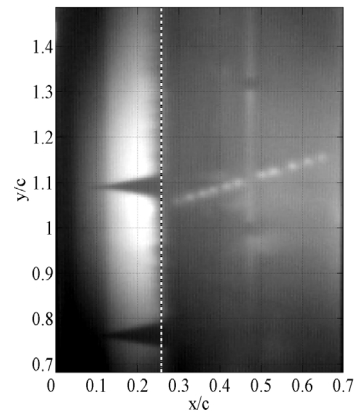


Figure 13: Infrared results obtained at $M = 0.25$ and $\alpha = 0.5^\circ$ in reference position.

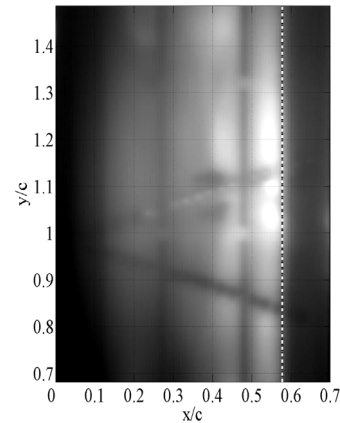


Figure 14: Infrared results obtained at $M = 0.25$ and $\alpha = 0.5^\circ$ after optimization.

showed a transition location at $x/c = 26\%$. The optimization (Figure 14) allowed a laminar boundary layer run to $x/c = 58\%$, which represented a significant improvement over the reference case (Figure 13). Some turbulent wedges caused by

leading edge contamination, due to dust particles in the flow, were visible in Figure 10-a. In addition to providing an on line verification of the Kulite dynamic pressure signals, the infrared measurement was particularly useful to detect those early artificial turbulent regions.

3 CONCLUSIONS

The results of the tests performed in wind tunnel using a morphing wing were shown. The optimization method did not use any CFD code but used the same optimization algorithm in real time. This optimization converged in approximately 10 minutes due to the slow response of the SMA actuators especially in the cooling phase of the cycle. It was observed that the airfoil realized by this method slightly differs from the optimization using CFD codes. This result was due to the fact that the cost function of the optimization (transition position) had discrete values (the sensors positions) and the maximum of the function was a plateau of different dY1 and dY2 values. The optimizer stopped at a certain value in function of the number and magnitudes of the searching steps. It was observed that the last searching step (searching of the maximum in eight points situated on a circle with ray of 0.5 mm – see Figure 9) was not necessary due to the cost function plateau of maximums.

ACKNOWLEDGEMENTS

The authors would like to thank the Consortium of Research in the Aerospace Industry in Quebec (CRIAQ) for funding the present work, and Thales Avionics and Bombardier Aerospace for their financial and technical. The authors would like also to thank George Henri Simon for initiating CRIAQ 7.1 project and Philippe Molaret from Thales Avionics for their collaboration on this work.

REFERENCES

- Allison, D. O., and Dagenhart, J. R., 1978, Design of a Laminar-Flow-Control Supercritical Airfoil for a Swept Wing, CTOL Transport Technology, NASA Langley Research Center, pp. 395–408.
- Andersen, G. R. et al., 2007, Aeroelastic modeling, analysis and testing of a morphing wing structure, AIAA-2007-1734, pp. 359-373.
- Bharti, S. et al., 2006, Optimal structural design of a morphing aircraft wing using parallel non-dominated sorting genetic algorithm II (NSGA II), *Smart Structures and Materials 2006: Industrial and Commercial Applications of Smart Structures Technologies*, Proceedings of SPIE Vol. 6166, pp. 1-12.
- Coutu, D., Brailovski, V., Terriault, P., 2009, Promising benefits of an active-extradors morphing laminar wing, *AIAA Journal of Aircraft*, Vol. 46(2), pp. 730-731.
- De Breuker, R. et al., 2007, Energy-based aeroelastic analysis of a morphing wing, *Proceedings of SPIE*, Vol. 6523, pp. 1-12.
- Gandhi, N. et al., 2007, Intelligent control of a morphing aircraft, Paper AIAA-2007-1716, pp. 166-182.
- Georges, T., Brailovski, V., Morellon, E., Coutu, D., and Terriault, P., 2009, Design of Shape Memory Alloy Actuators for Morphing Laminar Wing With Flexible Extradors, *Journal of Mechanical Design*, Vol. 131, No. 9, pp. 091006-1–091006-9.
- Hackenberg, Petra, 1995, Numerical optimization of the suction distribution for laminar flow control aerofoils, Doctoral Thesis, University of Southampton (United Kingdom).
- Hill climbing, From Wikipedia, the free encyclopedia [online], http://en.wikipedia.org/wiki/Hill_climbing
- Jacob, J.D., 1998, On the Fluid Dynamics of Adaptive Airfoils, *Proceedings of ASME International Mechanical Engineering Congress and Exposition* November 15-20, 1998, Anaheim, CA, USA.
- Jacob, J. D., 1999, Aerodynamic Flow Control Using Shape Adaptive Surfaces, ASME Paper No. DETC99/VIB-8323, *ASME 17th Biennial Conference on Mechanical Vibration and Noise, Symposium on Structronics, Mechatronics, and Smart Materials*, Las Vegas, Nevada, September.
- Khalid, M., 1993, Navier Stokes Investigation of Blunt Trailing Edge Airfoils using O-Grids, *AIAA Journal of Aircraft*, Vol.30, No.3, pp.797-800
- Khalid, M., and Jones, D. J., 1993, ACFD Investigation of the Blunt Trailing Edge Airfoils in Transonic Flow, *Proceedings of the Inaugural Conference of the CFD Society of Canada*, Montreal, June14-15.
- Love, M. H. et al., 2007, Demonstration of morphing technology through ground and wind tunnel tests, Paper AIAA-2007-1729, pp. 337-348.
- Mack, L. M., 1977, Transition and Laminar Instability, *Jet Propulsion Laboratory Publication 77-15*, Pasadena, CA.
- Mangalam, S. M., 2004, Real-Time Extraction of Hydrodynamic Flow Characteristics Using Surface Signature, *IEEE Journal of Oceanic Engineering*, Vol. 29, No. 3, pp. 622-630.
- Mébariki, Y., Mamou, M. and Genest, M., 2009, *Infrared Measurements of Transition Location on the CRIAQ project Morphing Wing Model*, NRC LTR- AL-2009-0075.
- Munday, D., Jacob, J. D., and Huang, G., 2002, Active Flow Control of Separation on a Wing with Oscillatory Camber, AIAA Paper No. 2002-0413, *40th AIAA Aerospace Sciences Meeting*, Reno, NV.

- Munday, D., Jacob, J. D., T. Hauser, and Huang, G., 2002, Experimental and Numerical Investigation of Aerodynamic Flow Control Using Oscillating Adaptive Surfaces, AIAA Paper No. 2002-2837, *1st AIAA Flow Control Conference*, St. Louis.
- Nitsche, W., Mirow, P., and Dorfler, T., 1989, Investigations on Flow Instabilities on Airfoils by Means of Piezofoil Arrays, *Laminar-Turbulent Transition Proceedings of the IUTAM Symposium*, Ecole nationale Supérieure de l'Aéronautique et de l'Espace, Toulouse, France, Sept. 11-15, Berlin and New York, Springer-Verlag, 1990.
- Popov, A-V., Labib, M., Fays, J., Botez, R.M., 2008, Closed loop control simulations on a morphing laminar airfoil using shape memory alloys actuators, *AIAA Journal of Aircraft*, Vol. 45(5), pp. 1794-1803.
- Popov, A-V., Botez, R. M., and Grigorie, L., 2009, Morphing Wing Validation during Bench Tests, *Canadian Aeronautics and Space Institute Annual General Meeting, Aircraft Design & Development Symposium*, 5-7 May, Kanata, Ontario.
- Pralits, J., 2003, Optimal Design of Natural and Hybrid Laminar Flow Control on Wings, Doctoral Thesis, Technical Report from Royal Institute of Technology, Stockholm, Sweden.
- Rioual, J.-L., Nelson, P. A., and Fisher, M. J., 1994, Experiments on the Automatic Control of Boundary-Layer Transition, *Journal of Aircraft*. Vol. 31, No. 6, pp. 1416-1418.
- Sainmont, C., Paraschivoiu, I., Coutu, D., 2009, Multidisciplinary Approach for the Optimization of a Laminar Airfoil Equipped with a Morphing Upper Surface, *NATO AVT-168 Symposium on "Morphing Vehicles"*, Evora, Portugal.
- Sanders, B. et al., 2003, Aerodynamic and aeroelastic characteristics of wings with conformal control surfaces for morphing aircraft, *Journal of Aircraft*, Vol. 40(1), pp. 94-99.
- Simulated annealing, From Wikipedia, the free encyclopedia [online], http://en.wikipedia.org/wiki/Simulated_annealing
- Skillen, M. D., Crossley, W. A., 2005, Developing response surface based wing weight equations for conceptual morphing aircraft sizing, Paper AIAA-2005-1960, pp. 2007-2019.
- Yang, S.-M., Han, J.-H., Lee, I., 2006, Characteristics of smart composite wing with SMA's and optical fibre sensors, *International Journal of Applied Electromagnetics and Mechanics*, Vol. 23, pp. 177-186.
- Zingg, D. W., Diosady, L., Billing, L., 2006, Adaptive Airfoils for Drag Reduction at Transonic Speeds, AIAA paper 2006-3656.

APPENDIX

- b = span of wing model (m)
 c = chord of wing airfoil (m)
 C_D = drag coefficient
 C_L = lift coefficient
 C_p = pressure coefficient
 M = Mach number
 N = natural logarithm of rapport between amplified perturbation and initial perturbation in laminar flow
 Re = Reynolds number
 x_{tr} = transition position (m)
 α = angle of attack of the wing ($^\circ$)

GENETIC ALGORITHM VERSUS ANT COLONY OPTIMIZATION ALGORITHM

Comparison of Performances in Robot Path Planning Application

Nohaidda Binti Sariff and Norlida Buniyamin

Faculty of Electrical Engineering, University Technology Mara, Shah Alam, Selangor Darul Ehsan, Malaysia
{norhaidasariff, nbuniyamin}@salam.uitm.edu.my

Keywords: Path Planning Algorithm, Mobile Robot Path Planning, Genetic Algorithm, Ant Colony Optimization.

Abstract: This paper presents the results of a research that uses a simulation approach to compare the effectiveness and efficiency of two path planning algorithms. Genetic Algorithm (GA) and Ant Colony Optimization (ACO) Algorithm for Robot Path Planning (RPP) were tested in a global static environment. Both algorithms were applied within a global map that provides feasible nodes from start point to goal. Performances between both algorithms were compared and evaluated in terms of computational efficiency by measuring the speed and number of iterations, accuracy of solution, solution variation and convergence behavior.

1 INTRODUCTION

Path planning (PP) research covers a wide area of robotics research that includes PP in static (Charles.W.Warren, 1993) (Xin, 2005) and dynamic environments (Mei, 2006) (Stentz, 1994). By assuming a robot has knowledge of the environment before it moves, the application of a model based approach to solve RPP problem in a global static environment was used in this research.

Examples of traditional approaches proposed by previous researchers to solve RPP problems are artificial potential field (Khatib, 1985), neural network (Xin, 2005), distance wave transform (Zelinsky, Oct 1993), heuristic algorithm known as A* algorithm (Charles.W.Warren, 1993) (Hart et al., 1968), and D* algorithm (Yahja, 2000). It has been proven in various researches that these algorithms were able to find global path successfully and that the various methods has its own strengths and limitations over others in certain aspect of path planning.

Recently, due to the evolution of PP algorithms (PPAs), researchers are viewing RPP problem as an optimization problem (Sariff, June 2006). This newer method focuses on finding an optimal path from start to destination point while satisfying the optimization criteria for the robot path, such as a short path with small computation time. In order to

solve the PP problem, the applications of artificial technologies (Netnevitsky, 2002) itself have been expanded by utilizing approaches such as Evolutionary Computation; Genetic Algorithm (N.Sivanandam, 2008) (Nagib, 2004) (Tu, 2003) (Ramakrishnan, 2001) and Swarm Intelligence; Ant Colony Optimization (Dorigo, 2004) (Dorigo and Gambardella, 1997) (Gengqian et al., 2005) in RPP research areas. Compared to the traditional approaches, this method provides robust and effective search techniques for optimization purposes which were widely used to solve the RPP problem.

Since its appearance in 1975 (Goldberg, 1994), GA has been used in solving many RPP optimization problems. GA is a search technique inspired by biology where it works based on the principle of the fittest of the chromosomes. With its ability to work with parallel search techniques, the use of GA contributed to the success of many RPP research. For example, (Nagib, 2004) proposed the use of GA to find robot path based on a map of free space nodes. (Sugihara, 1997) and (Ramakrishnan, 2001) also proposed the used of GA with different encoding techniques to ensure GA can find optimal path without depending on the feasible nodes given in the map. (Hu, 2004) modified classical GA by incorporating the domain knowledge into specialized operator to improve GA performances when it works in environments that consists of obstacles. Previous

research indicates that GA can be used to solve RPP in different applications and that the GA process to find the optimal path is affected by the representation of the solution, fitness function evaluation and genetic operators selection.

ACO, compared to GA is a newer optimization method. Introduced by (Dorigo, 2004) in approximately 1992, the application of this algorithm in RPP research increased rapidly as it is a powerful tool for solving hard combinatorial optimizations problem. ACO was inspired by analogy of behavior of real ants, when looking for foods. (Zheng, 2007) proposed the use of ACO to find robot path based on map of MAKLINK graph. (Mei, 2006) combined ACO with Artificial Potential Field to produced the path planning in dynamic environment. (Gengqian et al., 2005) have proven that ACO can find optimal path in their grid map by proposing its own probability equation. However, a literature study shows that the application of ACO to solve RPP problems has not been explored in detail.

The purpose of the research presented herewith is to examine the performances of ACO and GA in a given map (Sariff, 2009). The performances of both algorithm will be evaluated and compared in terms of computational efficiency, accuracy of solution, solution variation and convergence behavior. The goal is to enhance knowledge of optimization algorithms in RPP research area. In this paper, the mapping and path planning algorithms construction is first discussed. Then results and discussions provided. Finally, a conclusion that compares and summarizes the performances of ACO and GA is presented.

2 RESEARCH METHODOLOGY

Figure 1 illustrates the method applied within this research. The robot environment must initially be mapped using an appropriate global map (described in section 2.1 below). This map will create an output of nodes represented by xy coordinates. Then, GA and ACO will start to initialize the population of path using its own approaches from start to goal by using all the provided nodes including the start, goal and all intermediate nodes.

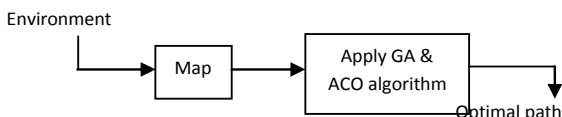


Figure 1: Proposed Method.

During the initialization, the integer number represented by each node will be used. However, during the evaluation, the real x-y coordinates will be used. At the end of the process, the optimal path will be found.

2.1 Environment Modeling

In this research, a 2D grid map with size 10x10 cm was used where the free space nodes (white cell) represents the area the robot can traverse including the robot size. The obstacles area (black cell) represents the boundary of obstacles with the safety region and the yellow grid represents the feasible free space nodes that can be traversed by the robot as shown in Figure 2. The feasible free space nodes have been located and routed randomly within this grid map by assuming the nodes are the free space nodes extract from the mapping algorithm itself. By using this map, the algorithm will start finding a solution by initializing the population of feasible path to goal based on the feasible nodes or unfeasible nodes (need to be added) available as shown in Figure 3.

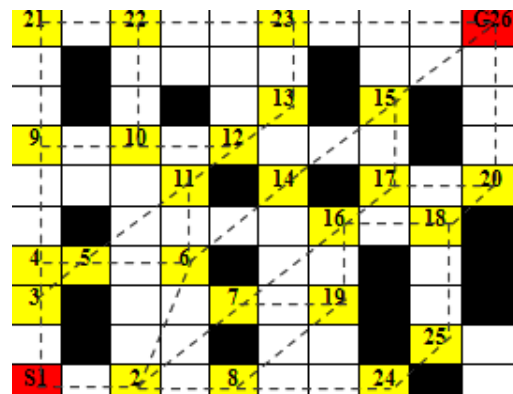


Figure 2: Global feasible map with 26 free space feasible nodes.

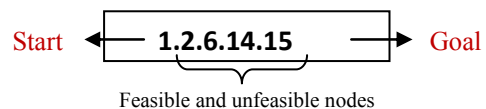


Figure 3: A sample of population consists of feasible nodes of Figure 2.

2.2 Genetic Algorithm Design for RPP

The outline of GA is given in Figure 4. The initial solutions of the RPP problem will initialize in population randomly. In the first case, the population will initialize based on the feasible nodes provided in the global map only. With the complete

population, the fitness is evaluated by using the formula below:

$$\text{Fitness node} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

$$\text{Total Fitness} = \begin{cases} \sum \text{Fitness node} & ; \text{Feasible} \\ 100 & ; \text{Unfeasible} \end{cases} \quad (2)$$

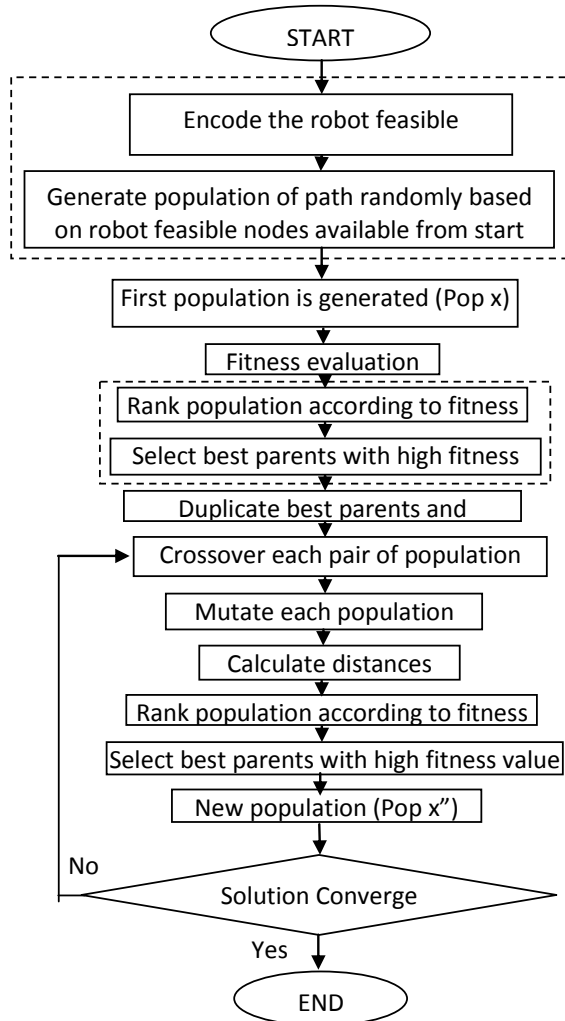


Figure 4: Outline of GA for RPP of a mobile robot.

After the fitness of each population has been evaluated, it will be ranked using an elitism approach. The shorter path will be represented with a high fitness value and will be selected to be carried forward to the next generation while the long path represented with a low fitness value will be eliminated and removed from the population. The good parents which is carried forward to the next generation will produce the diversity of population that consists of a good child from the genetic operators process. Then this process is repeated until

all of the GA population found the same optimal path with no difference of the fitness value where the distance is equal to 0. It is at this moment, that the solution converges. The type of GA and important parameters specifications related with GA used in this experimental research is defined in Table1 below:

Table 1: GA Parameter Specifications.

GA properties	Properties
Type of GA	Classical GA
Chromosomes type	Fixed length chromosomes
Population Size	50
Chromosomes length	15
Selection type	Elitism
Crossover type	Two point crossover
Mutation type	Flip bit
Crossover rate	0.75 of the population size
Mutation rate	0.75 of the population size
Convergence Criteria	Max-min of 20 pop ≤ 0.001
Maximum Iteration	40

2.3 Ant Colony Optimization Design for RPP

ACO algorithm used in this experiment is the Ant System (AS) algorithm as proposed by (Dorigo, 2004). However, a new heuristic equation of state transition rules is proposed for the RPP purposes. The evaluation fitness and ACO parameter setting was created based on the requirements of this research.

The design of AS for RPP was divided into three important rules which are state transition rules, local update rules and global update rules. At the beginning, ants will determine the next node to be visited by using the state transition rules based on heuristic and pheromone laid down by the ants as shown in derivation below:

$$\text{Probability } ij = \text{heuristic} * \text{pheromone} \quad (3)$$

$$= \left[\frac{1}{\text{distance between vector start to subpath and start to perpendicular subpath with reference goal}} \right]^\beta * (\text{trail} / \sum \text{trail})^\alpha$$

*β=heuristic coefficient, α=pheromone trail coefficient

An accurate value of distance by heuristic equation and the higher amount of pheromone of the visited node will be obtained by the ants that have higher probability to choose that nodes. Within these rules, ants can balance between the exploration and exploitation from the relatives coefficient provided, known as alpha and beta. During the construction of

the path, the pheromone will be reduced locally by the given evaporation rate by using the formula of update local rules below:

$$T_{ij} \text{ (new trail)} \leftarrow (1-\rho) * t_{ij} \text{ (old trail)}, \quad (4)$$

* ρ = evaporation rate

After all the ants complete the path to goal, then the process of global updating is applied where ants will deposits its pheromone based on the path distance.

$$t_{ij} \leftarrow t_{ij} + \sum \Delta t_{ij}^k \quad (5)$$

Δt_{ij}^k = amount pheromone of ant m deposits on the path it has visited. It's defined as below:

$$\Delta t_{ij}^k = \begin{cases} Q/C^k & ;\text{if arc } (i,j) \text{ belongs to path } P^k \\ 0 & ;\text{otherwise} \end{cases} \quad (6)$$

where Q is number of nodes and C^k is the length of path P^k built by the ants.

The amount of pheromone will continuously be updated until it attracts more ants from the next generation to follow the shorter path. Finally, the optimal robot path is found by using behavior of ants' concept as shown in Figure 5 below.

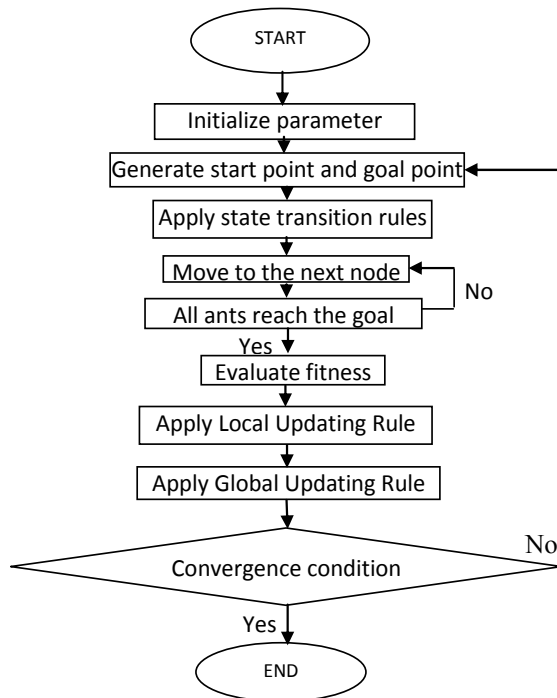


Figure 5: Outline of ACO for RPP of a mobile robot.

The parameter specifications of ACO utilized in this experiment is shown in Table 2.

Table 2: ACO Parameter Specifications.

ACO Properties	Properties
Population of ants	50 (same as GA)
Length of ants junction	15 (same as GA)
Pheromone coefficient, β	5
Heuristic coefficient, α	5
Evaporation rate, ρ	0.5
Convergence condition	Max-min of 20 pop ≤ 0.001 (same as GA)
Maximum Iteration	40 (same as GA)

2.4 Experiment

The method described is then translated and coded into MATLAB source code by using an appropriate function available within MATLAB 7.0.4. The simulation was carried out using a computer with Intel (R) Celeron (R) M processor 1.5 GHz with 504MB of RAM. Various Simulation results were then recorded based on the evaluation criteria required for experiment outcomes such as optimal path, path cost, time, number of iterations, etc.

3 RESULTS & DISCUSSIONS

3.1 Comparison of GA and ACO Computational Efficiency

The computational efficiency of both algorithms was measured by observing the computation time and number of iteration found by algorithms in 5 test runs. The optimal path found by both algorithms is a path with connection of feasible nodes 1.2.6.14.15.26 as shown in Figure 6 below with the path cost that is equal to 13.648 cm. The average time and iteration value is illustrated in Tables 3 and 4 below while Figures 7 and 8 below have been proposed to differentiate the values between both algorithms in each run time.

Based on results tabulated in Tables 3 and 4 below, the average time required by ACO to find the optimal path (in 5 test runs) is smaller compared to GA which shows that ACO can perform faster than GA. The computation time found by ACO in each run time is mostly less than 100 seconds while GA run times are in general more than 100 seconds with the highest run of more than 300 seconds. One of the factors that influence the increment of time and iteration is the population being initialized. The way ACO initializes the population by using a state transition rules is more efficient compared to GA that is based on random approaches. With the efficient derivation of state transition rules, ants

capable to determine the next node to be visited near the optimal node which will produce the population of ants that traverse near the optimal path to goal. During this process, ants will choose the nodes with high probability value (near the optimal node) and abandon the nodes with low probability value (far from optimal node). The effect of this process is that the number of optimal path from one generation to the next generation will increased rapidly and will simultaneously drive ACO to converge faster than GA.

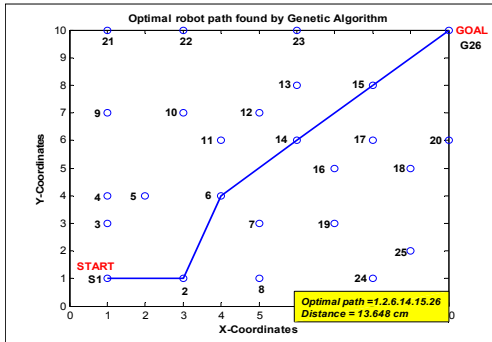


Figure 6: Optimal path found by GA in 1st test run.

However, with the GA, there is no rules to determine the optimal node as GA uses the random based approach. This will cause the number of optimal path in each generation increase in constantly because population of optimal path is keep changing depends on the random process itself. Therefore as results, GA needs more time and iteration compared to ACO in order to face with the difficulties of this random approach.

In addition, the way both algorithms carry forward the optimal path from one generation to the next generations will also influence the time and iteration the algorithms require to converge. ACO will carry forward the updated pheromone values each time it return back to the start point after it reach the goal point. The pheromone value carried by the ants is depending on the selected path traverse by the ants itself where this value will guide the next ants to choose the path for the next generations. Effect from the efficient local and global updating process, ACO shows the rapid increment of optimal path population in each generation which will drive ACO to converge faster compared to GA.

Vice versa with ACO, GA will select the good population (good parent) which have the highest fitness to be carry forward to the next generation. After that, this population will be duplicated and it will go through some of the process known as crossover and mutation to produce the next child. However, because the process to cross and mutate

will also determine randomly so the chances to get a good child from a good parent also become difficult and inconstant. Therefore, the increment of optimal path population in each generation is also not rapidly increase like ACO where it need more time and more generations to find the optimal path to goal. This has been proven in results illustrated in Tables 3 and 4 below where the average time and iteration in five test run times for ACO is smaller compared to GA.

Table 3: Computation Time & Iteration of GA.

Number of run	Optimal path	Distance (cm)	Time(sec)	Iteration
1	1.2.6.14.15.26	13.648	111.838	10
2	1.2.6.14.15.26	13.648	147.958	7
3	1.2.6.14.15.26	13.648	114.362	8
4	1.2.6.14.15.26	13.648	310.464	7
5	1.2.6.14.15.26	13.648	101.278	8
Avg Total		13.648	157.18	8

Table 4: Computation Time & Iteration of ACO.

Number of run	Optimal path	Distance (cm)	Time(sec)	Iteration
1	1.2.6.14.15.26	13.648	104.606	4
2	1.2.6.14.15.26	13.648	44.4	4
3	1.2.6.14.15.26	13.648	73.552	6
4	1.2.6.14.15.26	13.648	43.635	4
5	1.2.6.14.15.26	13.648	49.297	4
Avg Total		13.648	63.098	4.4

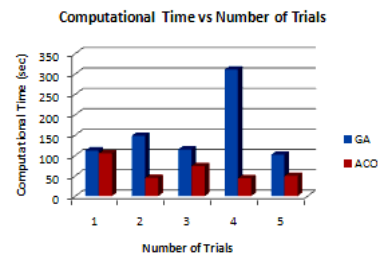


Figure 7: GA and ACO computation time.

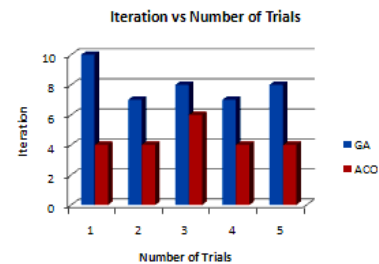


Figure 8: GA and ACO iteration.

3.2 Comparison of GA and ACO Accuracy of Solutions

The accuracy of the solution provided by both algorithms in finding the optimized path can be

measured from the quality of the path found in each test run times. The path is optimal if the path is complete, feasible (not obstruct by obstacles, from start to finish point), shorter and require small computation time. For this comparison purposes, the path based on results is tabulated in Table 5 below.

As depicted, in 5 test runs, ACO could generate 100% of optimal path in 5 test runs while only 60% of optimal path was generate by GA in 5 test runs. This shows that ACO can work effectively because the optimal path found each time the solution converge. However it was different with GA where affect from the random process, it will somehow cause GA to converge although it still in premature solution. Thus will cause the path being produced is not optimal, not feasible and not complete to goal as example shown in the 2nd and 3rd test run in Table 5 below. Although the path cost is less compare to other run time, it was still not considered as an optimal path because the path is not feasible and not complete to goal. As a result, there are only 3 test runs among 5 test runs that GA can obtained optimal path which is equal to 13.648 cm distance.

Table 5: Optimal Path Found by GA & ACO.

No	GA		ACO	
	Optimal path	Distance	Optimal path	Distance
1	1.2.6.14.15.26	13.648	1.2.6.14.15.26	13.648
2	1.3.5.11.12.13.26	13.5431	1.2.6.14.15.26	13.648
3	1.2.7.16.17.26	13.5431	1.2.6.14.15.26	13.648
4	1.2.6.14.15.26	13.648	1.2.6.14.15.26	13.648
5	1.2.6.14.15.26	13.648	1.2.6.14.15.26	13.648

3.3 Comparison of GA and ACO Solution Variation

The fitness of the path population will be evaluated after the population being initialized at initial stage of both algorithm processes. This fitness value represents accumulate data of distance obtained in each generations and can be used to determine the solution variation of both algorithms. This achieved by measuring the different between the maximum and minimum distance in each generation or by calculating the mean and standard deviation of the path distance in the generations itself. Table 6 and 7 tabulated below illustrates the reading of the maximum distance, minimum distance, differences between max and min distance, mean and standard deviation of the distance in 5th test runs. The reading of the distance is referring to the fitness value of the 1st population obtained at the 1st generation of the algorithm process.

Based on the results found, average distances between maximum and minimum value of ACO is

smaller compared to GA which is in ratio 1:9 or equal to 3 and 45. This is because ACO consists of accurate and robust initialization approach that capable to drive ants in the next generation to choose path which is near the optimal path while abandon the path which is far from the optimal path. This process then will affect the range of distance to be optimized by ACO in each generation is smaller compared to GA. Therefore, ACO can converge faster and the number of iteration also will be reduced. Table 7 above illustrate the range of distance found by ACO in the 1st generation which is differs from GA.

Table 6: Fitness Value of GA Populations.

Number of run	Max distance	Min distance	Differences (Δ max-min)	Mean, μ	Sd, δ	Iteration
1	58.03	13.65	44.382	22.99	0.22	10
2	60.45	13.65	46.799	23.06	0.31	7
3	58.36	13.65	44.712	22.92	0.26	8
4	60.45	13.65	46.799	23.09	0.31	7
5	58.34	13.65	44.692	22.89	0.26	8
Avg	59.126	13.65	45.477	22.99	0.27	8

Table 7: Fitness Value of Ants Populations.

Number of run	Max distance	Min distance	Differences (Δ max-min)	Mean, μ	Sd, δ	Iteration
1	18	13.65	4.352	13.781	0.00008	4
2	17.42	13.65	3.766	13.733	0.00004	4
3	17.42	13.65	3.766	13.835	0.00012	6
4	18	13.65	4.352	13.883	0.00028	4
5	14.49	13.65	0.837	13.723	0.00003	4
Avg	17.066	13.65	3.415	13.79	0.00011	22

For GA, there is no rules has been used to initialize the population where it based on random approaches. Effect from this process, the population in the initial generations will consists of optimal and non optimal path that will generate variety values of distance. This will cause the range of distance to be optimized by GA is bigger than ACO. Moreover, although GA will carry forward the optimal path to the next generation during the selection process, the possibility to obtain the population consists of non optimal child will repeated again. This is because the point to cross and mutate the chromosomes also will determine randomly and thus cause to the increment of the optimized data. As a result, it shows that the random initialization process of GA from one generation to other generation had cause GA to optimize the wide range of distance. Thus will also cause to the increment of time and iteration GA ta-

kes to find optimal solution.

The calculation of mean and standard deviation of the path distance of the 1st iteration in each test runs has been used to verify the range of distance optimized by both algorithms. Based on the results found in Tables 6 and 7 above, the value of the mean and standard deviation of ACO is smaller compared to GA. With the value of standard deviation of ACO which is approximate to 0, it can be proven that population of ACO are mostly consists of optimal path population because the data to be optimized is in a small range and near to 0 compared to GA. As a result, ACO will work efficiently and meet the convergence earlier compared to GA.

3.4 Comparison of GA and ACO Convergence Behavior

The efficiency of both algorithms to find optimal path during convergence time can be measured by observing the increment number of optimal path to goal in each generations as results tabulated in Table 8 and 9 below. In this experiment, the solution converge when the differences between maximum and minimum fitness of 20 of the 1st population is equal or less than 0.0001 (≤ 0.0001). This means that the algorithm will continuously repeat its process until the solution meets the requirement of the convergence that will drive the algorithm to stop its process.

Based on the results found, GA capable to find optimal path and converge around 215.361 sec in 7th iteration while ACO around 104.606 sec in 4th iteration. In GA, the increment of the optimal path population is slow and steady. 3 optimal paths found at initial stage of the random process which then followed by 4,5,7,13,17 and finally reach more than 20 population at the moment the solution converge at the 7th iteration. It was different with ACO where the number of optimal path population is increasing rapidly due to efficient rules provided. Start with 8 populations at the first place then continues with 11, 13 and finally 20 population in 4th times of iteration. From here, it has been proven that the number of optimal path increment from one generation to other generation also can be used to differentiate performances between both algorithms. ACO is the robust and efficient techniques compare to GA where it will not only increase optimal path rapidly but it capable to trigger itself to find the path faster with only a small number of iteration. The sufficient amount of population required for ACO to converge is the range of data to be minimized is in a big range as easy to found compared to GA.

Besides that, the change of the range of distance

in every generation also shows the efficiency of both algorithms reach the convergence solution. In ACO, the data to be optimized is decreased constantly and rapidly proportional to the increment of the number of population in each generation. This has been proven with the value of maximum distance, mean and standard deviation that will continuously decreasing until the solution converge at 4th generation as shown in Table 8 above. At the last generation where the solution converges, the value of mean is equal to the value of optimal path distance while the standard deviation is equal to 0 which shows that the solution converges efficiently. During this moment, ants will follow the same path and the path distance traverse by ants also become similar and the solution will reach convergence easily in a small computation time and a small number of iteration.

Table 8: GA converge at 7th generation.

Iteration	Mean, μ	Sd, δ	Max	Min	Number of optimal population
1	15.843	0.096	18	13.65	3
2	25.749	3.661	54.477	13.65	4
3	29.147	6.005	57.69	13.65	5
4	26.469	4.109	60.447	13.65	7
5	24.699	3.053	57.189	13.65	13
6	24.328	2.852	56.617	13.65	17
7	17.226	0.32	52.503	13.65	27(converge)
Avg	23.352	20.096	50.989	13.65	7

Table 9: ACO converge at 4th generation.

Iteration	Mean, μ	Sd, δ	Max	Min	Number of optimal population
1	13.956	0.002	18	13.65	8
2	13.805	0.0005	14.49	13.65	11
3	13.713	0.00008	13.90	13.65	13
4	13.648	0	13.65	13.65	20(converge)
Avg	13.781	0.0006	15.01	13.65	13

It was different with GA where the data of distance in each generation will keep changing and not constantly decreased like ACO. This is because the way this algorithm remains and increases the optimal path population in each generation was based on the random approaches. Effect from this random process, the population may consists of population of optimal or non-optimal population that consists variety amount of distance that will simultaneously contribute to inconstant distance value in each generations. With this inconstant value, the range data to be optimized in each generation cannot be predicted where sometimes it will converge in a small computation time and iteration if the range of data is small while

sometimes it need a long time and more iteration to converge if the range of data to be minimized is in a big range as results shown in Table 9 above.

4 CONCLUSIONS

The research indicates that GA and ACO were able to find an optimal path in feasible global static environments. The results show that for the selected environments, ACO has the capability to work more efficiently and more accurately than GA. This is because the computation time and iteration takes to find the optimal path is smaller. In addition, the optimal path found in each time run shows the accuracy of ACO. Furthermore, the range of data to be optimized is also smaller compared to GA which will also drive ACO behaviour to converge efficient and effectively. However, the advantages and limitations of both algorithms can be further explored to expand the applications of both optimization algorithms in RPP research area.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge University Technology MARA, Malaysia (UiTM) for supporting this research.

REFERENCES

- Charles.W.Warren (1993) Fast path planning using modified A* Method. *IEEE Transactions on Systems,Man and Cybernetics*.
- Dorigo, M. & Gambardella, L. M. (1997) Ant colony system: A Cooperative Learning Approach to the Traveling Salesman Problem(TSP). *Evolutionary Computation, IEEE Transactions on*.
- Dorigo, M., Stutzle T (2004) *Ant Colony Optimization*, The Bradford Book,The MIT Press Cambridge.Masachusetts, London, England.pp 1-305.
- Goldberg, D. (1994) Genetic and evolutionary algorithms come on age. *Proceedings of Communications ACM*, 37, 113-119.
- Genggqian, Tiejun L, Yuqing L, P. & Xiangdan P, H. (2005) The ant algorithm for solving robot path planning problem. *Third International Conference on Information Technology and Applications, 2005. ICITA 2005,4-9*
- Hart, P. E., Nilsson, N. J. & Raphael, B. (1968) A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *Systems Science and Cybernetics, IEEE Transactions on*, 4, 100-107.
- Hu, Y., X.Yang S (2004) A Knowledge Based Genetic Algorithm for Path Planning of Mobile Robot. *Proceedings of the 2004 IEEE International Conferences on Robotics and Automation New Oriens*, 4350-4355.
- Khatib, O. (1985) Real time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(10), 90-98.
- Mei, H., Tian Y,Zu L (2006) A Hybrid Ant Colony Optimization Algorithm for Path Planning of Robots in Dynamic Environment. *International Journal of Information Technology*, 12,No 3, 78-88.
- N.Sivanandam, S., N.Deepa S (2008) *Introduction to Genetic Algorithms*, Springer-Verlag Berlin Heidelberg 1-425.
- Nagib, G., Gharieb W (2004) Path planning for a mobile robot using Genetic Algorithm. *IEEE Proceedings of Robotics* 185-189.
- Netnevitsky, M. (2002) *Artificial Intelligence:A guide to intelligent Systems*, 2nd Edition, Addison-Wesley.
- Ramakrishnan, R., Zein Sabatto S (2001) Multiple Path Planning for a Group of Mobile Robot in a 2D Environment using Genetic Algorithms. *IEEE Transactions of Robotics and Systems*, 65-71.
- Sariff, N., Buniyamin N (2009) Comparative Study of Genetic Algorithm and Ant Colony Optimization Algorithm in Global Static Environment of Different Complexities. *2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA 2009), Daejeon, Korea*, 132-137.
- Sariff, N., Buniyamin N (June 2006) An Overview of Autonomous Robot Path Planning Algorithms. *4th Student Conference on Research and Development (Scored 2006)*, 184-188.
- Stentz, A. (1994) Optimal and Efficient Path Planning for Partially-Known Environments. *In Proceedings IEEE International Conference on Robotics and Automation*, 1-8.
- Sugihara, K., Smith.J (1997) GA for adaptive motion planning of an autonomous mobile robot. *Proc. 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '97)*, 138-143.
- Tu, J., X.Yang.S (2003) Genetic Algorithms Based Path Planning for a Mobile Robot. *Proceedings of the 2003 IEEE International Conference on Robotics & Automation, Taipei, Taiwan*, 1221-1226.
- Xin, D., Hua-Hua C and Wei Kang G (2005) Neural Network and Genetic Algorithm Based Global Path Planning in A Static Environment. *Journal of Zhejiang University Science*, 6A, 549-554.
- Yahja, A., Singh S, Stenz A (2000) An efficient on-line path planner for outdoor mobile robot. *Journal of Robotics and Autonomous Systems* 32, 129-143.
- Zelinsky, A., Yuta S (Oct 1993) A unified approach to planning, sensing and navigation for mobile robots. *Third International Symposium on Experimental Robotics Kyoto, Japan*, 28-30.
- Zheng, T. G., Huan H, Aaron S (2007) Ant Colony System ALgorithm for Real Time Globally Optimal Path Planing of Mobile Robots. *ACTA AUTOMATICA SINICA*, 33, 279-285.

A MULTI-ESTIMATION SCHEME FOR CONTROLLING THE BEVERTON-HOLT EQUATION IN ECOLOGY

S. Alonso-Quesada and M. De La Sen

Department of Electricity and Electronics, Faculty of Science and Technology, University of the Basque Country
Campus of Leioa, 48940-Leioa, Bizkaia, Spain
santiago.alonso@ehu.es, msen@we.lc.ehu.es

Keywords: Adaptive control, Beverton-Holt Equation, Carrying capacity, Control systems, Multi-estimation.

Abstract: This paper proposes an adaptive control algorithm to govern the solution of the Beverton-Holt equation under parametrical uncertainties and the potentially presence of additive disturbances. The control strategy is based on a multi-estimation scheme with a supervisor choosing on-line the active estimation model used to parameterize the controller. The tracking of a reference sequence with local modifications of the carrying capacity sequence around its nominal values is achieved with such a control strategy.

1 INTRODUCTION

Models based on the Beverton-Holt Equation (BHE) are very common in Ecology for the study of the evolution of species in their habitats, (Barrowman et al., 2003). Such models rely on more general discrete recursive equations proposed in (Stevic, 2010, Elsayed and Iricanin, 2009, Iricanin and Stevic, 2009a, 2009b). The BHE is a nonlinear equation given by (Beverton and Holt, 1957):

$$x_{k+1} = \frac{\mu_k K_k x_k}{K_k + (\mu_k - 1)x_k}, \quad k \in \mathbf{N}_0 := \mathbf{N} \cup \{0\} \quad (1)$$

where \mathbf{N} is the set of natural numbers, $x_0 > 0$ the initial species population size, x_k and x_{k+1} the population sizes at time instants kT (*spawning stock*) and $(k+1)T$ (*recruitment*), respectively, with T being the sampling period, and $\mu_k \in \mathbf{R}_0^+ := \mathbf{R}^+ \cup \{0\}$ and $K_k \in \mathbf{R}^+$ the population *intrinsic growth rate* and the *environment carrying capacity* at the time instant kT , respectively, with \mathbf{R}^+ being the set of positive real numbers. The intrinsic growth rate sequence $\{\mu_k\}_0^\infty$ is determined by life cycle and demographic properties like species growth rate, survivorship rate and so on. The carrying capacity sequence $\{K_k\}_0^\infty$ is a characteristic of the habitat depending on resources availability, temperature, humidity and so on.

Typically, $\mu_k > 1$ and so $\{\mu_k\}_0^\infty$ as $\{K_k\}_0^\infty$ are cyclic sequences as a consequence of periodic fluctuations are common in biological problems. The carrying capacity sequence is susceptible of being locally modified by means of small changes of temperature, humidity and so on around nominal values. This fact can be used to control the species population size in a closed or semi-closed habitat (De la Sen and Alonso-Quesada, 2008, De la Sen and Alonso-Quesada, 2009). Such control strategies take advance of the linearity of the Beverton-Holt inverse equation (BHIE) (Stevic, 2006) so that conventional techniques developed for linear control systems may be used in order to govern the BHIE solution and then also the BHE one. In such works, the controllers are designed for matching a prescribed reference model by the BHE model possibly affected by the presence of additive disturbances. The reference models are another BHE with suitable intrinsic growth rate and carrying capacity sequences. The paper (De la Sen and Alonso-Quesada, 2008) considers the perfect knowledge of the sequences defining the standard BHE while the research in (De la Sen and Alonso-Quesada, 2009) extends the discussion to the adaptive case since the intrinsic growth rate and carrying capacity sequences are partially or fully unknown. In both cases the environment carrying capacity may be locally modified around its reference values to achieve the prescribed behaviour.

The matching objective by local modifications of the carrying capacity sequence is only available and practical if the BHE to be controlled as well as the BHE used as the reference model are locally deviated from each other. However, such a condition may not be guaranteed, at least in an adaptive control context where some system parameters are unknown. In this sense, the *main contribution* of the present paper lies in the *design of an adaptive control scheme with a set of potential reference models, instead of a unique one, to be matched in order to circumvent such a drawback*. For such a purpose, *the reference models included in the set are suitably chosen such that at least one of them is sufficiently closed to the unknown BHE at any sampling time*. This quality may be guaranteed with the inclusion of a *large number of reference models in the control scheme and a well distribution of them within the BHE parameters space*. An estimation algorithm is associated to each reference model. Such estimators work in parallel and a supervisor activates on-line the estimation algorithm providing the closest estimated model to the unknown BHE at each sampling instant. The closeness is measured by means of the estimation error associated to each algorithm. The supervisor function implies the switching among the estimated models provided by the estimators included in the adaptive control scheme. Then, *a minimum residence time is maintained in operation the active estimated model in order to achieve a good tracking behaviour and the stability of the control system* (De la Sen and Alonso-Quesada, 2006, Narendra and Balakrishnan, 1997). In this way, the adaptive control scheme works with a time-varying reference model, what is compatible in a species population system subject to periodic fluctuations.

2 PROBLEM STATEMENT

The change of variable $s_k = x_k^{-1}$ in (1) leads directly to the BHIE (Stevic, 2006):

$$s_{k+1} = a_k s_k + b_k u_k, \quad s_0 = x_0^{-1} > 0 \quad (2)$$

where $a_k = \mu_k^{-1}$, $b_k = 1 - a_k$ and $u_k = K_k^{-1} \forall k \in \mathbf{N}_0$. Note that the inverse carrying capacity can act as a control action. If an additive disturbance sequence $\{\eta_k^0\}_0^\infty$ exists, one gets a more general version of (2):

$$s_{k+1} = a_k (s_k - u_k) + u_k + \eta_k^0 \quad (3)$$

The disturbance may include the effects in the solution of parametrical uncertainties, for instance, in the intrinsic growth rate, or effects, like migrations or local migrations which are not taken into account in the standard BHE. The following assumptions are considered related to the BHE:

Assumptions 1.

(i) $1 + \varepsilon_\mu \leq \mu_k < \infty$ and $\varepsilon_K \leq K_k < \infty \quad \forall k \in \mathbf{N}_0$ and some $\varepsilon_\mu, \varepsilon_K \in \mathbf{R}^+$.

(ii) $|\eta_k^0| \leq \bar{\eta}_k < \infty \quad \forall k \in \mathbf{N}_0$ with $\{\bar{\eta}_k\}_0^\infty$ known. ***

Remark 1.

(i) The BHIE is stable and controllable since both

$$a_k \leq \frac{1}{1 + \varepsilon_\mu} < 1 \quad \text{and} \quad b_k \geq \frac{\varepsilon_\mu}{1 + \varepsilon_\mu} > 0 \quad \forall k \in \mathbf{N}_0$$

derived from Assumption 1(i).

(ii) All solutions of the BHE and BHIE are uniformly bounded and positive provided that $x_0 > 0$ if both Assumptions 1 hold (De la Sen and Alonso-Quesada, 2009). ***

Since the control action is the inverse of the environment carrying capacity it is not admitted a large deviation from its nominal values for tracking purposes in a practical situation. That means that the reference model to be matched by the current BHE has to be sufficiently closed to it. Such a reference model might be another BHE as follows:

$$x_{k+1}^* = \frac{\mu_k^* K_k^* x_k^*}{K_k^* + (\mu_k^* - 1)x_k^*} \quad \forall k \in \mathbf{N}_0 \quad (4)$$

which defines the suitable solution through the appropriate reference values of the intrinsic growth rate and the environment carrying capacity sequences, $\{\mu_k^*\}_0^\infty$ and $\{K_k^*\}_0^\infty$ with $\mu_k^* > 1 \quad \forall k \in \mathbf{N}_0$.

. Its corresponding reference BHIE is:

$$s_{k+1}^* = a_k^* s_k^* + b_k^* r_k \quad (5)$$

with reference input $r_k = (K_k^*)^{-1}$, and parameter sequences $a_k^* = (\mu_k^*)^{-1}$ and $b_k^* = 1 - (\mu_k^*)^{-1} \quad \forall k \in \mathbf{N}_0$.

Assume that the carrying capacity $\{K_k^*\}_0^\infty$ and the intrinsic growth rate $\{\mu_k^*\}_0^\infty$ sequences of a reference BHE are given together with the sequences $\{\delta_k\}_0^\infty$ and $\{\lambda_k\}_0^\infty$, with $\delta_k \in [0, 1)$ and $\lambda_k \in \left[0, \frac{1}{\mu_k^*}\right)$ $\forall k \in \mathbf{N}_0$. The following definition and proposition

are concerned with the available BHE models for tracking, with a sufficiently small tracking error, a given BHE reference solution by local modifications of the environment carrying capacity (De la Sen and Alonso-Quesada, 2009).

Definition 1. A class $\mathbb{C}_{\text{BHE}}(\mathbf{K}_k^*, \mu_k^*, \delta_k, \lambda_k)$ of BHEs exists parameterized by some sequences $\{\mathbf{K}_k\}_0^\infty$ and $\{\mu_k\}_0^\infty$ such that $\mathbf{K}_k \in \left[\frac{1-\delta_k}{1+\lambda_k} \mathbf{K}_k^*, \frac{1+\delta_k}{1-\lambda_k} \mathbf{K}_k^* \right]$ and $\mu_k \in \left[\frac{(1+\lambda_k)\mu_k^*}{1+\lambda_k\mu_k^*}, \frac{(1-\lambda_k)\mu_k^*}{1-\lambda_k\mu_k^*} \right] \quad \forall k \in \mathbf{N}_0$. ***

Proposition 1. If (i) the upper-bound sequence $\{\bar{\eta}_k\}_0^\infty$ for the absolute value of the additive disturbance $\{\eta_k^0\}_0^\infty$ of the BHIE associated to a BHE belonging to the class $\mathbb{C}_{\text{BHE}}(\mathbf{K}_k^*, \mu_k^*, \delta_k, \lambda_k)$ is such that $\bar{\eta}_k \leq \frac{\delta_k(\mu_k^* - 1)}{(1+\delta_k)\mu_k^* \mathbf{K}_k^*} \quad \forall k \in \mathbf{N}_0$ and (ii) the initial condition s_0 fulfils $s_0 \geq s_0^*(1-\gamma_0) > 0$ for some monotonically increasing sequence $\{\gamma_k\}_0^\infty$ with $\gamma_k \in \mathbf{R}^+$, $\gamma_k < 1 \quad \forall k \in \mathbf{N}_0$ and such that $\text{Max}_{k \in \mathbf{N}_0} \left\{ \frac{1}{1+\delta_k} \left[\delta_k + \left(\frac{\varepsilon_\mu - 2}{\varepsilon_\mu + 2} \right)^2 \right] \right\} \leq \gamma_0 < 1$, then:

(i) The control law:

$$\mathbf{u}_k = \begin{cases} t_k r_k + f_k s_k - \omega_k & \text{if } s_k \leq \frac{2}{(1+\delta_k)\mathbf{K}_k^*} \\ r_k - \omega_k & \text{otherwise} \end{cases} \quad (6)$$

with the parameter sequences given by:

$$t_k = \frac{1-a_k^*}{1-a_k}; \quad f_k = 1-t_k; \quad \omega_k = \frac{\bar{\eta}_k}{1-a_k} \quad (7)$$

guarantees $\mathbf{K}_k \in \left[\frac{1-\delta_k}{1+\lambda_k} \mathbf{K}_k^*, \frac{1+\delta_k}{1-\lambda_k} \mathbf{K}_k^* \right] \quad \forall k \in \mathbf{N}_0$,

(ii) $s_k \geq s_k^*(1-\gamma_k) \quad \forall k \in \mathbf{N}_0$, where $\{s_k\}_0^\infty$ is the solution of the BHIE and $\{s_k^*\}_0^\infty$ the solution of the inverse of the reference BHE of the class $\mathbb{C}_{\text{BHE}}(\mathbf{K}_k^*, \mu_k^*, \delta_k, \lambda_k)$ and

(iii) the BHE solution $\{x_k\}_0^\infty$ is upper-bounded by the sequence $\left\{ \frac{x_k^*}{1-\gamma_k} \right\}_0^\infty$, where $\{x_k^*\}_0^\infty$ is the solution of the reference BHE of such a class. ***

Remark 2. Proposition 1 implies that given any BHE belonging to an arbitrary class $\mathbb{C}_{\text{BHE}}(\mathbf{K}_k^*, \mu_k^*, \delta_k, \lambda_k)$, local modifications of the carrying capacity around the reference sequence $\{\mathbf{K}_k^*\}_0^\infty$ can be used to achieve the control objective. Namely, a sufficiently small tracking error between the solutions of the given BHE and that of the reference one of such a class can be obtained. ***

3 ADAPTIVE CONTROL

An estimation scheme is incorporated to solve the control problem in the case that the intrinsic growth rate sequence $\{\mu_k\}_0^\infty$ of the BHE (or the sequence $\{a_k\}_0^\infty$ of the BHIE) is unknown. In the context of adaptive control, the BHIE (3) can be written as:

$$s_{k+1} = a(s_k - u_k) + u_k + \eta_k \quad (8)$$

for some unknown constant $a = \mu^{-1}$ with $\{\eta_k\}_0^\infty$ given by:

$$\eta_k := (a_k - a)(s_k - u_k) + \eta_k^0 \quad (9)$$

In this way, the nominal parameter a of the BHIE is constant and $\{\eta_k\}_0^\infty$ incorporates the deviations of the intrinsic growth rate with respect to the unknown constant μ and, possibly, other unstructured disturbance contributions in $\{\eta_k^0\}_0^\infty$. If $\eta_k \equiv 0$, the resulting particular case of (8) is called the nominal BHIE.

The estimation algorithm provides an estimated BHIE given by:

$$\hat{s}_{k+1} = \hat{a}_k(s_k - u_k) + u_k \quad (10)$$

where \hat{a}_k denotes the estimate of a at the k -th sample. Moreover, an estimation error given by:

$$e_{k+1} := s_{k+1} - \hat{s}_{k+1} = -\tilde{a}_k(s_k - u_k) + \eta_k \quad (11)$$

is associated to the estimation algorithm where $\tilde{a}_k := \hat{a}_k - a$ is the parametrical error. Finally, the tracking error between the solution of the BHIE to be controlled and that of the reference model (5) is:

$$\varepsilon_{k+1} := s_{k+1} - s_{k+1}^* = e_{k+1} + \hat{s}_{k+1} - s_{k+1}^* \quad (12)$$

The tracking error depends on the estimation error and the deviation of the estimated model from the reference one. Then, *the use of a multi-estimation scheme and a supervisor choosing the estimation algorithm providing the smallest estimation error*, instead of the use of a unique estimation algorithm, *will improve the tracking objective*. Furthermore, the deviation between the estimated model and the reference one can be sufficiently small if (i) each estimation algorithm is associated to a different BHE reference model defining a class $\mathbb{C}_{\text{BHE}}(\mathbf{K}_k^*, \mu_k^*, \delta_k, \lambda_k)$ and (ii) each one includes a parameter projection for guaranteeing the closeness of both corresponding estimated and reference models. In this way, *if the multi-estimation scheme is composed by a large number of reference model/estimation algorithm pairs and the reference models are well distributed within the definition domain of $\{a_k\}_0^\infty$, then at least one of the estimated models will be sufficiently close to the unknown BHE to be controlled for all time*. As a consequence, such an unknown BHE solution will be able to track that of the reference model associated to the estimation algorithm activated by the supervisor by means of locally modification of the environment carrying capacity around its nominal values. Both reasons motivate the use of a multi-estimation scheme with several estimation algorithms working in parallel in the adaptive control scheme. Furthermore, such a scheme makes that the reference model to be tracked is online changed by the supervisor, what is of interest for ecologic system subject to periodic fluctuations.

3.1 Multi-estimation Scheme

A set $S_e := \{1, 2, \dots, n_e\}$ of estimation algorithms working in parallel is considered. Each one is associated to a different class $\mathbb{C}_{\text{BHE}}(\mathbf{K}_k^*, \mu_k^*, \delta_k, \lambda_k)$. All of them use a least-squares algorithm with a parameters projection and a dead-zone. The projection is used to obtain an estimation model belonging to the corresponding class and the dead-zone for dealing with the presence of potentially

disturbances affecting to the nominal BHIE. Each algorithm is defined by:

$$\bar{a}_{k+1}^{(i)} = \hat{a}_k^{(i)} + \frac{\sigma_k^{(i)} (s_k - u_k) e_{k+1}^{(i)}}{1 + \beta_k^{(i)} (s_k - u_k)^2} \quad (13.a)$$

$$\hat{a}_{k+1}^{(i)} = \text{Proj} \left\{ \bar{a}_{k+1}^{(i)} \right\} = \begin{cases} \frac{1 - \lambda_{k+1}^{(i)} \mu_k^{*(i)}}{(1 - \lambda_{k+1}^{(i)}) \mu_k^{*(i)}} & \text{if } \bar{a}_{k+1}^{(i)} < \frac{1 - \lambda_{k+1}^{(i)} \mu_k^{*(i)}}{(1 - \lambda_{k+1}^{(i)}) \mu_k^{*(i)}} \\ \frac{1 + \lambda_{k+1}^{(i)} \mu_k^{*(i)}}{(1 + \lambda_{k+1}^{(i)}) \mu_k^{*(i)}} & \text{if } \bar{a}_{k+1}^{(i)} > \frac{1 + \lambda_{k+1}^{(i)} \mu_k^{*(i)}}{(1 + \lambda_{k+1}^{(i)}) \mu_k^{*(i)}} \\ \bar{a}_{k+1}^{(i)} & \text{otherwise} \end{cases} \quad (13.b)$$

$$\text{with } \hat{a}_0^{(i)} \in \left[\frac{1 - \lambda_0^{(i)} \mu_0^{*(i)}}{(1 - \lambda_0^{(i)}) \mu_0^{*(i)}}, \frac{1 + \lambda_0^{(i)} \mu_0^{*(i)}}{(1 + \lambda_0^{(i)}) \mu_0^{*(i)}} \right] \subseteq (0, 1)$$

where $e_k^{(i)} = s_k - \hat{s}_k^{(i)}$, $\forall k \in \mathbf{N}_0$ and $\forall i \in S_e$, is the estimation error of the i -th algorithm at the sampling instant kT . The real sequence $\{\beta_k^{(i)}\}_0^\infty$ is such that $\beta_k^{(i)} > 0 \quad \forall k \in \mathbf{N}_0$ and $\{\sigma_k^{(i)}\}_0^\infty$ a relative dead-zone defined as:

$$\sigma_k^{(i)} = \begin{cases} 0 & \text{if } |e_{k+1}^{(i)}| \leq \zeta^{(i)} \bar{\eta}_k \\ \frac{2\beta_k^{(i)} (\zeta^{(i)} - 1 - \zeta_1^{(i)})}{\zeta^{(i)}} & \text{if } |e_{k+1}^{(i)}| > \zeta^{(i)} \bar{\eta}_k \end{cases} \quad (13.c)$$

for some prefixed real constants $\zeta^{(i)} > 1$ and $\zeta_1^{(i)} \in (0, \zeta^{(i)} - 1)$ where $\{\bar{\eta}_k\}_0^\infty$ is a known upper-bound for $\{|\eta_k|\}_0^\infty$ [see Assumption 1(ii)].

Such an algorithm meets the following properties (De la Sen and Alonso-Quesada, 2009).

Lemma 1.

(i) The sequence $\{\hat{a}_k^{(i)}\}_0^\infty$ is bounded and converges asymptotically to a finite $\hat{a}_\infty^{(i)}$,

(ii) The sequences $\left\{ \left(\frac{\sigma_k^{(i)}}{1 + \beta_k^{(i)} (s_k - u_k)^2} \right)^{1/2} |e_{k+1}^{(i)}| \right\}_0^\infty$

and $\left\{ \left(\frac{\sigma_k^{(i)}}{1 + \beta_k^{(i)} (s_k - u_k)^2} \right)^{1/2} \bar{\eta}_k \right\}_0^\infty$ are bounded and

both tend asymptotically to zero. ***

3.2 Supervisory System

This element chooses on-line one of the estimation algorithms which compose the multi-estimation scheme, namely, that closest to the unknown BHE. For such a purpose, a cost function given by:

$$F_k^{(i)} = \sum_{j=0}^k \rho^{k-j} (e_j^{(i)})^2 \quad (14)$$

with the forgetting factor $\rho \in (0,1) \cap \mathbf{R}^+$, is evaluated by the supervisor $\forall k \in \mathbf{N}_0$ and $\forall i \in S_e$ and the estimation algorithm minimizing such a function is activated. Furthermore, the supervisor maintains activated such an algorithm at least a minimum number N_{\min} of sampling periods before switching to a different one. This residence time prevents against the instability of the control system caused by an eventual great amount of switches concentrated in a short time interval. Then, the switching law is given by:

$$c_k = \begin{cases} c_{k-1} & \text{if } k - k' < N_{\min} \\ \ell & \text{otherwise} \end{cases} \quad (15)$$

where $k'T$ is the sampling instant at which the last switching occurred before the current time instant kT and $\ell := \text{Min} \{i \in S_e \mid F_k^{(i)} \leq F_k^{(j)} \forall i, j \in S_e\}$.

3.3 Adaptive Control Law

An adaptive control law with the same structure as (6)-(7) by replacing the true parameter a_k by its estimate $\hat{a}_k^{(c_k)}$ and by deleting the correcting signal for disturbances ω_k is used to generate the suitable value for the carrying capacity sequence at each sampling time. The super-index (c_k) denotes the estimation algorithm which is maintained active by the supervisor at the current sampling instant kT . The control term relative to the disturbances is omitted since such disturbances are treated by the inclusion of the dead-zone in each estimation algorithm. Such a control law is:

$$u_k = \begin{cases} t_k (r_k - s_k) + s_k & \text{if } s_k \leq \frac{2}{(1 + \delta_k^{(c_k)}) K_k^{*(c_k)}} \\ (K_k^{*(c_k)})^{-1} & \text{otherwise} \end{cases} \quad (16)$$

where $\{K_k^{*(c_k)}\}_0^\infty$ is the carrying capacity sequence of the active reference model at the current sampling

instant and the sample-dependent controller parameter is given by:

$$t_k = \frac{1 - \hat{a}_k^{*(c_k)}}{1 - \hat{a}_k^{(c_k)}} \quad (17)$$

i.e., the control is parameterized from the active estimated model at the current sampling instant.

3.4 Stability Analysis

The following additional assumption has to be considered for proving the closed-loop stability.

Assumption 2. There exist known finite nonnegative real constants ϑ_1 and ϑ_2 such that $\bar{\eta}_k \leq \vartheta_1 + \vartheta_2 \text{Max}_{0 \leq j \leq k} \{s_j\}$. ***

Remark 3. This assumption implies a slow growing of the unknown disturbances with respect to the solution of the BHIE. This is a reasonable assumption used in adaptive control theory since a complete lack of knowledge of disturbances makes impossible the stabilization in the general case (De la Sen and Alonso-Quesada, 2006, Feng, 1999).

The control system stability is based on the following features: **(i)** the adaptive control law (16) with any of the estimation algorithms maintained active by the supervisor for all time stabilizes the control system while achieving a sufficiently small tracking error if the unknown BHE is closed to the reference BHE model corresponding to such an estimation algorithm for all time by means of locally modifications of the environment carrying capacity (De la Sen and Alonso-Quesada, 2009), and **(ii)** the switching law in the supervisory system guarantees a minimum residence time in the active estimation algorithm, which is crucial to avoid instability caused by an eventual high concentration of switches in a short time interval (Narendra and Balakrishnan, 1997). In summary, the switching law allows to change the reference model to be tracked by the current BHE to ensure the closeness between such a BHE and the active reference one.

4 NUMERICAL EXAMPLE

A BHE (1) defined by an unknown intrinsic growth rate sequence $\{\mu_k\}_0^\infty$, which is given by:

$$\mu_k = \begin{cases} 1.65 & \text{if } 360 \cdot j \leq k < 360 \cdot j + 29 \\ 1.6 & \text{if } 360 \cdot j + 30 \leq k < 360 \cdot j + 59 \\ 1.65 & \text{if } 360 \cdot j + 60 \leq k < 360 \cdot j + 89 \\ 1.75 & \text{if } 360 \cdot j + 90 \leq k < 360 \cdot j + 119 \\ 1.85 & \text{if } 360 \cdot j + 120 \leq k < 360 \cdot j + 149 \\ 1.95 & \text{if } 360 \cdot j + 150 \leq k < 360 \cdot j + 179 \\ 2 & \text{if } 360 \cdot j + 180 \leq k < 360 \cdot j + 209 \\ 1.95 & \text{if } 360 \cdot j + 210 \leq k < 360 \cdot j + 239 \\ 1.85 & \text{if } 360 \cdot j + 240 \leq k < 360 \cdot j + 269 \\ 1.8 & \text{if } 360 \cdot j + 270 \leq k < 360 \cdot j + 299 \\ 1.75 & \text{if } 360 \cdot j + 300 \leq k < 360 \cdot j + 329 \\ 1.7 & \text{if } 360 \cdot j + 330 \leq k < 360 \cdot j + 359 \end{cases} \quad (18)$$

and a known nominal environment carrying capacity sequence $\{K_k^{\text{nom}}\}_0^\infty$, given by:

$$K_k^{\text{nom}} = \begin{cases} 185 & \text{if } 360 \cdot j \leq k < 360 \cdot j + 29 \\ 180 & \text{if } 360 \cdot j + 30 \leq k < 360 \cdot j + 59 \\ 185 & \text{if } 360 \cdot j + 60 \leq k < 360 \cdot j + 89 \\ 200 & \text{if } 360 \cdot j + 90 \leq k < 360 \cdot j + 119 \\ 205 & \text{if } 360 \cdot j + 120 \leq k < 360 \cdot j + 149 \\ 210 & \text{if } 360 \cdot j + 150 \leq k < 360 \cdot j + 179 \\ 220 & \text{if } 360 \cdot j + 180 \leq k < 360 \cdot j + 209 \\ 215 & \text{if } 360 \cdot j + 210 \leq k < 360 \cdot j + 239 \\ 210 & \text{if } 360 \cdot j + 240 \leq k < 360 \cdot j + 269 \\ 200 & \text{if } 360 \cdot j + 270 \leq k < 360 \cdot j + 299 \\ 190 & \text{if } 360 \cdot j + 300 \leq k < 360 \cdot j + 329 \\ 185 & \text{if } 360 \cdot j + 330 \leq k < 360 \cdot j + 359 \end{cases} \quad (19)$$

$\forall k, j \in \mathbf{N}_0$, with $T=1$ day as sampling period, is considered. i.e., both are piecewise constant periodic sequences with period equal to 1 year. Note that each line of (18)-(19) corresponds to values of the sequences during a month approximately. The nominal carrying capacity sequence is susceptible of locally modifications in order to achieve the control objective. Such a *control objective* is that the BHE solution $\{x_k\}_0^\infty$ tracks a suitably chosen close reference sequence $\{x_k^{*(c_k)}\}_0^\infty$ with a sufficiently small tracking error. The reference sequence is chosen on-line by a supervisor among four potential sequences $\{x_k^{*(i)}\}_0^\infty$, for $i \in S_e := \{1, 2, 3, 4\}$, each one issued by the BHE reference model defining a different class $\mathbb{C}_{\text{BHE}}^{(i)}(K_k^{*(i)}, \mu_k^{*(i)}, \delta_k^{(i)}, \lambda_k^{(i)})$. Such reference models have been chosen so that at least one of them be sufficiently close to the unknown BHE at each sampling time. The four classes used in the example

are defined by the same carrying capacity $\{K_k^{*(i)}\}_0^\infty = \{K_k^{\text{nom}}\}_0^\infty$, $\{\delta_k^{(i)}\}_0^\infty$ and $\{\lambda_k^{(i)}\}_0^\infty$ sequences with $\delta_k^{(i)} = 0.0421$ and $\lambda_k^{(i)} = 0.0526 \quad \forall k \in \mathbf{N}_0$ and $\forall i \in S_e$ and different sequences $\{\mu_k^{*(i)}\}_0^\infty$ for the reference intrinsic growth rates, namely, $\mu_k^{*(1)} = 1.55$, $\mu_k^{*(2)} = 1.95$, $\mu_k^{*(3)} = 2.05$ and $\mu_k^{*(4)} = 1.75 \quad \forall k \in \mathbf{N}_0$.

The unknown BHE to be controlled is associated to the BHIE given by (8)-(9) with an unknown parameter $a=0.6$, which would correspond to a constant intrinsic growth rate $\mu = 1.6667$. Such a parameter has to be estimated to parameterize the adaptive control law (16) by using (14), (15) and (17). For such a purpose, four estimation algorithms working in parallel are included in the multi-estimation scheme, each one associated with each potential reference sequence $\{x_k^{*(i)}\}_0^\infty$ with $i \in S_e$. Each algorithm is defined by (13) with the same sequence $\{\beta_k^{(i)}\}_0^\infty$, namely $\beta_k^{(i)} = 10^{10} \quad \forall k \in \mathbf{N}_0$, and the same parameters $\zeta_1^{(i)} = 1.011$ and $\zeta_2^{(i)} = 0.01$ for all of them, i.e. $\forall i \in S_e$. Moreover, the constants $\vartheta_1 = 2 \times 10^{-5}$ and $\vartheta_2 = 10^{-6}$ are used to build the upper-bound of the contribution of the unmodeled dynamics. The estimation algorithms are, respectively, initialized with $\hat{a}_0^{(1)} = 0.66$, $\hat{a}_0^{(2)} = 0.49$, $\hat{a}_0^{(3)} = 0.46$ and $\hat{a}_0^{(4)} = 0.55$. Note that each estimated model is initialized within its corresponding class and they cannot leave from them due to the projection included in each estimation algorithm.

The initial population of the species is $x_0 = 300$ and that of the reference sequence $x_0^{*(c_0)} = 300$ with $c_0 = 1$ being the initialization for the switching law of the supervisor. The results obtained with the adaptive control system with the multi-estimation scheme are displayed in the following figures.

Figures 1 and 2 show the time evolution of the population size, active reference model solution and tracking error sequences in a year approximately. An acceptable tracking of the active reference by the supervisor can be observed from such figures. Figure 3 shows that local modifications of the inverse of the environment carrying capacity are sufficient to achieve such a tracking performance. Note that the control sequence is within the domain delimited by the lower and upper bounds associated

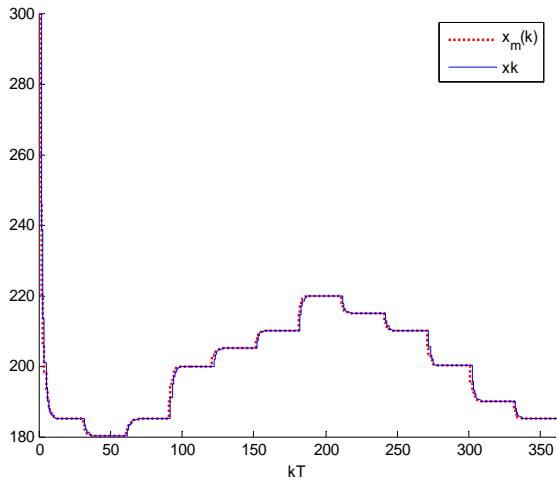


Figure 1: Evolution of the population size and the reference sequence activated by the supervisor.

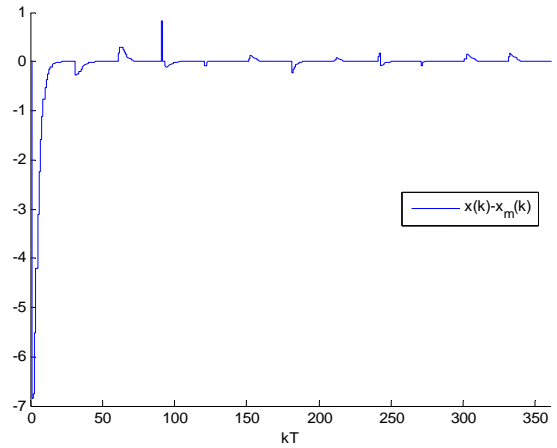


Figure 2: Evolution of the tracking error.

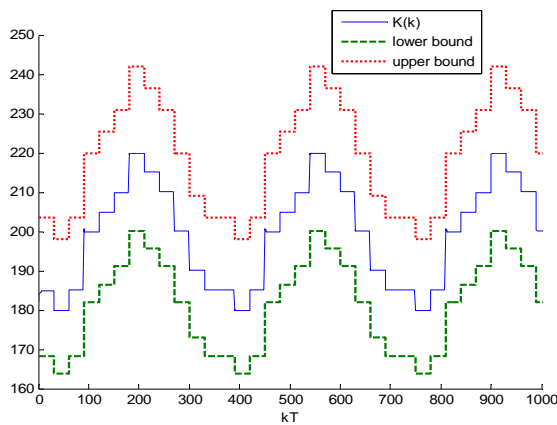


Figure 3: Evolution of the control sequence (inverse of the environment carrying capacity).

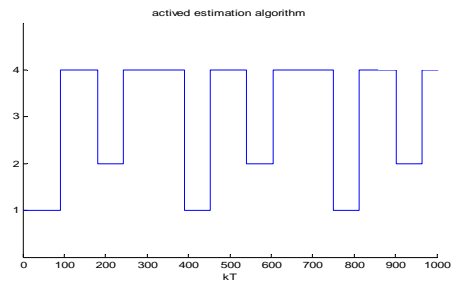


Figure 4: Estimation algorithm/reference model pair activated by the supervisor.

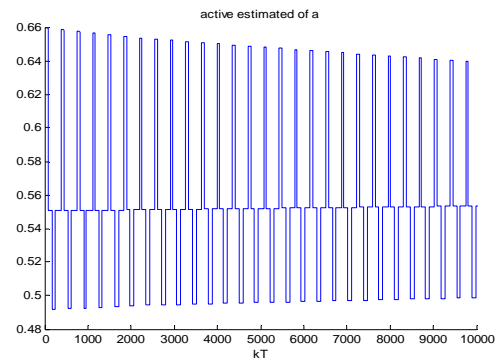


Figure 5: Evolution of the estimated of the active algorithm.

to local modifications around nominal values of the carrying capacity. Figure 4 displays the estimation algorithm which is online activated by the supervisor during the simulation. The active algorithm is changed by the supervisor several times during a year, which is reasonable due to the periodic fluctuations in the species intrinsic growth rate. Figure 5 displays the time evolution of the estimated of the unknown parameter corresponding to the active estimation algorithm.

Finally, the performance indexes given by

$$J_1(k) = \sum_{j=0}^k (x_j^{(i)} - x_j^{*(i)})^2 \quad \forall k \in \mathbf{N}_0 \quad \text{and} \quad \forall i \in S_e, \text{ if an}$$

adaptive control algorithm with a unique estimation algorithm (without supervisor) is used, or given by

$$J_m(k) = \sum_{j=0}^k (x_j^{(c_j)} - x_j^{*(c_j)})^2 \quad \text{if the multi-estimation}$$

scheme is used, are considered in order to compare the tracking performance of the developed multi-estimation scheme with the tracking results obtained with any of the single estimations algorithms working alone. Both indexes are measures of the tracking error accumulated during the simulation. Figure 6 below displays the performance indexes corresponding to the four simulations with the single

estimation algorithms and the simulation with the multi-estimation scheme incorporating the supervisor. Note that the best behaviour is obtained with the multi-estimation scheme, what motivates the use of the adaptive control strategy developed in this paper.

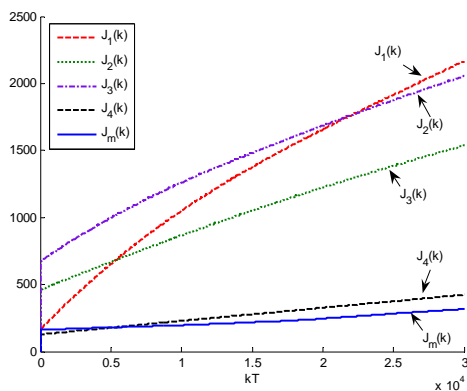


Figure 6: Tracking performances indexes.

5 CONCLUSIONS

BHE models are commonly used in Ecology to describe the time evolution of species populations in their habitats. Actually these models are subject to parametrical uncertainties what motivates the use of adaptive control techniques for such a purpose. The design of an adaptive control system with a multi-estimation scheme to achieve the solution of the BHE tracks a desired reference signal has been developed. The proposed use of a multi-estimation scheme instead of a single estimation one is due to two reasons, mainly. On one hand, Ecology systems are usually time-varying in the sense that their parameters suffer periodic fluctuations. On the other hand, the signal used as control is the inverse of the carrying capacity sequence, which depends on the habitat characteristics. Then, locally modifications of such a sequence around their nominal values are only available to control the BHE solution. This constraint makes that a suitable tracking performance is only guaranteed if the BHE and the reference model are locally deviated from each other. Then, a set of potential reference models, each one associated to an estimation algorithm, instead of a unique one improves the tracking behavior as it has been illustrated by some simulation results.

Future research will extend these adaptive control techniques to other Ecological systems as, for example, epidemic propagation models.

ACKNOWLEDGEMENTS

The authors are very grateful to MCYT by its support through grants DPI2006-00714 and DPI2009-07197.

REFERENCES

- Barrowman, N.J., Myers, R.A., Hilborn, R., Kehler, D.G., Field, C.A., 2003. The variability among populations of coho salmon in the maximum productive rate and depensation. *Ecological Applications* 13, pp. 784-793.
- Beverton, R.J.H., Holt, S.J., 1957. On the dynamics of exploited fish populations. *Fish. Invest.* 19, p. 1.
- De la Sen, M., Alonso-Quesada, S., 2006. Adaptive control of time-invariant systems with discrete delays subject to multiestimation. *Discrete Dynamics in Nature and Society* 2006, Article ID 41973, 27 pages, doi: 10.1155/DDNS/2006/41973.
- De la Sen, M., Alonso-Quesada, S., 2008. A control theory point of view on Beverton-Holt equation in population dynamics and some of its generalizations. *Applied Mathematics and Computation* 199, pp. 464-481.
- De la Sen, M., Alonso-Quesada, S., 2009. Control issues for the Beverton-Holt equation in ecology by locally monitoring the environment carrying capacity: Non-adaptive and adaptive cases. *Applied Mathematics and Computation* 215, pp. 464-481.
- Elsayed, E. M., Iricanin, B. D., 2009. On a max-type and a min-type difference equation. *Applied Mathematics and Computation* 215, pp. 608-614.
- Feng, G., 1999. Analysis of a new algorithm for continuous-time robust adaptive control. *IEEE Transactions on Automatic Control* 44, pp. 1764-1768.
- Iricanin, B. D., Stevic, S., 2009. Eventually constant solutions of a rational difference equation. *Applied Mathematics and Computation* 215, pp. 854-856.
- Iricanin, B. D., Stevic, S., 2009. On some rational difference equations. *Ars Combinatoria* 92, pp. 67-72.
- Narendra, K. S., Balakrishnan, J., 1997. Adaptive control using multiple models. *IEEE Transactions on Automatic Control* 42, pp. 171-187.
- Stevic, S., 2006. A short proof of the Cushing-Henson conjecture. *Discrete Dynamics in Nature and Society* 2006, Article ID 37264, 5 pages, doi: 10.1155/DDNS/2006/37264.
- Stevic, S., 2010. On a generalized max-type difference equation from automatic control theory. *Nonlinear Analysis – Theory, Methods and Applications* 72, pp. 1841-1849.

STRONG STABILIZATION BY OUTPUT FEEDBACK CONTROLLERS FOR INPUT-DELAYED LINEAR SYSTEMS

Baozhu Du, James Lam

*Department of Mechanical Engineering, University of Hong Kong, Hong Kong, China
dubaozhu@gmail.com, jlam@hku.hk*

Zhan Shu

*Hamilton Institute, National University of Ireland, Maynooth, Ireland
hustd8@gmail.com*

Keywords: Algorithm, Delay Systems, Stability, Strong Stabilization.

Abstract: This paper addresses the strong stabilization problem for continuous-time linear systems with an unknown input delay using a dynamic output feedback. New criteria for output feedback stabilizability are proposed for the closed-loop system in terms of matrix inequalities with the separation of controller gain and not only Lyapunov matrix but also system matrices. Based on the new characterization, an iterative algorithm is given to design the strong output feedback controllers with the aid of a slack matrix introduced. The effectiveness and merits of the proposed approach are shown through a numerical example.

1 INTRODUCTION

It is well known that even a simple linear system with a single delay imposes difficulties and restrictions on the design of a stabilization controller. The stabilization problem for linear systems with an unknown delay in the input signal is still a difficult one as seen in (Chen and Zheng, 2006), (Respondek, 2008) and (Tadmor, 2000) (and the references therein). For this type of systems, few stabilization methods have been developed either with state feedback controllers or output feedback controllers, especially in strong stabilization analysis.

Strong stabilization, which is to design a stable stabilizing feedback controller for the given plant, is of great importance in the physical implementation of the control since unstable controllers may lead to unpredictable results in case of sensor faults and plant uncertainties/nonlinearities. The strong stabilization problem for linear delay-free systems has been studied in various frameworks (See (Cao and Lam, 2000), (Feintuch, 2008), (Vidyasagar, 1985)). For linear time-invariant systems, necessary and sufficient conditions shown in (Youla et al., 1974) for the existence of a stable stabilizing controller says that a rational plant is strongly stabilizable if and only if its number of unstable poles (counted according with their McMillan degrees) between every pair of right-half

plane blocking zeros is even. Approaches utilizing \mathcal{H}_2/LQG optimal control theory have been suggested subsequently to modify the cost function and Kalman filtering Riccati equation in order to guarantee the stability of the optimal controller. Campos-Delgado and Zhou (Campos-Delgado and Zhou, 2001) converted the stable \mathcal{H}_∞ design problem into a 2-block standard \mathcal{H}_∞ problem via the parametrization of all suboptimal \mathcal{H}_∞ controllers, and reduced higher order controller designed in (Zeren, 1997) by a two-step reduction algorithm. Yoon and Kimura (Yoon and Kimura, 2006) presented a topological result on the robustness of nonstrong stabilizability and obtained two classes of nonstrongly stabilizable systems. However, little attention has been paid toward this issue for input-delayed systems since the stability constraints on the controllers are very hard to reflect on the cost functions and more difficult to implement than those without the strong stabilization requirement.

Related to the above remarks, a natural question to ask is how to design a dynamic output feedback (DOF) controller to strongly stabilize a system with an unknown input delay. This paper discusses in detail the output feedback stabilization problem for linear input-delayed systems using a new approach in the state space. A new stability condition of static output feedback (SOF) stabilization in terms of matrix inequalities is proposed first in Section 3. Advan-

tages of such a characterization is twofold. First, the decoupling of the input and the gain-output matrix enables us to parameterize the controller matrix by a free matrix parameter. Second, the separation of the Lyapunov matrix and the controller matrix avoids imposing any constraint on the Lyapunov matrix when the controller matrix is parameterized. With the aid of the free-weighting matrix introduced and the separation of the Lyapunov matrix and the controller matrix, in Sections 4 and 5, delay-independent DOF strong stabilization of a general dynamic controller is realized through an iterative algorithm. The effectiveness and merits of the proposed approach are shown in Section 6 through numerical examples in the end of the paper.

2 NOTATION AND PRELIMINARIES

Notation: Throughout this paper, for real symmetric matrices X and Y , the notation $X \geq Y$ (respectively, $X > Y$) means that the matrix $X - Y$ is positive semi-definite (respectively, positive definite). 0 in a matrix inequality is a null matrix with an appropriate dimension. The superscript “ T ” represents the transpose of the matrix and the asterisk “ $*$ ” in a matrix stands the term which is induced by symmetry. $\text{col}\{\cdot\}$ denotes a matrix column with blocks given by the matrices in $\{\cdot\}$. A block diagonal matrix with diagonal blocks A_1, A_2, \dots, A_r will be denoted by $\text{diag}\{A_1, A_2, \dots, A_r\}$. Matrices, if their dimensions are not explicitly stated, are assumed to have compatible dimensions for algebraic operations.

Consider the following linear time-invariant system with delayed and non-delayed inputs,

$$(\Sigma) : \begin{aligned} \dot{x}(t) &= Ax(t) + B_0u(t) + B_1u(t-d) \\ y(t) &= Cx(t) \end{aligned}$$

where $x(t) \in \mathbb{R}^n$ is the state with the initial function $\phi(t)$ when $t \in [-d, 0]$, and $y(t)$ is the measurement output. Here, A, B_0, B_1, C are the system state, the control input and the measured output matrices, respectively, and $d > 0$ is an unknown constant input delay.

The following lemma is needed in the paper.

Lemma 1 (Finsler’s Lemma). *Consider real matrices F and Ω such that $\Omega = \Omega^T$ and F has full row rank. F^\perp is the orthogonal complement of F which is (possibly non-unique) defined as the matrix with maximum column rank satisfying $FF^\perp = 0$ and $F^{\perp T}F^\perp > 0$. Then the following statements are equivalent:*

1. *There exists a vector $\xi(t) \in \mathbb{R}^n$ such that $\xi^T(t)\Omega\xi(t) < 0$ and $F\xi(t) = 0$;*

2. *There exists a scalar $\mu \in \mathbb{R}$ such that $\Omega + \mu F^T F < 0$;*
3. *The following condition holds: $F^{\perp T}\Omega F^\perp < 0$.*

3 SOF STABILITY ANALYSIS

An SOF controller under consideration is of the form,

$$(C_1) : u(t) = Ky(t)$$

where K is the controller gain to be designed. When SOF controller (C_1) is applied to (Σ) , the closed-loop system is

$$(\Sigma_{c1}) : \dot{x}(t) = (A + B_0KC)x(t) + B_1KCx(t-d)$$

The following delay-independent criterion utilizes a free matrix $P_2 > 0$ to describe the stabilizability of system (Σ) associated with controller (C_1) in a special form.

Theorem 1. *The closed-loop system (Σ_{c1}) is asymptotically stable, if there exist matrices $P_1 > 0, P_2 > 0, S > 0$, and K such that*

$$\Upsilon_1 = \begin{bmatrix} \mathbf{P}^T \mathbf{A} + \mathbf{A}^T \mathbf{P} + \mathbf{S}_1 & \mathbf{P}^T \mathbf{B}_1 \\ \mathbf{B}_1^T \mathbf{P} & \mathbf{S}_2 \end{bmatrix} < 0 \quad (1)$$

where $\mathbf{A} = \begin{bmatrix} A & B_0 \\ KC & -I \end{bmatrix}, \mathbf{B}_1 = \begin{bmatrix} 0 & B_1 \\ 0 & 0 \end{bmatrix}, \mathbf{S}_1 = \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix}, \mathbf{S}_2 = \begin{bmatrix} -S - C^T K^T P_2 KC & C^T K^T P_2 \\ P_2 KC & -P_2 \end{bmatrix},$
and $\mathbf{P} = \begin{bmatrix} P_1 & 0 \\ -P_2 KC & P_2 \end{bmatrix}.$

Proof: System (Σ_{c1}) is asymptotically stable (Gu et al., 2003) if there exist matrices $P_1 > 0$ such that

$$\Upsilon_2 \triangleq \begin{bmatrix} S + P_1(A + B_0KC) & P_1 B_1 KC \\ +(A + B_0KC)^T P_1 & -S \end{bmatrix} < 0 \quad (2)$$

In the following, we establish the equivalence of (1) and (2).

(Sufficiency) By pre- and post multiplying (1) by $\begin{bmatrix} S_1^T & 0 \\ 0 & S_1^T \end{bmatrix}$ and $\begin{bmatrix} S_1 & 0 \\ 0 & S_1 \end{bmatrix}$ with $S_1 = \begin{bmatrix} I \\ KC \end{bmatrix}$, respectively, we have

$$\begin{bmatrix} S_1^T \mathbf{P}^T \mathbf{A} S_1 + S_1^T \mathbf{A}^T \mathbf{P} S_1 + S_1^T S_1 & S_1^T \mathbf{P}^T \mathbf{B}_1 S_1 \\ * & S_1^T \mathbf{S}_2 S_1 \end{bmatrix} < 0 \quad (3)$$

and in that

$$\begin{aligned} & \begin{bmatrix} I \\ KC \end{bmatrix}^T \begin{bmatrix} P_1 & -C^T K^T P_2 \\ 0 & P_2 \end{bmatrix} \begin{bmatrix} A & B_0 \\ KC & -I \end{bmatrix} \begin{bmatrix} I \\ KC \end{bmatrix} \\ &= \begin{bmatrix} P_1 & 0 \end{bmatrix} \begin{bmatrix} A + B_0 KC \\ 0 \end{bmatrix} = P_1(A + B_0 KC) \\ & \begin{bmatrix} I \\ KC \end{bmatrix}^T \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} I \\ KC \end{bmatrix} = S \end{aligned}$$

$$\begin{aligned} & \begin{bmatrix} I \\ KC \end{bmatrix}^T \begin{bmatrix} P_1 & -C^T K^T P_2 \\ 0 & P_2 \end{bmatrix} \begin{bmatrix} 0 & B_1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} I \\ KC \end{bmatrix} \\ & = P_1 B_1 K C \\ & \begin{bmatrix} I \\ KC \end{bmatrix}^T \begin{bmatrix} -S - C^T K^T P_2 K C & C^T K^T P_2 \\ P_2 K C & -P_2 \end{bmatrix} \begin{bmatrix} I \\ KC \end{bmatrix} \\ & = -S \end{aligned}$$

Thus inequality (3) is in fact (2).

(Necessity) Assume that (2) holds. There must be a sufficiently large matrix $P_2 > 0$, such that

$$\begin{aligned} & - \begin{bmatrix} B_0^T \\ B_1^T \end{bmatrix} \begin{bmatrix} P_1 & 0 \end{bmatrix} \Upsilon_2^{-1} \begin{bmatrix} P_1 \\ 0 \end{bmatrix} \begin{bmatrix} B_0 & B_1 \end{bmatrix} \\ & - \begin{bmatrix} 2P_2 & 0 \\ * & P_2 \end{bmatrix} < 0 \end{aligned}$$

Then straightforward manipulation and Schur complement equivalence yields that

$$\begin{aligned} & T_1^T \begin{bmatrix} \Upsilon_2 & \begin{bmatrix} P_1 B_0 \\ 0 \\ -2P_2 \end{bmatrix} & \begin{bmatrix} P_1 B_1 \\ 0 \\ 0 \\ -P_2 \end{bmatrix} \\ * & * & * \\ * & * & * \end{bmatrix} T_1 \\ & = \begin{bmatrix} \bar{S}_1^T & 0 \\ * & \bar{S}_1^T \end{bmatrix} \Upsilon_1 \begin{bmatrix} \bar{S}_1 & 0 \\ * & \bar{S}_1 \end{bmatrix} < 0 \end{aligned}$$

$$\text{where } T_1 = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \text{ and } \bar{S}_1 = \begin{bmatrix} I & 0 \\ KC & I \end{bmatrix}$$

are nonsingular matrices. This completes the proof. \square

Remark 1. *Theorem 1 provides an equivalent form to design a static output feedback controller for the input-delayed systems. The advantage of Theorem 1 lies in not only the separation of B_0 , B_1 and KC , but also in the separation of Lyapunov matrix P_1 and the controller matrix K . This feature enables us to parametrize K by a tuning matrix $P_2 > 0$, independent of the Lyapunov matrix used for checking stability or performances directly. Therefore, less conservative results will be obtained, comparing with previous approaches, since no additional constraints induced to deal with the nonconvex terms of the Lyapunov matrix and the controller matrix when it is parametrized.*

Remark 2. *Intuitively, if d is known, the stabilization problem of systems $\dot{x}(t) = Ax(t) + Bu(t)$ and $y(t) = Cx(t)$ can be treated by a delayed output feedback controller $u(t) = K[y(t) + Zy(t-d)]$, where Z is a tuning matrix satisfying $KZ = ZK$, in the sense that the system $\dot{x}(t) = Ax(t) + Bu(t) + BZu(t-d)$ can be stabilized by an SOF controller $u(t) = Ky(t)$.*

4 DOF STRONG STABILIZATION

Consider a general form of dynamic controller as follows:

$$\begin{aligned} (C') : \quad \dot{\vartheta}(t) &= K_A \vartheta(t) + K_B y(t) \\ u(t) &= K_c \vartheta(t) + K_D y(t) \end{aligned}$$

The input-delayed system (Σ) with controller (C') gives the following closed-loop system ($\Sigma_{c'}$):

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\vartheta}(t) \end{bmatrix} = (\hat{A} + \hat{B}_0 \hat{K} \hat{C}) \begin{bmatrix} x(t) \\ \vartheta(t) \end{bmatrix} + \hat{B}_1 \hat{K} \hat{C} \begin{bmatrix} x(t-d) \\ \vartheta(t-d) \end{bmatrix}$$

$$\text{where } \hat{A} = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix},$$

$$\hat{B}_0 = \begin{bmatrix} 0 & B_0 \\ I & 0 \end{bmatrix}, \hat{B}_1 = \begin{bmatrix} 0 & B_1 \\ 0 & 0 \end{bmatrix}, \hat{C} = \begin{bmatrix} 0 & I \\ C & 0 \end{bmatrix}$$

and $\hat{K} = \begin{bmatrix} K_A & K_B \\ K_C & K_D \end{bmatrix}$ is the controller gain matrix to be determined. Only the system data are involved in the above shorthands, and $\hat{A} + \hat{B}_0 \hat{K} \hat{C}$ and $\hat{B}_1 \hat{K} \hat{C}$ are affine in the controller gain \hat{K} .

The problem of strong stabilization is regarded as searching one or more common positive definite matrices to guarantee the stability of both the closed-loop system ($\Sigma_{c'}$) and its stabilizing controller (C'). For (C'), it is asymptotically stable if and only if there exists a matrix $S_c > 0$ such that $S_c K_A + K_A^T S_c < 0$. A delay-independent criterion utilizing the free matrix $P_2 > 0$ to describe the strong stabilizability of system (Σ) associated with controller (C') is given as follows.

Theorem 2. *Controller (C') strongly stabilizes (Σ) if there exist matrices $P_1 > 0$, $S > 0$, $P_2 = \text{diag}\{P_{21}, P_{22}\} > 0$, L, N satisfying*

$$\begin{aligned} \Phi(N) \triangleq & \begin{bmatrix} P_1 \hat{A} + \hat{A}^T P_1 & P_1 \hat{B}_0 + 2\hat{C}^T L^T \\ +S + 2M & -2P_2 \\ * & * \\ * & * \\ & 0 & P_1 \hat{B}_1 \\ & 0 & 0 \\ -S + M & \hat{C}^T L^T \\ * & -P_2 \end{bmatrix} < 0 \quad (4) \\ & \begin{bmatrix} 0 & I \end{bmatrix} (L + L^T) \begin{bmatrix} 0 \\ I \end{bmatrix} < 0 \quad (5) \end{aligned}$$

where $M = -N^T L \hat{C} - \hat{C}^T L^T N + N^T P_2 N$. Under the above conditions, the gain matrix of a stabilizing controller (C') can be parametrized as $\hat{K} = P_2^{-1} L$.

Proof: Expanding inequality (1), with $A, B_0, B_1, C,$

K replaced by \hat{A} , \hat{B}_0 , \hat{B}_1 , \hat{C} , \hat{K} , yields that

$$\begin{bmatrix} P_1\hat{A} + \hat{A}^T P_1 + S & P_1\hat{B}_0 + 2\hat{C}^T \hat{K}^T P_2 \\ -2\hat{C}^T \hat{K}^T P_2 \hat{K} \hat{C} & P_1\hat{B}_1 + 2\hat{C}^T \hat{K}^T P_2 \\ * & -2P_2 \\ * & * \\ * & * \\ 0 & P_1\hat{B}_1 \\ 0 & 0 \\ -S - \hat{C}^T \hat{K}^T P_2 \hat{K} \hat{C} & \hat{C}^T \hat{K}^T P_2 \\ * & -P_2 \end{bmatrix} < 0 \quad (6)$$

Here, it suffices to prove that the feasibility of (6) is equivalent to that of (4) in terms of their respective variables.

(Sufficiency) Assume (4) holds. It follows that $P_2 > 0$, and let that $\hat{K} = P_2^{-1}L$ is well defined, and $L = P_2\hat{K}$. Substituting it into (4) and noting, for any real matrix N with appropriate dimension,

$$(N - \hat{K}\hat{C})^T P_2 (N - \hat{K}\hat{C}) \geq 0$$

we have (6) holds with property that all the terms $-\hat{C}^T \hat{K}^T P_2 \hat{K} \hat{C}$ in the diagonal.

$$-\hat{C}^T \hat{K}^T P_2 \hat{K} \hat{C} \leq N^T P_2 N - N^T L \hat{C} - \hat{C}^T L^T N = M$$

(Necessity) Assume (6) holds. Then, by setting $N = \hat{K}\hat{C}$, we obtain

$$\begin{aligned} & -\hat{C}^T \hat{K}^T P_2 \hat{K} \hat{C} \\ &= -\hat{C}^T \hat{K}^T P_2 \hat{K} \hat{C} + (N - \hat{K}\hat{C})^T P_2 (N - \hat{K}\hat{C}) \\ &= -N^T P_2 \hat{K} \hat{C} - \hat{C}^T \hat{K}^T P_2 N + N^T P_2 N \end{aligned}$$

Substituting it into (6), and denoting $L = P_2\hat{K}$, (4) is obtained.

Due to $P_2 = \text{diag}\{P_{21}, P_{22}\} > 0$, from $L = P_2\hat{K} = \begin{bmatrix} P_{21}K_D & P_{21}K_C \\ P_{22}K_B & P_{22}K_A \end{bmatrix}$, the inequality (5) is used to ensure the matrix K_A stable, meaning the stability of the controller (C'). This completes the proof. \square

Remark 3. It is worth pointing out that the parametrization of the controller matrices by our approach is fairly flexible. Indeed, the parametrization of the strongly stabilizing controller (C') mainly depends on the free parameter P_2 , which can be set to be any positive definite matrix without loss of generality. Thus more synthesis problems such as simultaneous stabilization, structural controller synthesis can be treated readily in the same framework.

5 PARAMETRIZATION DESIGN OF CONTROLLER

We are now in a position to design controller gains via an effective algorithm. When N is fixed, (4) becomes

a strict LMI problem, which can be verified easily by conventional LMI solver. The remaining problem is how to select the matrix N . It can be seen from the proof of Theorem 2 that the left hand side of (4), $\Phi(N)$ achieves its minimum when $N = P_2^{-1}L\hat{C}$, which can be used to construct an iteration rule. We summarize briefly our analysis on N in the following proposition.

When $P_1 > 0$, $P_2 > 0$, S , L are fixed, the following relationship holds for any real matrix N ,

$$\Phi(P_2^{-1}L\hat{C}) \leq \Phi(N)$$

It follows that the scalar ε satisfying $\Phi(N) < \varepsilon I$ achieves its global minimum only if $N = P_2^{-1}L\hat{C} = \hat{K}\hat{C}$. Therefore, the following iteration algorithm is constructed to solve the condition of Theorem 2.

Algorithm OFSS (Output Feedback Strong Stabilization):

- Step 1. Set $m = 1$, and $\varepsilon_0^* > 0$, $c > 0$ be three prescribed initial values. Select an initial matrix N_1 such that the closed-loop system ($\Sigma_{c'}$), where $\hat{K}\hat{C}$ is substituted by N_1 , is stable.
- Step 2. For the fixed N_m , solve the following convex optimization problem with respect to L_m , $P_{1m} > 0$, $P_{2m} > 0$, $S_m > 0$:

$$\begin{aligned} & \min \quad \varepsilon_m \\ & \text{s.t.} \quad \Phi(N_m) < \varepsilon_m I \\ & \quad \quad \varepsilon_m > -c \end{aligned} \quad (7)$$

where $\Phi(N_m)$ is the function $\Phi(N)$ defined in (4). Denote ε_m^* as the minimized value of ε_m satisfying (7). If $\varepsilon_m^* \leq 0$, the system (Σ) is stabilizable via the DOF controller (C'). The gain matrix \hat{K} of (C') can be obtained as $\hat{K} = P_{2m}^{-1}L_m$, STOP, else, go to next step.

- Step 3. If $|\varepsilon_m^* - \varepsilon_{m-1}^*| \leq \delta$, a prescribed tolerance, then go to Step 4, else update N_{m+1} as

$$N_{m+1} = (P_{2m})^{-1}L_m\hat{C}$$

and set $m = m + 1$, then go to Step 2.

- Step 4. The system may not be strongly stabilizable via the controller (C'). STOP. (Or choose another initial value N_1 , then run the algorithm again.)

Remark 4. It follows from that the sequence $\{\varepsilon_m^*\}$ is monotonic decreasing with respect to m and has a lower bound c . Therefore, the stopping of the iteration is guaranteed.

Remark 5. The initial value of N_1 can be considered as a state feedback stabilizing controller matrix, which can be found by existing stabilization approaches. Like many other iteration algorithms, the

sequence of iterations depends on the selection of initial values, and appropriate selection will improve the solvability. Here, we attempt to get a relaxing state feedback controller N_1 which just satisfy $\hat{A} + \hat{B}_0$ or $\hat{A} + (\hat{B}_0 + \hat{B}_1)N$ stable, as the initial value N_1 for system (Σ_c) . There has much conservatism since it is only a delay-independent approximative solution. If failed, Zhang et.al in (Zhang et al., 2005) gave a further method to obtain a new state- and input-delay-dependent state feedback controller to ensure the stability of the closed-loop system. The numerical examples in the following section will illustrate that Algorithm OFSS is relaxed to rely on the initial matrix N_1 .

Remark 6. The approach proposed in the paper is in fact not a conservative one. The direct iterative procedure (D-K iteration) may generate a feasible solution. However, the success rate may be low. As is well known, even for LTI systems without delay, the DOF controller design is a non-convex problem, and is likely to be NP-hard. To cope with a nonconvex problem via convex approach, there are generally two recipes. One is the so called relaxation, and the other is the local optimization. The relaxation approach is easy to implement, but may introduce conservatism in some cases. LMI approaches can be regarded as one kind of relaxation. For the local optimization, one wants to seek a point that is only locally optimal, which means that it minimizes the objective function among feasible points that are near it. Therefore, the initial values are critical to such optimization problems, and good initial values may generate a globally optimal solution. Most exact approaches to DOF synthesis, including CCL, ILMI, alternating projection, D-K iteration, nonsmooth optimization for instance, involve local optimization. However, few approaches have systematic procedures to even determine an initial value. Obviously, finding an initial stabilizing state-feedback gain is more desirable than guessing a stabilizing DOF one. In this sense, the selection of initial values in this paper is more desirable than a direct iterative procedure (D-K iteration). In fact, as we have shown in the proposition, a globally optimal solution of conditions (4) and (5) is obtained only if N is a stabilizing state-feedback gain, which means that our iteration begins with a set of necessary N for the matrix inequalities conditions (4) and (5) to be feasible rather than random guesses.

6 NUMERICAL EXAMPLE

This section presents a numerical example to demonstrate the validity of the proposed method in this pa-

per to design a DOF strong stabilization controller. Consider a linear input-delayed system (Σ) with the parameters as follows:

$$A = \begin{bmatrix} 0.9926 & 0.1443 \\ 0 & -0.3698 \end{bmatrix}, B_0 = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, B_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

The input delay d is constant and it has a particular form with $C = I$. Now we apply the proposed approach to find DOF controllers to stabilize this system. An initial matrix N_1 is chosen for DOF controller (C') which is obtained directly by solving state stabilization conditions for a system pair (\hat{A}, \hat{B}_0) defined in (Σ_c) , $\hat{A}X + \hat{B}_0Y + (\hat{A}X + \hat{B}_0Y)^T < 0$ and $X > 0$, with setting $N_1 = YX^{-1}$. Two cases are considered as follows with $\epsilon_0^* = 10$:

- a. Full order DOF controller

$$N_1 = \begin{bmatrix} 0.0013 & 0.0066 & -0.4916 & 0.0038 \\ 0.0091 & 0.0068 & 0.0052 & -0.4918 \\ 1.4990 & 0.1517 & 0.0015 & 0.0017 \end{bmatrix}$$

is chosen as the initial matrix in Algorithm OFSS. After 1 iteration, a desired strong DOF controller (C') is obtained as

$$\begin{cases} \dot{\vartheta}(t) = \begin{bmatrix} -0.8119 & 0.0034 \\ 0.0046 & -0.8125 \end{bmatrix} \vartheta(t) \\ \quad + \begin{bmatrix} -0.0068 & 0.0112 \\ 0.0089 & 0.0112 \end{bmatrix} y(t) \\ u(t) = \begin{bmatrix} -0.0021 & 0.0003 \end{bmatrix} \vartheta(t) \\ \quad + \begin{bmatrix} 1.4883 & 0.1834 \end{bmatrix} y(t) \end{cases}$$

The eigenvalues of the controller matrix K_A are -0.8082 and -0.8162 .

- b. Lower order DOF controller

$N_1 = \begin{bmatrix} 0.0057 & 0.0086 & - \\ 0.4982 & 1.5019 & 0.1460 & 0.0024 \end{bmatrix}$ is chosen as initial matrix, and a desired strong DOF controller (C') is obtained after 1 iteration,

$$\begin{cases} \dot{\vartheta}(t) = -0.8030\vartheta(t) + \begin{bmatrix} 0.0012 & 0.0145 \end{bmatrix} y(t) \\ u(t) = -0.0012\vartheta(t) + \begin{bmatrix} 1.4763 & 0.1784 \end{bmatrix} y(t) \end{cases}$$

Furthermore, consider the same model with a different output matrix $C = [0.9556 \ 0.1132]$. With the same method to calculate initial matrix N_1 as the above model, two kinds of DOF stabilizing controllers are given by applying Algorithm OFSS again with 1 iteration.

- a. Full order DOF controller

$$N_1 = \begin{bmatrix} 0.0019 & 0.0010 & -0.4929 & 0.0086 \\ 0.0010 & 0.0020 & 0.0089 & -0.4932 \\ 1.4995 & 0.1504 & 0.0060 & 0.0078 \end{bmatrix}$$

$$\begin{cases} \dot{\vartheta}(t) = \begin{bmatrix} -0.7880 & 0.0416 \\ 0.0389 & -0.7743 \end{bmatrix} \vartheta(t) \\ \quad + \begin{bmatrix} 0.0257 \\ 0.0379 \end{bmatrix} y(t) \\ u(t) = \begin{bmatrix} 0.0010 & 0.0024 \end{bmatrix} \vartheta(t) + 1.5792y(t) \end{cases}$$

The eigenvalues of the controller matrix K_A are -0.8220 and -0.7403 .

- b. Lower order DOF controller

$$N_1 = \begin{bmatrix} 0.0054 & 0.0020 & -0.4944 \\ 1.4926 & 0.1487 & 0.0039 \end{bmatrix}$$

$$\begin{cases} \dot{\vartheta}(t) = -0.7926\vartheta(t) + 0.0289y(t) \\ u(t) = 0.0054\vartheta(t) + 1.5683y(t) \end{cases}$$

It is known from the above computational cases that the algorithm converges to the feasible solutions quickly for the arbitrarily chosen of initial matrix N_1 very much while designing any order strong DOF controllers.

7 CONCLUSIONS

This paper has developed the strong output feedback control problem for an input-delayed system from a new perspective. Input-delay-independent stabilization criteria for output feedback controllers are derived from a new equivalent characterization on stabilizability of the system in terms of matrix inequalities by introducing a slack positive definite matrix, and an iterative algorithm is developed to solve these conditions. Although common to other approaches, the proposed approach is not guaranteed to find a solution even it exists, it is quite effective since there is no need to introduce additional constraints to linearize the product term of Lyapunov matrix and controller gain when parametrized.

ACKNOWLEDGEMENTS

The research was supported by GRF HKU 7031/09E.

REFERENCES

- Campos-Delgado, D. U. and Zhou, K. (2001). H_∞ strong stabilization. *IEEE Trans. Autom. Control*, 46(12):1968–1972.
- Cao, Y. and Lam, J. (2000). On simultaneous H_∞ control and strong H_∞ stabilizations. *Automatica*, 36(6):859–865.

- Chen, W. and Zheng, W. (2006). On improved robust stabilization of uncertain systems with unknown input delay. *Automatica*, 42:1067–1072.
- Feintuch, A. (2008). On strong stabilization for linear continuous time time-varying systems. *Systems & Control Letters*, 57:691–695.
- Gu, K., Kharitonov, V., and Chen, J. (2003). *Stability of Time-Delay Systems*. Birkhauser, Boston, MA.
- Respondek, J. S. (2008). Approximate controllability of the n^{th} order infinite dimensional systems with controls delayed by the control devices. *Int. J. Syst. Sci.*, 39(8):765–782.
- Tadmor, G. (2000). The standard H_∞ problem in systems with a single input delay. *IEEE Trans. Autom. Control*, 45(3):382–397.
- Vidyasagar, M. (1985). *Control System Synthesis: A Factorization Approach*. MIT Press, Cambridge, MA.
- Yoon, M. G. and Kimura, H. (2006). A topological result on strong stabilization problem. *IEEE Trans. Autom. Control*, 51(4):657–661.
- Youla, D. C., Bongiorno, J. J., and Lu, N. C. (1974). Single-loop feedback stabilization of linear multivariable dynamical plants. *Automatica*, 10(2):159–173.
- Zeren, M. (1997). On stable H_∞ controller design. In *Proceedings of American Control Conference*, pages 1302–1306, Albuquerque, NM.
- Zhang, X., Wu, M., She, J., and He, Y. (2005). Delay-dependent stabilization of linear systems with time-varying state and input delays. *Automatica*, 41(8):1405–1412.

SMART HOME

From User's Behavior to Prediction of Energy Consumption

Lamis Hawarah, Mireille Jacomino and Stephane Ploix

Laboratoire G-SCOP, INP Grenoble, UJF, CNRS, 46 Avenue Felix Viallet, 38031 Grenoble, France

{lamis.hawarah, mireille.jacomino, stephane.ploix}@g-scop.inpg.fr

Keywords: Energy consumption, User's behavior, Bayesian network, Learning system.

Abstract: This paper concerns a home automation system of energy management. Such a system aims at keeping under control the energy consumption in housing. The expected energy consumption is scheduled over one day. Each hour a total amount of energy is available that is a resource constraint for the expected energy plan. The expected consumption is totally derived from users behavior which are quite different from one housing to another, and rather difficult to predict. This paper proposes a *Learning System* to predict the user's requests of energy. The proposed method relies on Bayesian networks.

1 INTRODUCTION

A *Home Automation System* basically consists of household appliances connected by both energy and communication networks. Smart Home and more generally Smart Building are spreading out. They aim at first increasing comfort and security, second enabling remote access to information about the appliances and the buildings and third managing the appliances. The system addressed in this paper is concerned with energy management in Smart Home. It aims at planning the best energy assignment satisfying the availability energy constraints and the users' requests (Palensky et al., 1997), (HA et al., 2006). In this paper, energy is restricted to the electric consumption. (HA et al., 2006), (Abrás et al., 2007) present a three-layers household energy control system: anticipative layer, reactive layer and device layer. The anticipative layer depicted in (Ha, 2005) and (Abrás et al., 2008) is mainly concerned with the energy plan. The anticipative plan relies on predictions of environmental parameters (weather forecast, solar radiation, ...) and energy consumption.

In order to keep under control the total amount of consumed energy every hour, and then avoid peak consumptions and minimize the energy cost, the *Home Automation System* has to schedule as much as possible the energy consumptions in the most appropriate time periods. For example, the washing machine could be planned before or after the oven in a low energy cost period as far as such a plan satisfies the predicted user's request. The efficiency of

the anticipated plan is as good as the prediction of the user's request. Indeed if the actual user's behavior is far from the predicted one, then the reactive layer has to stop an appliance in order to satisfy the available energy constraint for example, and schedule this appliance later without any energy cost optimization.

This paper focuses on the prediction of the user's behavior. A *Learning System* is proposed to predict the inhabitant's requests for each hour of a 24 hours anticipative time period. The system is based on the use of Bayesian Network to predict the user's behavior. Bayesian Networks (BNs) are a field of Machine Learning, capable of representing and manipulating arbitrary probability distributions over arbitrary random variables (Russell and Norvig, 2003), (Naïm et al., 2004). They are especially well suited for modeling uncertain knowledge in expert systems (Heckerman, 1995). The paper is organized as follows: first, related works concerning the use of household appliances are presented. The next section shows the way how a Bayesian Network can be used. The proposed approach to predict the user's behavior in housing is explained in section 2. A real database concerning 100 houses in France is used to build standard profiles from which the *Learning System* deduces the predicted user's behavior. Finally, some results and perspectives are discussed.

1.1 Related Works

Various works have been done to study the impact of the user's behavior on the energy consumption in

the housings. (Wood and Newborough, 2003), (Wood and Newborough, 2007) study the interaction between the user and the appliances. The appliances are classified into four categories according to their level of automation and the number of settings. For the appliances with low level of automation the user needs to be in the proximity of the appliances to be set. They achieved up to 10~20% reduction in energy consumption of households by changing the user's behavior. Other studies are interested in modeling and simulating the activities of the user (Zimmerman, 2007). The activities of one and several people are integrated into simulators of buildings performances to get more use dependent results. This approach models all users as individual agents with different behaviors. Different functions and functional units such as work places are modelled also. The main results of this work is that the activities of individuals and groups in office environments can be modelled on the basis of communicating agents. (Ha et al., 2006) studies and analyzes the user's behavior in housing to make embodiment of user's interface in Ubiquitous environment. Behavior patterns are analyzed by classifying data into 5W(who, what, where, when and why) and 1H (How). All these works focuses on the design of displays in order to change the user's behavior.

1.2 Bayesian Networks

A Bayesian Network (BN) is a graphical model for probabilistic relationships among a set of variables (Pearl, 1986). BNs model causal relationships. They are represented as directed acyclic graphs, where each node represents a different random variable. A directed edge from the node X (*cause node*) to the node Y (*effect node*) indicates that X has a direct influence on Y. This influence is quantified by the conditional probability $P(Y|X)$, stored at node Y. The nodes in a network can be of two types: *evidence node* when its value is observed, and *query node* when its value has to be predicted. A Conditional Probability Table (CPT) is assigned to each node in the network. Such probabilities may be set by an expert or using a registered data. BNs are based on the conditional independence; each node is conditionally independent of its non-descendants given its parents. When a node has no parent, its CPT specifies the prior probability. There are two types of learning: 1) *the structure learning* in which the best graph representing the problem is searched; 2) *the parametric learning* in which the structure of the network is known and the conditional probability is estimated at each node. Once the Bayesian Network is constructed, it

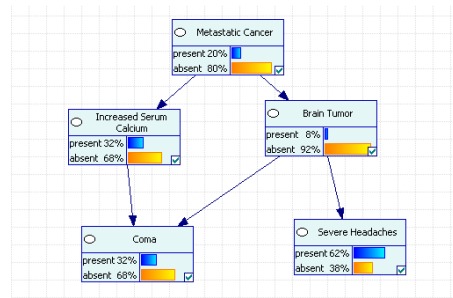


Figure 1: Bayesian Network for Coma problem.

can be used to compute the probability distribution for a query variable, given a set of evidence variables. This operation is called *inference*. For example, one can identify the causes by calculating the most probable cause given some information (Figure 1), or predict the effects of a cause by calculating the most frequent value of a node given some observations. Exact and approximate approaches of inference can be used (Russell and Norvig, 2003), (Naïm et al., 2004).

Bayesian Networks are used in a large range of applications: telecommunications (Ezawa and Schuermann, 1995), display management for time-critical decisions (Horvitz and Barry, 1995), industry (Hart and Graham, 1997), health (Becker et al., 1998), communication (Barco et al., 2002), etc.

2 PROBLEM STATEMENT

The objective of the work presented in this paper is to propose a *Learning System* able to deliver to the *Home Automation System* the useful information about the energy consumption in a given housing. This useful information is the prediction of the user's requests. A preliminary step consists in identifying a set of standard profiles of the users' requests. Then the *Learning System* has two tasks: 1) First, identify the most appropriate standard profile for a given user in order to exhibit the corresponding prediction of the energy consumption at each hour; 2) Second, built the learnt model of the energy consumption. The standard profiles are defined for each appliance and associated service to the inhabitants. They are built using a database. A questionnaire is used to identify the most appropriate standard profile for a given user. The questions concern the appliances in housing, the frequency of their use, etc.

The proposed method is depicted in the figure2. After the standard starting profile has been chosen for each service it is integrated in a Bayesian Network which will predict the actual user's requests. The actual energy consumption of the user is sent

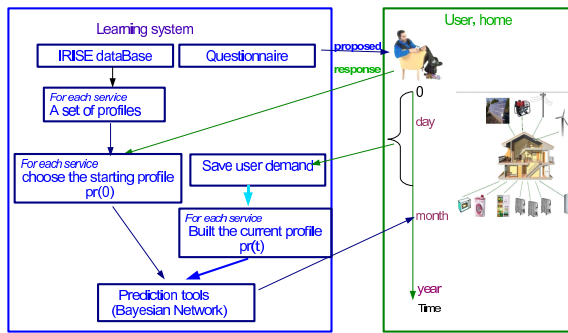


Figure 2: Structure of the Learning System.

to the *Learning System* also. This observation concerns the actual consumption of the appliances, the date, the hour, the consumed energy and the duration. After some time of observation, the *Learning System* can build the new profile dedicated to the actual requests of the user. In this paper, the process to identify the standard profiles and the Bayesian model of the *Learning System* are described.

3 BUILDING THE STANDARD PROFILES

3.1 Energy Database

The project *Residential Monitoring to Decrease Energy Use and Carbon Emissions in Europe (REMODOECE)*¹ provides an energy database. This is a European database on residential consumption, including Central and Eastern European Countries, as well as new European Countries (Bulgaria and Romania).

This database stores the characteristics of the residential electric consumption by country. The *IRISE* project is a part of *REMODOECE*. It has been chosen for our study to identify the standard profiles $pr(0)$. It deals only with houses in France. One database is available for every house; in such a database, the information is recorded every 10 minutes during one year for each appliance in the house. The consumed energy at every time period by every appliance is given in this database. However, these data have to be processed before using.

It is possible to know the number of people who live in each house. The presence of the user is important for the energy consumption but it is not explicitly known in the database.

¹<http://www.isr.uc.pt/~remodece/>

Date	Starting Ho	Duration	Energy	Day	Month	Starting numb
1999-09-29	[18-19]	30	748	Wednesday	"September"	1
1999-10-03	[12-13]	40	481	Sunday	"October"	1
1999-10-03	[13-14]	30	273	Sunday	"October"	1
1999-10-03	[18-19]	40	767	Sunday	"October"	1
1999-10-04	[11-12]	60	1453	Monday	"October"	1
1999-10-05	[14-15]	70	1417	Tuesday	"October"	1
1999-10-06	[11-12]	50	697	Wednesday	"October"	1
1999-10-07	[19-20]	30	1058	Thursday	"October"	1
1999-10-08	[11-12]	50	695	Friday	"October"	1
1999-10-10	[19-20]	40	579	Sunday	"October"	1
1999-10-11	[11-12]	60	605	Monday	"October"	1
1999-10-11	[18-19]	30	809	Monday	"October"	1
1999-10-12	[18-19]	50	1123	Tuesday	"October"	1
1999-10-13	[18-19]	40	1088	Wednesday	"October"	1
1999-10-14	[18-19]	40	1005	Thursday	"October"	1
1999-10-15	[18-19]	30	509	Friday	"October"	1
1999-10-16	[11-12]	80	1981	Saturday	"October"	1
1999-10-17	[12-13]	90	2120	Sunday	"October"	1
1999-10-17	[18-19]	40	767	Sunday	"October"	1
1999-10-18	[18-19]	40	502	Monday	"October"	1

Figure 3: The database after treatment for the Electric-oven.

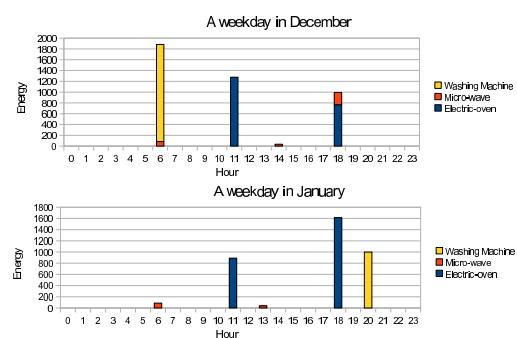


Figure 4: The consumed energy in a weekday in January and in December.

3.2 Data Preprocessing

From the *IRISE* data, a user's request concerns one or more services like *cooking in oven*, *clothe washing*, *water heating*, etc. A standard profile is a structured information derived from these raw data for every service. The *Home Automation System* anticipates the energy consumption from the following information about the user's requests:

- When is the service requested?
- How much energy does the service consume?
- What is the duration of the service?

This information is available in the database except for the duration. The figure 3 shows the preprocessed data for one appliance. Each row is the set of interesting information for one hour in the year: duration, energy, day, month, the number of times that the appliance has been started during the hour is also extracted from the raw data.

Analyzing the data one can notice 1) first the actual diversity of the use of each appliance in a given house and 2) second how difficult it is to characterize this use. The figure 4 shows the mean consumption

of the *Electric-oven*, the *Micro-wave* and the *Washing machine* in a weekday in January and in a weekday in December. The consumed energy is not identical at the same hour in the two months. Nevertheless, some similarities exist. A standard profile gives the probability that the appliance will be started at each hour and the associated expected energy consumption for different types of days statistically representative.

Given only one day, for example Monday, the obtained information is very accurate. But the learning of such an information would be long, because an observation and the derived statistical process could be involved every seven days for one given day in the week. On the other hand, the average value obtained over all the year without any differentiation among the days would not be interesting because the derived prediction would be an average very far from each actual request.

3.3 Statistical Picture

A profile is a statistical picture of a service in a housing. This statistical study is performed over a time period that is the largest period allowing to compute significant probabilities. A time period may be the day, the month or the day in a month. For each service, for a chosen time period, the profile consists of:

- The conditional probability that the service starts every hour;
- The average duration every hour;
- The average consumed energy every hour;

The value of the conditional probability, the duration or the energy for a service are calculated from the preprocessing data.

It can also be useful to calculate the probability that the appliance starts at each hour over all the year without taking into account any time period. This kind of information can be used to briefly depict the profiles and then identify the most appropriate profile to a given user.

Parts of the profile of the *Electric-oven* service taken in example are depicted in the figures 5- 9. These profiles concern the house 2000997 from the *IRISE* database.

The figure 5 represents the probability that the *Electric-oven* starts at each hour for each month over one year, from October to September. For example, the probability that the *Electric-oven* starts at 6 pm in October is 0.41 (figure 6). The figure 7 shows the probability that the *Electric-oven* starts at 6 pm for each weekday. For example, the probability on Sunday at 6 pm is 0.2. The probability that the *Electric-oven* starts at 6 pm in October on Sunday is 0.60 (fig-

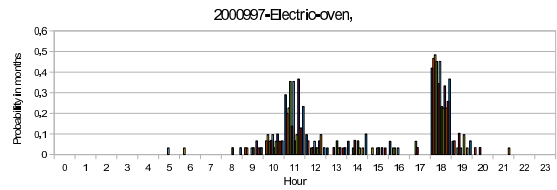


Figure 5: The probability of the Electric-oven in the house 2000997 on months.

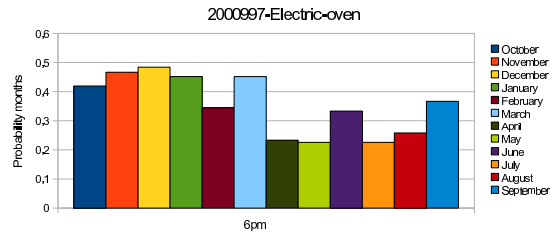


Figure 6: The probability of the Electric-oven in the house 2000997 at 6pm on months.

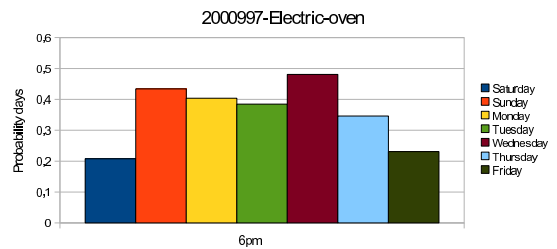


Figure 7: The probability of the Electric-oven in the house 2000997 at 6pm on days.

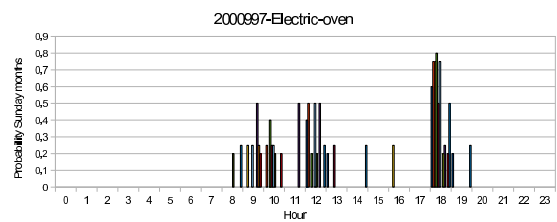


Figure 8: The probability of the Electric-oven in the house 2000997 on Sunday over all the months.

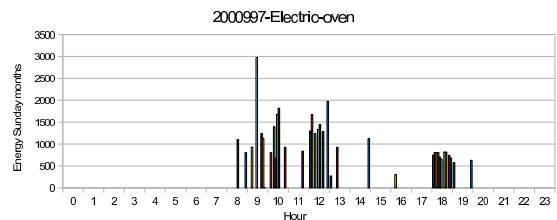


Figure 9: The Energy of the Electric-oven in the house 2000997 on Sunday over all the months.

ure 8). The average energy of the *Electric-oven* at 6 pm in October on Sunday is 751Wh (figure 9). Such profiles can be exhibit for the service duration as well.

4 LEARNING BAYESIAN SYSTEM

The Bayesian Network is used to predict the user's requests. The Conditional Probability Distribution at each node of the BN is computed from both the standard starting profile and the actual observations of energy consumption in the housing. There are two types of nodes in this network: 1) the probabilistic nodes in which a Conditional Probability Table is associated; 2) the deterministic nodes which values are specified exactly by the values of its parents, with no uncertainty (Russell and Norvig, 2003). For the deterministic nodes, the probability distributions are no longer needed to be specified, but instead only certain states. In this work, all the energetic services in the house like the *cooking service* or the *washing service*, etc are represented in the Bayesian Network. There are three causal nodes in the *Learning System*:

- Hour with 24 values from 0 to 23
- Month with 12 values from January to December
- Day with 7 values from Saturday to Friday

All these nodes are probabilistic.

Three nodes are associated to each service in the housing:

- Starting of the service with tow values {yes, no}
- Energy which is a deterministic node
- Duration which is a deterministic node

To simplify the presentation of the network, only the *Electric-oven* is dealt with during two days (Saturday and Sunday) in October. The hour values are reduced to 3 which are {11am, 12am, 1pm}. This network is given in the figure 10.

The Conditional Probability Distribution corresponding to the *Starting Electric-oven* node is part of the profile of the *Electric-oven*. Given the hour, the month and the day, the *Home Automation System* uses this network to obtain the probability of starting, the average energy and the average duration of the services. For example, if the hour is Sunday 12am, the Bayesian Network provides the probability 0.4 of starting for the *Electric-oven*.

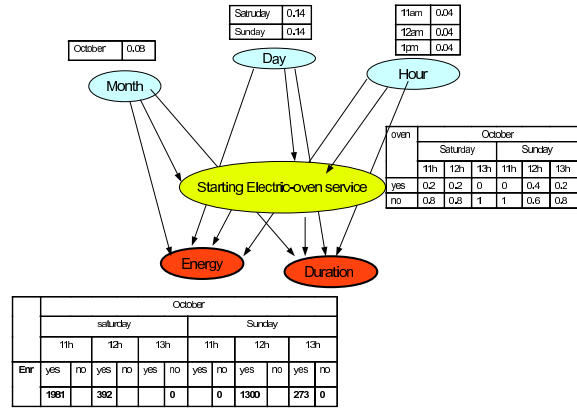


Figure 10: A part of the Bayesian Network.

4.1 Segmentation between Days and Months

In order to exhibit the standard profiles from the database with the best accuracy, the discriminating parameters such as days and months have to be found. For this purpose, a dissimilarity index and a clustering algorithm are defined.

4.1.1 Dissimilarity Index

The probability that a service starts at each hour given the day (figure 7) or the month (figure 5) is used to identify if the days or the months discriminate the service. Comparison between the months all over the year is performed, as well as between the days all over the year. The proposed Dissimilarity index is based on the *Manhattan distance* given in the equation 1. It is used to calculate the difference between two months or two days over 24 hours.

$$Diff(X, Y) = \sum_{i=0}^{23} |x_i - y_i| \quad (1)$$

Where $X=x_i$ is the probability that the service starts in a month (or day) A a each hour i ; $Y=y_i$ is the probability that the service starts in another month (or day) B . The label i represents the hour. Then, the Dissimilarity index is defined as follows:

$$DI(X, Y) = \frac{Diff(X, Y)}{\frac{\sum_i (x_i + y_i)}{2}} \quad (2)$$

$\frac{\sum_i (x_i + y_i)}{2}$ is a normative coefficient from which the Dissimilarity index measures the relative dispersion of the starting probability over the months (or days).

Therefore, when the value of $DI(X, Y)$ is small, X and Y are close together. If $DI(X, Y)$ is large, then

Table 1: Example of the Dissimilarity Index.

Days1	Days2	Diff	$\frac{\sum_i(x_i+y_i)}{2}$	DI	Decision
Sat	Sun	0,79	0,93	0,86	YES
Sat	Mon	0,5	0,72	0,69	NO
Sat	Tues	0,64	0,81	0,79	NO
Sat	Wed	0,83	0,95	0,88	YES
Sat	Thur	0,54	0,78	0,69	NO
Sat	Fri	0,3	0,75	0,39	NO
Sun	Mon	0,71	0,87	0,82	YES
Sun	Tues	0,85	0,96	0,88	YES
Sun	Wed	0,9	1,1	0,82	YES
Sun	Thur	0,91	0,93	0,97	YES
Sun	Fri y	0,85	0,9	0,94	YES
Mon	Tues	0,37	0,76	0,48	NO
Mon	Wed	0,6	0,89	0,67	NO
Mon	Thur	0,39	0,73	0,53	NO
Mon	Fri	0,56	0,7	0,79	NO
Tues	Wed	0,38	0,98	0,39	NO
Tues	Thur	0,21	0,82	0,26	NO
Tues	Fri	0,46	0,79	0,59	NO
Wed	Thur	0,52	0,95	0,55	NO
Wed	Fri	0,69	0,92	0,75	NO
Thur	Fri	0,36	0,76	0,48	NO

X and Y are quite different. A threshold is arbitrary fixed to 0.8.

An example is given in the table 1². In this example the difference between two days in the house 2000997 for the *Electric-oven* over all the year is calculated. The difference between two days is significant if the ratio between $Diff(X, Y)$ and $\frac{\sum_i(x_i+y_i)}{2}$ is bigger than the given threshold.

4.1.2 Clustering Algorithm

The Dissimilarity index given in the equation 2 helps the *Learning System* to decide if two days or two months can be merged for a given service. Then the standard profile associated with this service is reduced in size and by the same time it will require less observations to be adjusted to the actual user. This type of treatment is called *Clustering*. *Clustering* can be considered as the most important unsupervised learning problem. It is a *process of partitioning a set of data (or objects) in a set of meaningful sub-classes, called cluster* (Zaane, 1999). A cluster is therefore a collection of *similar* objects that are *dissimilar* to the objects belonging to other clusters. There are some clustering algorithms like *K-means*, *k-Medoid*, *hierarchical algorithm*, etc. In this paper, a clustering algorithm is proposed as follows based on the Dissimilarity index. The objects can be the

²Sat: Saturday, Sun: Sunday, Mon: Monday, Tues: Tuesday, Fri: Friday, Wed: Wednesday, Thur: Thursday.

months or the days.

Segment(*input E: Set of elements, DI: Dissimilarity indexes; Output C: set of clusters C_e*)

1. Find the closest two elements (*e_x, e_y*) ;

$$DI(e_x, e_y) = \text{Min}\{DI(e_i, e_j), e_i \in E, e_j \in E, i \neq j\}$$

Add to C_e if:

$$DI(e_x, e_y) < 0.8, \text{ then } C_e = \{e_x, e_y\}$$

2. If C_e is empty then go to the step 7 else go to the step 3
3. Calculate the set E₁ = E - C_e
4. For each element e_z of E₁ if $DI(e_z, e_x) < 0.8$ and $DI(e_z, e_y) < 0.8$ then add e_z to C_e
5. Add the cluster C_e to C
6. **Segment**(E - C_e, DI, C)
7. For each element e_r of E: Add e_r to C
8. Return C

This algorithm takes a set of elements which may be the starting probabilities at each hour over days or the months. It takes also the Dissimilarity index between each two elements (table 1). The first step of this algorithm consists in finding the closest two elements (e_x, e_y) which have the smallest Dissimilarity index. Then, for every remaining element from E, the algorithm finds all the other elements which are closer to e_x and e_y than the given threshold. The obtained set C_e represents the first cluster. This algorithm is recursive. It is iteratively called to find all the clusters. It ends when the remaining Discrimination indexes are all greater than the given threshold.

For example, the clusters obtained by applying this algorithm on the table 1 are: C₁ = {Sat, Mon, Tues, Wed, Thurs, Fri} and C₂ = {Sun}. That means that the use of the *Electric-oven* is different from the other days on **Sunday**.

5 CONCLUSIONS AND PERSPECTIVES

This paper focuses on the prediction of the user's behavior in housing and his derived energy consumption. It is a very important predictive problem in a *Home Automation System*. The objective is to construct a *Learning System* able to predict the user's

behavior in housing with regards to his energy consumption. The proposed system builds a set of profiles from the *IRISE* databases for each appliance. A profile is defined by the probability that the associated service starts, the average consumed energy and the average duration. Also, each profile is characterized by the set of days and the set of months during which the consumption is specific. A questionnaire is proposed to the user concerning the use of its appliances. The comparison between the response of the user and the set of standard profiles allows to provide starting standard profiles to the *Home Automation System*. These values are introduced into a Bayesian Network to be adjusted with the actual consumption of the user. Future works will be dedicated to perform the segmentation to the *IRISE* data in order to identify the standard profiles. Then a questionnaire will be defined and the way how to process the comparison between the response and the standard profiles will be addressed.

REFERENCES

- Abras, S., Ploix, S., Pesty, S., and Jacomino, M. (2007). A multi-agent design for a home automation system dedicated to power management. In *Proceedings of the IFIP Conference on Artificial Intelligence Applications and Innovations*, Athen, Greece. Springer.
- Abras, S., Ploix, S., Pesty, S., and Jacomino, M. (2008). An anticipation mechanism for power management in a smart home using multi-agent systems. In *ICTTA'08: Proceedings of the 3rd International Conference on Information and Communication Technologies: from Theory to Applications*, pages 110–116, Damascus, Syria. IEEE Computer Society.
- Barco, R., Nielsen, L., Guerrero, R., Hylander, G., and Patel, S. (2002). Automated troubleshooting of a mobile communication network using bayesian networks. *Mobile and Wireless Communications Network, 2002. 4th International Workshop on*, pages 606 – 610.
- Becker, A., Geiger, D., Schffer, A. A., and Schaffer, A. A. (1998). Automatic selection of loop breakers for genetic linkage analysis. *Human Heredity*, 48:49–60.
- Ezawa, K. and Schuermann, T. (1995). Fraud/uncollectible debt detection using a bayesian network based learning system: A. In *Proceedings of the 11th Annual Conference on Uncertainty in Artificial Intelligence (UAI-95)*, pages 157–16, San Francisco, CA. Morgan Kaufmann.
- HA, L., Ploix, S., Zamai, E., and Jacomino, M. (2006). A home automation system to improve the household energy control. In *INCOM2006 12th IFAC Symposium of Information Control Problems in Manufacturing*, Saint Etienne, France.
- Ha, S., Jung, H., and Oh, Y. (2006). Method to analyze user behavior in home environment. *Personal Ubiquitous Comput.*, 10(2-3):110–121.
- Ha, D. L.; Ploix, S. Z. E. . J. M. (2005). Control of energy consumption in home automation by ressource constraint scheduling. In *The 15th International Conference on Control System and ComputerScience*.
- Hart, P. E. and Graham, J. (1997). Query-free information retrieval. *IEEE Intelligent Systems*, 12(5):32–37.
- Heckerman, D. (1995). A tutorial on learning bayesian networks. Technical report, Communications of the ACM.
- Horvitz, E. and Barry, M. (1995). Display of information for time-critical decision making. In *In Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 296–305. Morgan Kaufmann.
- Naim, P., Wullemin, P.-H., Leray, P., Pourret, O., and Becker, A. (2004). *Réseaux bayésiens*. Eyrolles, Paris.
- Palensky, P., Dietrich, D., Posta, R., and Reiter, H. (1997). Demand side management in private homes by using lonworks. *Vortrag: WFCS97 2nd IEEE Workshop on Factory Communication Systems, Barcelona*, pages 341 – 347.
- Pearl, J. (1986). Fusion, propagation, and structuring in belief networks. *Artif. Intell.*, 29(3):241–288.
- Russell, S. J. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Pearson Education.
- Wood, G. and Newborough, M. (2003). Dynamic energy-consumption indicators for domestic appliances: environment, behaviour and design. *Energy and Buildings*, pages 821–841.
- Wood, G. and Newborough, M. (2007). Influencing user behaviour with energy information display systems for intelligent homes. *International journal of energy research*, vol. 31, no1:56–78.
- Zaane, O. R. (1999). *Principles of Knowledge Discovery in Databases - Chapter 8: Data Clustering*.
- Zimmerman, G. (2007). Modeling and simulation of individual user behavior for building performance predictions. In *SCSC: Proceedings of the 2007 summer computer simulation conference*, pages 913–920, San Diego, CA, USA. Society for Computer Simulation International.

A SPATIAL ONTOLOGY FOR HUMAN-ROBOT INTERACTION

Lamia Belouaer, Maroua Bouzid and Abdel-illah Mouaddib
Université de Caen - UFR des Sciences, Département d'Informatique
Campus Côte de Nacre, Bd Maréchal Juin, BP 5186, 14032 Caen Cedex, France
f_name.s_name@info.unicaen.fr

Keywords: Spatial Representation and Reasoning, Spatial Relations, Ontology, Planning, Human-robot Interaction.

Abstract: Robotics quickly evolved in the recent years. This development widened the intervention fields of robots. Robots interact with humans in order to serve them. Improving the quality of this interaction requires to endow robots with spatial representation and/or reasoning system. Many works have been dedicated to this purpose. Most of them take into account metric, symbolic spatial relationships. However, they do not consider fuzzy relations given by linguistic variables in humans language in human-robot interaction. These relations are not understood by robots. Our objective is to combine human representation (symbolic, fuzzy) of space with the robot's one to develop a mixed reasoning. More precisely, we propose an ontology to manage both spatial relations (topological, metric), fuzziness in spatial representation. This ontology allows a hierarchical organization of space which is naturally manageable by humans and easily understandable by robots. Our ontology will be incorporated into a planner by extending the planning language PDDL.

1 INTRODUCTION

Robotics quickly evolved in the last decade and there is an increasing demand for intelligent systems like robots that can help in daily life. This development widened the intervention fields of robots such as a public area where robots interact with humans in order to serve them. To improve the quality of this interaction, robots should behave as much as possible like humans. This requires to endow robots with representations and/or systems of reasoning directly inspired by humans. We focus on human-robot interaction (HRI) based on spatial organization of observed structures, in order to plan robots actions.

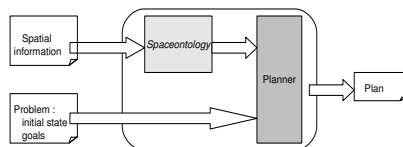


Figure 1: Spatial Planner.

Our goal is to develop a planner allowing to solve problems taking into account spatial information. This planner, called *Spatial Planner* (Fig. 1), consists of two sub-systems. The first sub-system, *SpaceOntology* allows a spatial representation and reasoning model. As input, it takes a set of imprecise and incomplete spatial information. However,

as output, it provides a structured knowledge about the environment to explored in planning. The second sub-system, *Planner*, defines the set of actions to be executed by the robot in order to achieve its mission.

In this paper, we focus on *SpaceOntology* that models spatial representation and reasoning for better mediation between humans and robots. This ontology concerns:

- Hierarchical representation of space. The space is structured to be manageable by humans and robots.
- Numerical/Symbolic representation of space. From the human's point of view, the space is generally considered in a symbolic way (in, disjoint, north, close, ...). From the robot's point of view, the space is considered in numerical way (angles, distances, ...).

SpaceOntology gives a description of the environment (hierarchical organization, spatial relations) understandable by both humans and robots.

This paper is organized as follows. In section 2, we place our work with regard to the state of art of the spatial representation and ontologies. In section 3, we present our framework to model spatial representation and reasoning. In section 4, we present *SpaceOntology*. In section 5, we present how we use ontology to develop a mixed reasoning (from human to robot and

from robot to human). In conclusion, we present our future works.

2 RELATED WORK

2.1 Spatial Relations

The spatial relations have been developed in many domains (image processing (Bloch and Ralescu, 2003), GIS (Casati et al., 1998), ...). They can be divided into topological, directional and distance relations (Kuipers and Levitt, 1988). In this paper, we consider all these relations.

In robotics, quantitative representations of spatial relations are commonly applied. Purely quantitative representations have limitations particularly when imprecise knowledge use spatial relations expressed in linguistic terms, particular in HRI. Imprecision has to be taken into account in such problems. It is often inherent to human language. It may be caused by imprecision about the objects to recognize due to the absence of crisp contours or by the imprecise semantics of some relationships (eg. quite far, ...), or else by a kind of task we would like fulfill in HRI. For example, we may want a robot go towards a person while remaining at security distance of it.

Our objective is to combine all symbolic representation with robotic numeric representation to develop a mixed reasoning.

2.2 Ontologies and Spatial Dimension

Different techniques of spatial representation and reasoning have been proposed. Most are based on constraints, logical and algebraic approaches (Balbiani et al., 1999). However, these approaches can not manage quantitative, qualitative and imprecise knowledge at the same time. In HRI, we need to combine this knowledge. For this reason, we must use an unified framework to cover large classes of problems and potential applications, and able to give rise to instantiation adapted to each particular application. Ontologies (Gruber et al., 1995) appear as an appropriate tool toward this aim.

Spatial ontologies can be found in some fields such as GIS (Casati et al., 1998), Virtual Reality (Dasiopoulou et al., 2005), Robotics (Dominey et al., 2004), ... All these ontologies are focused on the representation of spatial concepts according to the application domains. A major weakness of usual ontological technologies is their inability to represent and to reason with imprecision. An interesting work presented in (Hudelot et al., 2008) overcome this limit.

3 OUR FRAMEWORK

In our work, we set up an ontology to manage both spatial relations, fuzziness in spatial representation. Moreover, our ontology allows an organization of space naturally manageable by human and easily understandable by the robot: *hierarchical representation of space*.

3.1 Hierarchical Organization of Space

Hierarchical organization of space reduces the amount of information considered for coping with a complex, high-detailed world: concepts are grouped into more general ones and these are considered new concepts that can be abstracted again. The result is a hierarchy of abstractions (or specialization) or a hierarchy of concepts that ends when all information is modeled by a single universal concept (or we reach a desired level of specialization). Thus, we consider this hierarchy to describe the considered space.

Our organization is made from the highest abstraction level to lowest (most detailed one) unlike the organization described in (Fernández-Madriral et al., 2004). The highest level represents the environment with the maximum amount of detail available. The lowest level represents the environment by a single concept. Hierarchical representation of space allows us to represent this space in a structure easily manageable by human and robot. In addition, it provides better performance than flat representation in navigation or path planning.

3.1.1 Spatial Entity

All concepts in spatial representation are called *Spatial Entities*. A spatial entity is localizable in a given space by one of its attributes or by geometric transformation. From a geometric point of view, a spatial entity ϵ is defined by a rectangle $rect_\epsilon$ corresponding to its axis-aligned bounding rectangles.

From hierarchical organization of space, derives two categories of spatial entities. *Space* represents a global space. This entity is the highest abstraction and the lowest level in the hierarchy organization. *Region* represents any spatial entity belonging to different hierarchical levels (intermediate and final). A region is a sub-space included in the given space. A region is itself considered as a space that can be decomposed into different sub-regions.

3.2 Spatial Relations

A spatial relation requires a reference frame. For example, the relation *bench is in front of coffee machine*.

The semantic of the relation is not the same depending on whether the reference system is the coffee machine itself or an external observer. In order to have a unique meaning and to remove the ambiguity, three concepts have to be specified : the target object, the reference object and the reference system (Hudelot, 2005). In our work, we consider both *Intrinsic* and *Egocentric* reference frame.

3.2.1 Topological Relationships

We consider the ALBR relations defined in **ABLR (Above Below West Right)** (Laborie et al., 2006). This algebra balances between expressiveness and the number of relations (reasoning/complexity). **ABLR** reduces the number of relations while preserving the directionality property of the representation defined in (Allen, 1983). A topological relation is an **ABLR** relation. This relation is a couple $\langle r_X, r_Y \rangle$, where:
 $r_X \in \{Left(L), OverlapsLeft(O_L), Contains(C_x), Inside(I_x), OverlapsRight(O_R), Right(R)\}$
 and
 $r_Y \in \{Above(A), OverlapsAbove(O_A), Contains(C_y), Inside(I_y), OverlapsBelow(O_B), Below(B)\}$.

3.2.2 Metrical Relationships

Metrical relations concern distance and orientation relations (Bloch, 2005). We consider a 2D representation of the space given by (O, \vec{i}, \vec{j}) . The origin O is a reference system that can be intrinsic or egocentric. In the following, a rectangle denoted ϵ represents a spatial entity. Its symmetry center will be known as the spatial entity name. $P_x(\epsilon)$ (resp. $P_y(\epsilon)$) denotes the projection of ϵ on (\vec{i}) axis (resp. (\vec{j}) axis).

In HRI under spatial constraints, fuzzy information is a key point as said in section 2. In this work, vagueness and ambiguity concern the vagueness of the relationship itself. Indeed, we don't need to evaluate if a spatial entity is in north of a referent spatial entity since spatial entities are crisp. The application of fuzzy approach mainly concerns the distance relationship. The aim is to find a way to represent the symbolic direction and distance relationship (based on linguistic variables) by a numerical direction and distance and vice versa.

Directional Relationships. We describe directional relations through cardinal direction : *West of*, *North of*, *East of* and *South of*. We associate for every semantic direction semantic *West of*, *East of*, *North of* and *South of* following respective functions $West_R$, $East_R$, $North_R$ (R is a referent object) and $South_R$. $West_R(\epsilon)$ denotes ϵ is left of R given by $West_R(\epsilon) = \{P_x(\epsilon) - P_x(R) \leq 0\}$. $East_R(\epsilon)$ denotes ϵ is right of R

given by $East_R(\epsilon) = \{P_x(\epsilon) - P_x(R) \geq 0\}$. $North_R(\epsilon)$ denotes ϵ is in north of R given by $North_R(\epsilon) = \{P_y(\epsilon) - P_y(R) \geq 0\}$. $South_R(\epsilon)$ denotes ϵ is in south of R given by $South_R(\epsilon) = \{P_y(\epsilon) - P_y(R) \leq 0\}$.

The representation of the 8 cardinal relationships is possible by combining these four functions. Consider the example of the directional relation the bench b is north and east of coffee machine c_m . This corresponds to the combination of $North_{c_m}(b)$ and $East_{c_m}(b)$:

$$North_{c_m}(b) \oplus East_{c_m}(b) = \begin{cases} P_x(b) - P_x(c_m) \geq 0 \\ P_y(b) - P_y(c_m) \geq 0 \end{cases} \quad (1)$$

This representation allows us to express other directional relations (at the same level, between, ...).

Distance Relationships. We consider four linguistic variables to describe distance relations: *close to*, *close to enough*, *far from enough*, *far from*. We note $d(\epsilon, r)$ in \mathbb{R}^+ the euclidean distance between the point of symmetry of two rectangles representing two regions (r referent object and ϵ target object). The aim is to find a way to represent the four linguistic variables already defined to evaluate distance by a numeric value to evaluate it. To do so, we consider two degrees, defined in (Schockaert, 2008), $N_{(\alpha, \beta)}(p, q)$ (2) and $F_{(\alpha, \beta)}(p, q)$ (3). The degree N represents two points p and q are near each other and the degree F represents how p is far from q ($\alpha, \beta > 0$). We have defined a hierarchical space organization. This has an impact on the distance evaluation. Indeed, the distance of $2m$ in a city is considered as *near*, however, $2m$ in an office is considered as *far*. From these information, we define for each hierarchical level an α and a β depending on the scale of this level.

$$N_{(\alpha, \beta)}(p, q) = \begin{cases} 1 & \text{if } d(p, q) \leq \alpha \\ 0 & \text{if } d(p, q) \geq \alpha + \beta \\ \frac{\alpha + \beta - d(p, q)}{\beta} & \text{otherwise } \beta \neq 0 \end{cases} \quad (2)$$

$$F_{(\alpha, \beta)}(p, q) = \begin{cases} 1 & \text{if } d(p, q) > \alpha + \beta \\ 0 & \text{if } d(p, q) \leq \alpha \\ \frac{d(p, q) - \alpha}{\beta} & \text{otherwise } \beta \neq 0 \end{cases} \quad (3)$$

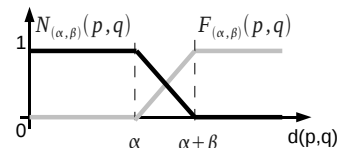


Figure 2: Graphical representation of relationship between $N_{(\alpha, \beta)}(p, q)$ and $F_{(\alpha, \beta)}(p, q)$.

From equations 2 and 3 and organization given in figure 2, it is easy to deduce that: (1) if $d \in [0, \alpha]$ then

d is considered as *close*, (2) if $d \in]\alpha, \alpha + \frac{\beta}{2}]$ then d is considered as *close enough*, (3) if $d \in]\alpha + \frac{\beta}{2}, \alpha + \beta]$ then d is considered as *far enough*, (4) if $d \in]\alpha + \beta, +\infty[$ then d is considered as *far*.

4 IMPLEMENTATION

As a formal language, we opted for OWL DL formalism (McGuinness et al., 2004; Baader et al., 2003). This formalism benefits from the compactness and expressiveness of DL. Indeed, an important characteristic of DL is their reasoning capabilities of inferring implicit knowledge from the explicitly represented knowledge. In this section, we describe how we present and reason about spatial knowledge.

4.1 Spatial Entities as Concepts

One of important concepts of *SpaceOntology* is the concept **Space** ($\text{Space} \sqsubseteq \text{T}$) (T for Thing¹). This concept represents a global environment (i.e a country, a city, ...). Also, we define a concept **Regions**. This concept is a subclass of concept **Space** ($\text{Regions} \sqsubseteq \text{Space}$). Thus, we can consider the hierarchical definition of space. Indeed, a region itself is a space in the next hierarchical level. Furthermore, the hierarchical relationship between concepts **Space** and **Regions** is given by subsumption. We offer the following links *consistsOf* and *isPartOf*. The link *consistsOf* can express that space consists of one or more regions. The link *isPartOf* can express one region may belong to one or more spaces. These relationships are symmetrical and transitive.

$$\text{Space} \sqsubseteq \text{T} \sqcap \exists \text{ consistsOf.Regions} \sqcap \geq 1 \text{ consistsOf}$$

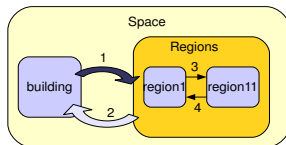
$$\text{Regions} \sqsubseteq \text{Space} \sqcap \exists \text{ isPartOf.Space} \sqcap \geq 1 \text{ consistsOf}$$


Figure 3: Links between concepts and their instantiation.

From these links and their properties, we can compose relationships between these two concepts. For instance, the composite (2) relationship in figure 3, is

¹Thing is an OWL class that represents the set containing all individuals. Because of this all classes are subclasses of OWL:Thing.

derived from the transitive links *isPartOf* (4) between *region1* with *region1* and *region1* with *building*. Through this relationship we can deduce that *region1* is part of the *building*.

4.2 Spatial Relations as Concepts

A spatial relation is not considered in our ontology as a property between two regions but as a concept on its own; **SpatialRelations** ($\text{SpatialRelations} \sqsubseteq \text{T}$). This concept represents a set of all spatial relations between two regions. A **SpatialRelations** subsumes **TopologicalRelations** and **MetricRelations** which itself subsumes **DirectionalRelations** and **DistanceRelations** which itself subsumes **DistanceAccordingToActions** and **DistanceAccordingToHierarchicalLevel**.

4.2.1 HasRelation Concept

To define a spatial relationship between two regions describing a given configuration, we need to link these regions with a spatial relationship.

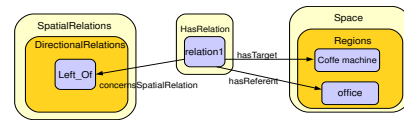


Figure 4: Links between concepts and their instantiation.

As already defined, a spatial relationship is given by the concept **SpatialRelations**. We define the link **HasRelation** as a concept which refers to the set of spatial relations for which target and reference entities are defined. This concept is useful to describe spatial configurations.

$$\begin{aligned} \text{HasRelation} \sqsubseteq \text{T} \sqcap \exists \\ \text{concernsSpatialRelation.SpatialRelations} \sqcap = 1 \\ \text{concernsSpatialRelation} \sqcap \exists \\ \text{hasReferent.Regions} \sqcap \geq 1 \text{ hasReferent} \sqcap \exists \\ \text{hasTarget.Regions} \sqcap = 1 \text{ hasTarget} \end{aligned}$$

Consider as an example that human asks the robot to enter into "the office left of the coffee machine". In *SpaceOntology*, this expression is formalized as follows. First, identify this expression by *relation1*. We note $\text{relation1:HasRelation}^2$. We consider that *relation1* is an instantiation of the concept **HasRelation**. *Left_of* is an instantiation of the concept **DirectionalRelations** according to a defined reference system for spatial relations ($\text{Left_of:DirectionalRelations}$). The office and

²These symbols are defined in description logics syntax and interpretation

the coffee machine are instantiations of the concept **Regions** (`office:Regions` and `coffee:Regions`).

5 OPERATING SPACE ONTOLOGY

In this section, we present the exploitation of our ontology and the methods for reasoning. As already mentioned, we developed an ontology to provide a basis spatial data to be used after in planning problems to improve HRI. Exploitation of ontology is necessary for path planning between two positions. In this paper, we present how and by which methods from ontology we can extract the paths between two positions even if the information is incomplete. Work on path planning is the subject of future work.

5.1 Example

Consider an HRI problem in a building. A human asks the robot to fetch a bottle near the coffee machine located in the hallway of the third floor. Then, to bring back the bottle to the human who is on the first floor (initial robot position's). Specifically, robot should compute the path between its position and bottle position's, catching the bottle and after to give it to human. To explain the reasoning, we consider in the following only the first part of the task, namely; fetching the bottle.

5.2 Reasoning

In *SpaceOntology*, we insisted on two key notions: hierarchy of space and spatial relationships. Thus, we rely on these notions for reasoning.

Consider the example given in section 5.1. By giving a map with all the corridors and all access, defining a path by considering the size of the map becomes quickly expensive. Hierarchical organization of space simplifies the path computation. Indeed, it helps to decompose the problem into 3 sub-problems: (1) from initial position reach an access point to the third floor, (2) from this access point plan to reach the third floor and (3) from arrival position on the third floor, plan to reach the coffee machine.

Reasoning for the first and last steps requires more detailed information than the second step. For this step, we must ignore the details given in two other steps. However, considering floors like black boxes does not guarantee the path quality. Consider that quality is related to speed. Passing through certain corridors with big distances can be faster than through the ones with short distances but with many obstacles.

Thus, hierarchical organization of a space, involves reasoning at each level.

Thus, we can construct a path between any two given locations in an accurate (i.e in room number 3) or approximate (i.e somewhere on the first floor). In this paper, we do not present an algorithm for finding paths, but rather we present a structure generated from the ontology providing a set of possible paths between two positions by considering the space hierarchy. For the target object, in our case the bottle, we define the concept of *target zone*. Considering the hierarchy of the environment, we define the target zone as $T_Z^l(o)$, where l is the hierarchy level and o the object or region targeted. Thus, we can deduce from *SpaceOntology* a hierarchy for the target zone. This allows us to determine the most abstract target zone ($T_Z^0(o) = \textit{building}$) and the more detailed one ($T_Z^3(o) = \textit{region of coffee machine}$).

Another key knowledge in this work are spatial relationships. They allow, given an environment, to describe its spatial configuration (obstacle position's). For instance, we can describe that the corridor H is adjacent to the door *doorB* of the office B . Here, an example from *SpaceOntology.owl* allowing to illustrate the example.

```
<HasRelationWithIntersection rdf:ID="relation3">
  <intersectionresult>
    <Regions rdf:ID="doorB"/>
  </intersectionresult>
  <hasTarget>
    <Regions rdf:ID="corridorH"/>
  </hasTarget>
  <hasReferent>
    <Regions rdf:ID="officeB"/>
  </hasReferent>
  <concernsSpatialRelation>
    <TopologicalRelations rdf:ID="leftinside">
      <isahRelation rdf:resource="#Horiz_L"/>
      <isavRelation rdf:resource="#Ver_Iy"/>
    </TopologicalRelations>
  </concernsSpatialRelation>
</HasRelationWithIntersection>
```

As already mentioned, our strategy of searching a path is to find a path divided into different portions. Each part belongs to a single hierarchical level. To do this, we need a structure for this type of dedicated research. This structure is generated from *SpaceOntology*. Thus, we define the *Crossing Network Graph*.

5.2.1 Crossing Network Graph

A *Crossing Network Graph* (Γ_G) is a directed graph. A node in this graph represents a network of passage. The nodes are organised hierarchically. The arcs represent relationships between nodes as described in *SpaceOntology*.

A *Crossing Network*(Γ) is a graph whose nodes are *Crossing Network*. Edges represent spatial adjacency relations between two nodes described in *SpaceOntology* giving a contact point between these nodes, known as gateways. A gateway (i.e door, passing lane, ...) allows transitions between adjacent spaces and between spaces adjacent in different hierarchical levels. Edges are labeled by a couple (*pass*, *dist*), where *pass* gives the gateway connecting these regions (or networks) and *dist* is the distance separating these regions (or networks) passing through this gateway. There are two types of crossing networks: (1) Low-level crossing Networks are crossing networks whose nodes are crossing networks. It used such a network mainly as we have not reached the level of specialization wanted (or fixed). (2) High-level crossing Networks are crossing networks whose nodes are the regions. It used when level of specialization desired (or fixed) is reached.

The construction of *Crossing Network Graph* is done from the abstract level to fixed detailed level. First, we consider the target zone of the most abstract level. In the same way, we consider the initial zone of the most abstract level in the ontology. We select the most abstract target zone and initial zone targeted areas as these two zones are included in the same region. For instance, we consider the third floor (target zone) and the first floor (initial). It requires setting different gateways allowed to exit the initial zone and enter to the target zone. From the spatial relationships of adjacency defined *SpaceOntology*, we can find with backward mechanism all possible paths to reach the initial region.

6 CONCLUSIONS

This paper presents a spatial representation using an ontology allowing us to represent and reason on spatial objects represented from different point of views (human and robot). Future work will concern to integrate it in planning by extending the planning language PDDL. This is an innovative concept. In this paper, several aspects are still cause for thought as the assessment of a relationship without a fixed target or the implementation of an algorithm to generate a path according to some optimality criteria. These points will be the subject of future work.

REFERENCES

Allen, J. (1983). Maintaining knowledge about temporal intervals.

- Baader, F., Calvanese, D., McGuinness, D., Patel-Schneider, P., and Nardi, D. (2003). *The description logic handbook: theory, implementation, and applications*. Cambridge Univ Pr.
- Balbani, P., Condotta, J.-F., and del Cerro, L. F. (1999). A new tractable subclass of the rectangle algebra. In *IJCAI*, pages 442–447.
- Bloch, I. (2005). Fuzzy spatial relationships for image processing and interpretation: a review. *Image and Vision Computing*, 23:89–110.
- Bloch, I. and Ralescu, A. (2003). Directional relative position between objects in image processing: a comparison between fuzzy approaches. *Pattern Recognition*.
- Casati, R., Smith, B., and Varzi, A. (1998). Ontological tools for geographic representation. (FOIS'98).
- Dasiopoulou, S., Mezaris, V., Kompatsiaris, I., Papastathis, V., and Strintzis, M. (2005). Knowledge-assisted semantic video object detection. *IEEE Transactions on Circuits and Systems for Video Technology*.
- Dominey, P., Boucher, J., and Inui, T. (2004). Building an adaptive spoken language interface for perceptually grounded human–robot interaction. In *Proceedings of the IEEE-RAS/RSJ international conference on humanoid robots*.
- Fernández-Madrugal, J., Galindo, C., and González, J. (2004). Assistive navigation of a robotic wheelchair using a multihierarchical model of the environment. *Integrated Computer-Aided Engineering*.
- Gruber, T. et al. (1995). Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human Computer Studies*.
- Hudelot, C. (2005). Towards a cognitive vision platform for semantic image interpretation; application to the recognition of biological organisms.
- Hudelot, C., Atif, J., and Bloch, I. (2008). Fuzzy spatial relation ontology for image interpretation. *Fuzzy Sets and Systems*.
- Kuipers, B. (2000). The spatial semantic hierarchy. *Artificial Intelligence*, 119:191–233.
- Kuipers, B. and Levitt, T. (1988). Navigation and mapping in large scale space. *AI magazine*, 9:25.
- Laborie, S., Euzenat, J., and Layaida, N. (2006). A spatial algebra for multimedia document adaptation. *SMAT*.
- McGuinness, D., Van Harmelen, F., et al. (2004). OWL web ontology language overview. *W3C recommendation*.
- Schockaert, S. (2008). *Reasoning about fuzzy temporal and spatial information from the web*. PhD thesis, Ghent University, 2008.

REINFORCEMENT LEARNING FOR ROBOT CONTROL USING PROBABILITY DENSITY ESTIMATIONS

Alejandro Agostini and Enric Celaya

Institut de robòtica i Informàtica Industrial, (CSIC-UPC), Barcelona, Spain
{agostini, celaya}@iri.upc.edu

Keywords: Machine Learning in control applications, Reinforcement learning.

Abstract: The successful application of Reinforcement Learning (RL) techniques to robot control is limited by the fact that, in most robotic tasks, the state and action spaces are continuous, multidimensional, and in essence, too large for conventional RL algorithms to work. The well known curse of dimensionality makes infeasible using a tabular representation of the value function, which is the classical approach that provides convergence guarantees. When a function approximation technique is used to generalize among similar states, the convergence of the algorithm is compromised, since updates unavoidably affect an extended region of the domain, that is, some situations are modified in a way that has not been really experienced, and the update may degrade the approximation. We propose a RL algorithm that uses a probability density estimation in the joint space of states, actions and Q -values as a means of function approximation. This allows us to devise an updating approach that, taking into account the local sampling density, avoids an excessive modification of the approximation far from the observed sample.

1 INTRODUCTION

Any robotic application requires a precise control of the robot effectors. In many situations, the design of the control system and the necessary tuning of its parameters becomes a hard task, and there is a great interest in providing the robot with the ability to learn by itself how to control its effectors through experience. The most usual approach to achieve this is Reinforcement Learning (RL) (Sutton and Barto, 1998), in which the robot must find an optimal action policy by trial and error using a signal, called reinforcement, that tells how good was the result obtained after each executed action.

Reinforcement Learning algorithms, like value iteration or policy iteration, are known to converge in the case of finite state-action spaces, where a tabular representation of the value function or the utility function Q can be used. However, in a typical robot control problem, the state and action spaces are continuous or too large, so that it is necessary to use some form of function approximation. In this case convergence is no longer granted in general, and this is attributed to the fact that the update on the function approximation that takes place after each experience is not local to the precise observed point in

the state-action space, but influences the values in other regions, possibly undoing the learning done so far (Riedmiller, 2005a). This problem is magnified when the sampling of the domain is too biased, causing some regions to be updated much more often than others.

Recently, much work has been done in RL with function approximation. For example, (Ernst et al., 2005) proposed the *fitted Q Iteration algorithm*, based on previous work of (Ormoneit and Sen, 2002) on kernel-based RL, and using randomized trees for function approximation. Similarly, (Riedmiller, 2005a) proposed the Neural Fitted Q Iteration (NFQ) algorithm, which uses a multi-layer perceptron for function representation. In a different approach, (Rasmussen and Kuss, 2004) used Gaussian Processes (GP) to model the system dynamics, and a further GP to represent the Value function. The same approach is followed by (Rottmann and Burgard, 2009), while (Engel et al., 2005) used a GP to directly represent the Q -function in a model-free setting. All these algorithms fall into the class of the so-called *fitted value iteration* algorithms (Gordon, 1995), which, in order to approximate the desired function, take a finite number of training samples and try to fit the function to them in a batch, iterative process. The usual approach

to obtain the training samples, when the knowledge of the system dynamics is available, consists in sampling uniformly all the state-action space to build a training set that covers all possible situations sufficiently well. Clearly, this procedure is not possible when dealing with a real system with unknown dynamics, in what case samples can only be observed while interacting with the real system. In the simplest cases, it is possible to roughly cover the whole state space by chaining a number of random actions, as in (Ernst et al., 2005). However, when the problem grows in complexity, the probability of executing a random sequence that drives the system to the interesting regions of the workspace may be too low to be achieved in practical time. In such cases it is necessary to exploit the knowledge already obtained with previous interactions (Riedmiller, 2005a; Ernst et al., 2005).

It has to be noted that the need of exploiting what has been learned so far introduces a tendency to experience the most promising states much more often than others, and this systematically produces a very biased sampling that aggravates the perturbing effect caused by non-local updating pointed out before. In (Riedmiller, 2005a), this problem is avoided by assuring that all datapoints are used for update the same number of times. This is made possible by remembering a dense enough set of transitions and performing full updates in batch mode. In fact this is a common trait of all fitted value iteration algorithms. From a computational point of view, this approach is very computationally intensive, since all datapoints are used a large number of times until convergence is reached. A more efficient approach would result if, instead of retraining with old data in batch, an incremental updating could be achieved in which the perturbing effect of new samples on old estimations was attenuated.

In the present work, we address the problem of the biased sampling with incremental updating. In our approach, we take into account how often each region of the domain has been visited, updating more locally those regions that are more densely sampled. To do this, we need an estimation of the sampling density, for what we use a Gaussian Mixture Model (GMM) representing a probability density of samples in the joint space of states, actions, and Q -values. At the same time, this density estimation can be used as a means of function approximation for the Q -function. Density estimations are receiving increasing interest in the field of machine learning (Bishop, 2006), since they keep all the information contained in the data, that is, they provide estimations not only for the expected function value, but also for its uncertainty.

Despite density estimations are more demanding than simple function approximation (due to the fact that they embody more information), their use for function approximation has been advocated by different authors (Figueiredo, 2000; Ghahramani and Jordan, 1994), noting that simple and well understood tools like the Expectation-Maximization (EM) algorithm (Dempster et al., 1977) can be used to obtain accurate estimations of the density function.

The rest of the paper is organized as follows: Section 2 briefly resumes the basics of RL. Section 3 introduces the concepts of GMM for multivariate density estimation, and the EM algorithm in its batch version. In Section 4 we define the on-line EM algorithm for the GMM. In Section 5, we present our approach to deal with biased sampling. In Section 6 we develop our RL algorithm using density estimation of the Q -value function, involving action evaluation and action selection. Section 7 describes the test control application to show the feasibility of the approach. We conclude in Section 8 with a discussion of the proposed approach.

2 THE REINFORCEMENT LEARNING PARADIGM

In the RL paradigm, an agent must improve its performance by selecting actions that maximize the accumulation of rewards provided by the environment (Sutton and Barto, 1998). At each time step, the agent observes the current state s_t and chooses an action a_t according to its policy $a = \pi(s)$. The environment changes to state s_{t+1} in response to this action, and produces an instantaneous reward $r(s_t, a_t)$. The agent must experiment by interacting with the environment in order to find the optimal action policy from the outcome of its past experiences. One of the most popular algorithms used in RL is Q -Learning (Watkins and Dayan, 1992), which uses an action-value function $Q(s, a)$ to estimate the maximum expected future cumulative reward that can be obtained by executing action a in situation s and acting optimally thereafter. Q -learning uses a sampled version of the Bellman optimality equations (Bellman and Dreyfus, 1962) to estimate instantaneous q values,

$$q(s_t, a_t) = r(s_t, a_t) + \gamma \max_a Q(s_{t+1}, a) \quad (1)$$

where $\max_a Q(s_{t+1}, a)$ is the estimated maximum cumulative reward corresponding to the next observed situation s_{t+1} , and γ is a discount factor, with values in $[0,1]$ that regulates the importance of future rewards with respect to immediate ones. At a given stage of

the learning, the temporary policy can be derived from the estimated Q -function as,

$$\pi(s) = \underset{a}{\operatorname{argmax}} Q(s, a) \quad (2)$$

In actor/critic architectures, a policy function (called the actor) is learned and explicitly stored, so that actions are directly decided by the actor and do not need to be computed from the maximization in (2). Despite this computational advantage, the learning of an actor may slow down convergence, since then the learning of the Q -function must be done on-policy instead of off-policy, and both functions, actor and critic, must adapt to each other to reach convergence. In our implementation we avoid the use of an actor, and thus we must face the problem of maximizing the $Q(s, a)$ function in (2).

The basic formulation of Q -learning assumes discrete state-action spaces and the Q -function is stored in a tabular representation. For continuous domains a function approximation is required to represent the Q -function and generalize among similar situations. In next sections we present our proposal for function approximation using density estimations.

3 DENSITY ESTIMATION WITH A GAUSSIAN MIXTURE MODEL

A Gaussian Mixture Model (Bishop, 2006) is a weighted sum of multivariate Gaussian probability density functions, and is used to represent general probability density distributions in multidimensional spaces. It is assumed that the samples of the distribution to be represented have been generated through the following process: first, one Gaussian is randomly selected with *a priori* given probabilities, and then, a sample is randomly generated with the probability distribution of the selected Gaussian. According to this, the probability density function of generating sample \mathbf{x} is:

$$p(\mathbf{x}; \Theta) = \sum_{i=1}^K \alpha_i \mathcal{G}(\mathbf{x}; \mu_i, \Sigma_i) \quad (3)$$

where K is the number of Gaussians of the mixture; α_i , usually denoted as the mixing parameter, is the prior probability, $P(i)$, of Gaussian i to generate a sample; $\mathcal{G}(\mathbf{x}; \mu_i, \Sigma_i)$ is a multidimensional normal Gaussian function with mean vector μ_i and covariance matrix Σ_i ; and $\Theta = \{\{\alpha_1, \mu_1, \Sigma_1\}, \dots, \{\alpha_K, \mu_K, \Sigma_K\}\}$ is the whole set of parameters of the mixture. By allowing the adaption of the number K of Gaussians

in the mixture, any smooth density distribution can be approximated arbitrarily close (Figueiredo, 2000). The parameters of the model can be estimated using a maximum-likelihood estimator (MLE). Given a set of samples $\mathbf{X} = \{\mathbf{x}_t; t = 1, \dots, N\}$, the likelihood function is given by

$$\mathcal{L}[\mathbf{X}; \Theta] = \prod_{t=1}^N p(\mathbf{x}_t; \Theta). \quad (4)$$

The maximum-likelihood estimation of the model parameters is the Θ that maximizes the likelihood (4) for the data set \mathbf{X} . Direct computation of the MLE requires complete information about which mixture component generated which instance. Since this information is missing, the EM algorithm, described in the next section, is often used.

3.1 The Expectation-maximization Algorithm

The Expectation-Maximization (EM) algorithm (Dempster et al., 1977) is a general tool that permits to estimate the parameters that maximize the likelihood function (4) for a board class of problems when there are some missing data. The EM method first produces an estimation of the expected values of the missing data using initial values of the parameters to be estimated (E step), and then computes the MLE of the parameters given the expected values of the missing data (M step). This process is repeated iteratively until a convergence criterion is fulfilled.

In this section we briefly describe how EM is applied to the specific case of a GMM. The process starts with an initialization of the mean vectors and covariance matrices of the Gaussians. The E step consists in obtaining the probability $P(i|\mathbf{x}_t)$ for each component i of generating instance \mathbf{x}_t , that we denote by $w_{t,i}$,

$$w_{t,i} = P(i|\mathbf{x}_t) = \frac{P(i)p(\mathbf{x}_t|i)}{\sum_{j=1}^K P(j)p(\mathbf{x}_t|j)} = \frac{\alpha_i \mathcal{G}(\mathbf{x}_t; \mu_i, \Sigma_i)}{\sum_{j=1}^K \alpha_j \mathcal{G}(\mathbf{x}_t; \mu_j, \Sigma_j)} \quad (5)$$

where $t = 1, \dots, N$ and $i = 1, \dots, K$. The maximization step consists in computing the MLE using the estimated $w_{t,i}$. It can be shown (Duda et al., 2001) that, for the case of a GMM, the mixing parameters, means, and covariances are given by

$$\alpha_i = \frac{1}{N} \sum_{t=1}^N w_{t,i} \quad (6)$$

$$\mu_i = \frac{\sum_{t=1}^N w_{t,i} \mathbf{x}_t}{\sum_{t=1}^N w_{t,i}} \quad (7)$$

$$\Sigma_i = \frac{\sum_{t=1}^N w_{t,i} (\mathbf{x}_t - \mu_i) (\mathbf{x}_t - \mu_i)^\top}{\sum_{t=1}^N w_{t,i}} \quad (8)$$

4 ON-LINE EM

Estimating a probability density distribution by means of the EM algorithm involves the iteration of E and M steps on the complete set of available data, that is, the mode of operation of EM is in batch. However, in Q -learning, sample data are not all available at once: they arrive sequentially and must be used online to improve the policy that will allow an efficient exploration-exploitation strategy. This prevents the use of the batch EM algorithm, and requires an on-line, incremental version of it. Several incremental EM algorithms have been proposed for the Gaussian Mixture Model applied to clustering or classification of stationary data (Song and Wang, 2005; Arandjelovic and Cipolla, 2005).

The approach proposed in (Song and Wang, 2005) is not strictly an on-line EM algorithm. It applies the conventional batch EM algorithm onto separate data streams corresponding to successive episodes. For each new stream, a new GMM model is trained in batch mode and then merged with the previous model. The number of components for each new GMM is defined using the Bayesian Information Criterion, and the merging process involves similarity comparisons between Gaussians. This method involves many computationally expensive processes at each episode and tends to generate more components than actually needed. The applicability of this method to RL seems limited, not only for its computational cost, but also because, due to the non-stationarity of the Q -estimation, old data should not be taken as equally valid during all the process.

The work of (Arandjelovic and Cipolla, 2005) performs incremental updating of the density model using no historical data and assuming that consecutive data vary smoothly. The method maintains two GMMs: the current GMM estimation, and a previous GMM of the same complexity after which no model updating (i.e. no change in the number of Gaussians) has been done. By comparing the current GMM with the historical one, it is determined if new Gaussians are generated or if some Gaussians are merged together. Two observed shortcomings of the algorithm are that the system fails when new data is well explained by the historical GMM, and when consecutive data violate the condition of smooth variation.

In (Sato and Ishii, 2000), an on-line EM algorithm

is presented for the Normalized Gaussian Network (NGnet), a model closely related to the GMM. This algorithm is based on the works of (Nowlan, 1991; Neal and Hinton, 1998). In (Nowlan, 1991) a method for the incremental adaptation of the model parameters using a forgetting factor and cumulative statistics is proposed, while in (Neal and Hinton, 1998) the method in (Nowlan, 1991) is evaluated and contrasted with an incremental version which performs steps of EM over a fixed set of samples in an incremental way. The method proposed in (Sato and Ishii, 2000) uses foundations of both works to elaborate an on-line learning algorithm to train a NGnet for regression, where weighted averages of the model parameters are calculated using a learning rate that implicitly incorporates a forgetting factor to deal with non-stationarities. Inspired by this work, we developed an on-line EM algorithm for the GMM. Our approach uses cumulative statistics whose updating involves a forgetting factor explicitly.

4.1 On-line EM for the GMM

In the on-line EM approach, an E step and an M step are performed after the observation of each individual sample. The E step does not differ from the batch version (equation (5)), except that it is only computed for the new sample. For the M step, the parameters of all mixture components are updated with the new sample. For this, we define the following time-discounted weighted sums

$$W_{t,i} = [[1]]_{t,i} \quad (9)$$

$$X_{t,i} = [[\mathbf{x}]]_{t,i} \quad (10)$$

$$(XX)_{t,i} = [[(\mathbf{x} - \mu_i)(\mathbf{x} - \mu_i)^\top]]_{t,i} \quad (11)$$

where we use the notation:

$$[[f]]_{t,i} = \sum_{\tau=1}^t \left(\prod_{s=\tau+1}^t \lambda_s \right) f_{\tau,i} w_{\tau,i} \quad (12)$$

where λ_t , which ranges in $[0,1]$, is a time dependent discount factor introduced for forgetting the effect of old, possibly outdated values. Observe that for low values of λ_t , the influence of old data decreases progressively, so that they are forgotten along time. This forgetting effect of old data is attenuated when λ_t approaches 1: in this case, old and new data have the same influence in the sum. As learning proceeds and data values become more stable, forgetting them is no more required and λ_t can be made to progressively approach 1 to reach convergence.

The discounted sum $W_{t,i}$ can be interpreted as the accumulated number of samples (composed of fractions $w_{t,i}$) attributed to unit i along time, with forgetting. Similarly, $X_{t,i}$ corresponds to the accumulated

sum with forgetting of sample vectors \mathbf{x} attributed to unit i , which is used to derive the mean vector μ_i . In the same way, $(XX)_{t,i}$ accumulates the matrices of covariances of the vectors \mathbf{x} attributed to unit i , which are used to calculate the covariance matrix Σ_i .

From (12), we obtain the recursive formula:

$$[[f]]_{t,i} = \lambda_t [[f]]_{t-1,i} + f_{t,i} w_{t,i} \quad (13)$$

When a new sample \mathbf{x}_t arrives, all the accumulators are updated with the incremental formula (13), and new estimators for the GMM parameters are obtained as:

$$\alpha_{t,i} = \frac{W_{t,i}}{\sum_{j=1}^K W_{t,i}} \quad (14)$$

$$\mu_{t,i} = \frac{X_{t,i}}{W_{t,i}} \quad (15)$$

$$\Sigma_{t,i} = \frac{(XX)_{t,i}}{W_{t,i}}. \quad (16)$$

The approximation capabilities of a GMM depend on the number K of Gaussians of the mixture. Since we can not determine the most appropriate number beforehand, the number of Gaussians can be incremented on-line. We delay the explanation of the process for unit generation until Section 6.2.

5 DEALING WITH BIASED SAMPLING

In the incremental formula (13), the factor λ_t is used to progressively replace (forget) old data by new arrived ones in a smooth way. This is the desired effect when data are presented in a statistically unbiased way, so that all past entries are equally forgotten at the arrival of each new sample. However, in RL, data are not uniformly distributed on the state-action space, but are obtained along the trajectories followed by the agent, and are particularly biased toward the good-valued regions due to the need for exploitation. That is, convergence regions are more frequently sampled, causing their densities to increase, but this is at the expense of lowering densities and forgetting data in other regions. This is undesirable since, statistics in regions with low Q values, and hence sparsely sampled, will get their data lost. This can be seen by setting $w_{t,i} = 0$ in equation (13), what yields:

$$[[f]]_{t,i} = \lambda_t [[f]]_{t-1,i}, \quad (17)$$

showing that the accumulators of units that are seldom activated will decay to 0. To avoid this we modify the updating formula (13) in this way,

$$[[f]]_{t,i} = \lambda_t^{w_{t,i}} [[f]]_{t-1,i} + f_{t,i} w_{t,i}. \quad (18)$$

In this updating formula, the power $w_{t,i}$ prevents undesired changes in the parameters of the Gaussians which are not responsible of generating the observed values. Thus, if we make $w_{t,i} = 0$ in (18), what we get is:

$$[[f]]_{t,i} = [[f]]_{t-1,i}, \quad (19)$$

so that the values of the statistics of the inactive units remain unchanged. On the other hand, in the limit case of $w_{t,i} = 1$, corresponding to a full activation of unit i , the effect of the new updating formula is the same as before:

$$[[f]]_{t,i} = \lambda_t [[f]]_{t-1,i} + f_{t,i} \quad (20)$$

Therefore, we will adopt the updating formula (18) that keeps better track of less explored regions.

6 Q-LEARNING WITH A GMM

In the case of Q -Learning, samples are of the form $\mathbf{x}_t = (\mathbf{s}_t, a_t, q(\mathbf{s}_t, a_t))$, consisting of the visited state \mathbf{s}_t , the executed action a_t , and the estimated value of $q(\mathbf{s}_t, a_t)$ as given by eq. (1). To obtain this estimation we need to evaluate $\max_a Q(\mathbf{s}_{t+1}, a)$, where $Q(\mathbf{s}, a)$ is defined as the expected value of q given \mathbf{s} and a for the joint probability distribution provided by the GMM:

$$p(\mathbf{s}, a, q) = \sum_{i=1}^K \alpha_i \mathcal{G}(\mathbf{s}, a, q; \mu_i, \Sigma_i), \quad (21)$$

$$Q(\mathbf{s}, a) = E[q|\mathbf{s}, a] = \mu(q|\mathbf{s}, a) \quad (22)$$

To compute this, we must first obtain the distribution $p(q|\mathbf{s}, a)$. Decomposing the covariances Σ_i and means μ_i in the following way:

$$\mu_i = \begin{pmatrix} \mu_i^{(\mathbf{s}, a)} \\ \mu_i^q \end{pmatrix} \quad (23)$$

$$\Sigma_i = \begin{pmatrix} \Sigma_i^{(\mathbf{s}, a)(\mathbf{s}, a)} & \Sigma_i^{(\mathbf{s}, a)q} \\ \Sigma_i^{q(\mathbf{s}, a)} & \Sigma_i^{qq} \end{pmatrix}, \quad (24)$$

the probability distribution of q , for the given state \mathbf{s} and a tentative action a , can then be expressed as:

$$p(q|\mathbf{s}, a) = \sum_{i=1}^K \beta_i(\mathbf{s}, a) \mathcal{G}(q; \mu_i(q|\mathbf{s}, a), \sigma_i(q)) \quad (25)$$

where,

$$\mu_i(q|\mathbf{s}, a) = \mu_i^q + \Sigma_i^{q(\mathbf{s}, a)} \left(\Sigma_i^{(\mathbf{s}, a)(\mathbf{s}, a)} \right)^{-1} \left((\mathbf{s}, a) - \mu_i^{(\mathbf{s}, a)} \right) \quad (26)$$

$$\sigma_i^2(q) = \Sigma_i^{qq} - \Sigma_i^{q(\mathbf{s}, a)} \left(\Sigma_i^{(\mathbf{s}, a)(\mathbf{s}, a)} \right)^{-1} \Sigma_i^{(\mathbf{s}, a)q} \quad (27)$$

$$\beta_i(\mathbf{s}, a) = \frac{\alpha_i \mathcal{G}(\mathbf{s}, a; \mu_i^{(\mathbf{s}, a)}, \Sigma_i^{(\mathbf{s}, a)})}{\sum_{j=1}^K \alpha_j \mathcal{G}(\mathbf{s}, a; \mu_j^{(\mathbf{s}, a)}, \Sigma_j^{(\mathbf{s}, a)})}. \quad (28)$$

From (25) we can obtain the conditional mean and variance, $\mu(q|\mathbf{s}, a)$ and $\sigma^2(q|\mathbf{s}, a)$, of the mixture at a point (\mathbf{s}, a) as:

$$\mu(q|\mathbf{s}, a) = \sum_{i=1}^K \beta_i(\mathbf{s}, a) \mu_i(q|\mathbf{s}, a) \quad (29)$$

$$\sigma^2(q|\mathbf{s}, a) = \sum_{i=1}^K \beta_i(\mathbf{s}, a) (\sigma_i^2(q) + (\mu_i(q|\mathbf{s}, a) - \mu(q|\mathbf{s}, a))^2) \quad (30)$$

Equation (29) is the estimated Q value for a given state and action, while (30) is its estimated variance. Our purpose was to find the action that maximizes $Q(\mathbf{s}, a)$. Unfortunately, this cannot be done analytically, but an approximated value can be obtained by numerical techniques. In our implementation, we take the approach of computing $Q(\mathbf{s}, a)$ for a finite number of actions, and taking the action that provides the largest Q value as the approximated maximum.

6.1 Action Selection

If our purpose is to greedily exploit the policy learned so far, as defined in (2), we must execute the action a corresponding to the maximum $Q(\mathbf{s}, a)$ value for the current state \mathbf{s} , obtained as just explained. However, during learning, exploratory actions are also necessary. This is the classical exploration/exploitation tradeoff. Instead of following an undirected exploration method (using the nomenclature of (Dearden et al., 1998)), we make use of the uncertainty in the estimation of the Q values to balance the information gain obtained from exploration, with the reward gain of executing the optimal action. For this, to each evaluated action, we assign a Q value obtained stochastically from a normal distribution with mean (29) and variance (30), and select the action that provided the highest value. This exploration strategy increases the probability of executing exploratory actions when predictions are less certain, but this probability decreases when the mean value is lower.

6.2 Unit Generation

Since the main purpose of our GMM is to represent the Q function, the generation of new Gaussians is principally driven by the need to better approximate the set of observed q values. Thus, a new Gaussian is generated when the two following conditions are satisfied: 1) The estimation error of the observed q

value is larger than a predefined value δ , and 2) The sample density in the experienced instance is below a threshold ρ . These criteria are expressed as:

$$(q(\mathbf{s}, a) - \mu(q|\mathbf{s}, a))^2 \geq \delta \quad (31)$$

and

$$p(\mathbf{s}, a, q) = \sum_{i=1}^K \alpha_i \mathcal{G}(\mathbf{s}, a, q; \mu_i, \Sigma_i) \leq \rho \quad (32)$$

Whenever both criteria are fulfilled, a Gaussian is generated with parameters given by

$$W_{K+1} = 1 \quad (33)$$

$$\mu_{K+1}(\mathbf{s}, a, q) = (\mathbf{s}_t, a_t, q(\mathbf{s}_t, a_t)) \quad (34)$$

$$\Sigma_{K+1} = C \text{diag}\{d_1, \dots, d_D, d_a, d_q\}, \quad (35)$$

where d_i is the total range size of variable i ; D is the dimension of the state space; and C is a positive value defining the dispersion of the new Gaussian. The imposition of the second condition is to avoid a proliferation of Gaussians in densely sampled regions.

7 EXPERIMENTS

To demonstrate the performance of the method we apply it to the classical benchmark problem of swinging up and stabilizing an inverted pendulum with limited torque (Doya, 2000). The task consists in swinging the pendulum until reaching the upright position and then stay there indefinitely. The optimal policy for this problem is not trivial since, due to the limited torques available, the controller has to swing the pendulum several times until its kinetic energy is large enough to overcome the load torque and reach the upright position.

The state space is two-dimensional and is configured by the angular position θ and angular velocity $\dot{\theta}$. We take advantage of the symmetry of the problem by identifying states with inverted angular position and velocity: $(\theta, \dot{\theta}) \sim (-\theta, -\dot{\theta})$. The GMM for density estimation is defined in the four-dimensional joint space $\mathbf{x}=(\theta, \dot{\theta}, a, q)$. As the reward signal (or, in this case, the cost that we have to minimize) we simply take the absolute value of the angle of the pendulum from its top position: $\text{cost}(\theta, \dot{\theta}) = -r(\theta, \dot{\theta}) = |\theta|$ which ranges in the interval $[0, \pi]$. The discount coefficient γ in equation 1 is set to 0.85.

The setting of our system consists in the following. We provide the system with 10 initial Gaussians. The elements of the mean μ_i of the mixture component i are selected randomly, except for the q variable that is initialized to the maximum possible value to

favor exploration of unvisited regions. The initial covariance matrices Σ_i are diagonal and the variance for each variable is set to one tenth of the total span of its range. Each Gaussian is initialized with an accumulated number of samples $W_i = 0.1$. This small value makes the component i to have little influence in the estimation while there is no, or little, updating.

The discount factor λ_t for the computation of the discounted sums (Section 4) takes values from the equation

$$\lambda_t = 1 - 1/(at + b), \quad (36)$$

where b fixes the initial value of λ_t and a determines its growth rate toward 1. In our experiments we set $a = 0.001$, and the value of b depends on whether the updating formula (13) or (18) is used. In the first case, $b = 10$, and when (18) is used, $b = 1000$ to compensate for the effect of the exponent $w_{t,i} < 1$. For the only purpose of the following discussion, we will refer to updates performed according to (13) as *simply local updating*, while updates performed according to (18) as *exponentially local updating*.

We performed the experiments using episodes of 50 seconds with an actuation interval of 0.1 seconds. At the beginning of each episode, the pendulum is placed in the hang-down position. At the end of each episode, a test of 30 seconds is performed exploiting the policy learned so far. As the result of the test we take the sum of the costs obtained at each time step of 0.1 sec.

Figure 1 shows the performance of the learning system when using simply local updating. The graphic corresponds to the average of 50 independent experiments of 100 episodes each. The system shows a stable performance after about 60 episodes, and reaches a total accumulated cost of around 100. The best performance obtained by exhaustive manual tuning was near 44, and corresponds to the line at the lower part of the figure. The best result obtained by a single experiment test was 44.42. In average, the total number of Gaussians at the end of the experiments was about 84.

In order to compare these results with those of (Riedmiller, 2005b), we will take into account the total number of times the system needs to be updated with a sample to achieve the control. Riedmiller reports that the swing-up and balance task required 100 iterations of the NFQ algorithm, each one requiring 1000 epochs of batch learning with the Rprop learning algorithm to train the neural net with an unspecified number D of samples. This gives a total of $100,000 \times D$ sample updates. In our case, good control is obtained after approximately $60 \times 500 = 30,000$ updates, which is significantly better.

These results were obtained with the simply lo-

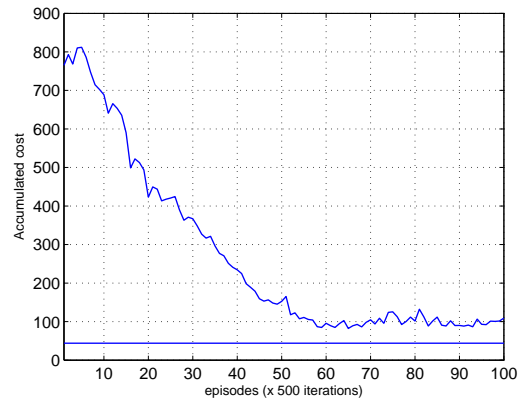


Figure 1: Average over 50 experiments of the accumulated cost, on tests of 30 seconds, with simply local updating.

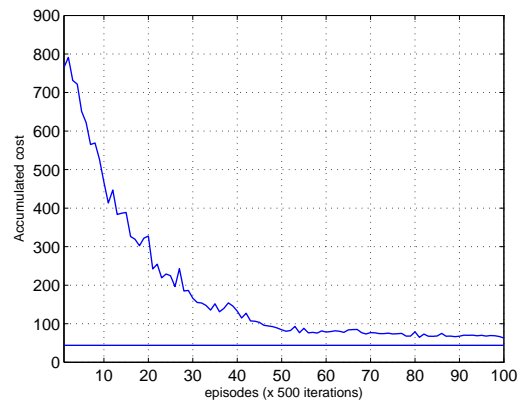


Figure 2: Average over 50 experiments of the accumulated cost, on tests of 30 seconds, with exponentially local updating.

cal updating of formula (13), which is sensible to the effect of the biased sampling. Despite being good results, we observe that the learning curve presents some fluctuations that prevent the system to converge to a value nearer to the theoretical optimum. Such fluctuations are caused by transient learning phases during which the system is not able to swing the pendulum up, until a good policy is recovered again. This is precisely the problem we anticipated: as far as the system stabilizes near a good policy, it experiences samples mostly near the optimal policy, so that the Q estimation of less experienced actions degrades, and eventually, suboptimal actions gain temporary control until the system relearns their correct value. This is the reason by which we introduced the exponent $w_{t,i}$ in the update formula (18) for exponentially local updating. Its effectiveness is shown in Figure 2.

Results show that exponentially local updating achieves convergence slightly faster and with a much

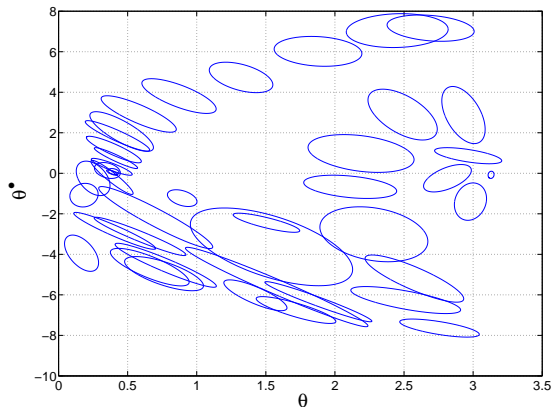


Figure 3: Distribution of Gaussians in a projection of the joint space to the state space.

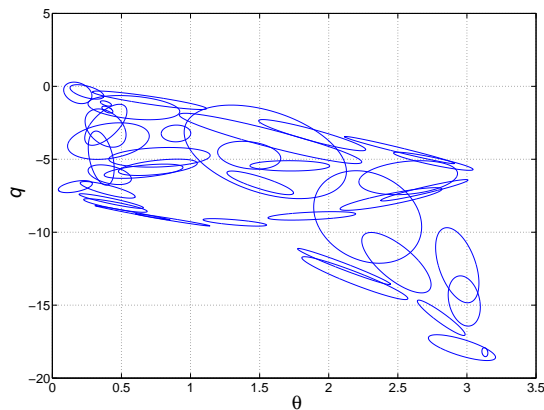


Figure 4: Distribution of Gaussians in a projection of the joint space to the (θ, q) space.

more stable behavior. In this case the average cost is reduced to near 64, that is just 20 above the theoretical optimum, which compared with the $100 - 44 = 56$ of the simply local updating corresponds to an improvement by a factor between 2 and 3. The number of Gaussians used in this case is also lowered to less than 60, in average. To check the effectiveness of exponentially local updating to prevent forgetting, we observe that in the course of the 50 experiments, the system failed to swing-up the pendulum after the 50th episode only in a single occasion, compared with far more than 30 with simply local updating.

Figures 3 and 4 show two projections of the Gaussians of a typical GMM obtained for this problem after training. It can be seen that they are not equally distributed along the whole configuration space, but concentrate in the most common trajectories of the system, what constitutes an efficient use of resources.

8 CONCLUSIONS

We proposed a new approach for Q -Learning in continuous state-action spaces, in which a Gaussian Mixture Model that estimates the probability density in the joint state-action- Q value space is used for function approximation. From this joint distribution we can obtain, not just the expected value of Q for a given state and action, but a full probability distribution that is used to define a directed exploration-exploitation strategy. As a further benefit, from the density estimation in the joint space we can also obtain the sampling density in the state-action space. This information is used to remedy the problem of biased sampling inherent to on-line Reinforcement Learning. For this, we modified the incremental updating rule of an on-line EM algorithm in order to avoid forgetting data of less frequently sampled regions, even when exploration is reiteratively done near the goal configurations.

Tests performed on a classical RL problem, the swing-up and balance of an inverted pendulum, show that our approach improves the results of previous works when considering the number of sample updates required to achieve the goal. The comparison between our basic approach, using simply local updating, and the proposed improvement using exponentially local updating, shows that the approach is effective in reducing the perturbing effect of biased sampling.

Finally, we want to point out that the information provided by the density estimation has not been fully exploited yet. We expect to use the density information in future works to better guide exploration during learning.

ACKNOWLEDGEMENTS

This research was partially supported by Consolider Ingenio 2010, project CSD2007-00018.

REFERENCES

- Arandjelovic, O. and Cipolla, R. (2005). Incremental learning of temporally-coherent gaussian mixture models. In *Technical Papers - Society of Manufacturing Engineers (SME)*.
- Bellman, R. and Dreyfus, S. (1962). *Applied Dynamic Programming*. Princeton University Press, Princeton, New Jersey.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

- Dearden, R., Friedman, N., and Russell, S. (1998). Bayesian q-learning. In *In AAAI/IAAI*, pages 761–768. AAAI Press.
- Dempster, A., Laird, N., Rubin, D., et al. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.
- Doya, K. (2000). Reinforcement learning in continuous time and space. *Neural Comput.*, 12(1):219–245.
- Duda, R. O., Hart, P. E., and Stork, D. G. (2001). *Pattern classification*. John Wiley and Sons, Inc, New-York, USA.
- Engel, Y., Mannor, S., and Meir, R. (2005). Reinforcement learning with gaussian processes. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 201–208, New York, NY, USA. ACM.
- Ernst, D., Geurts, P., and Wehenkel, L. (2005). Tree-based batch mode reinforcement learning. *J. Mach. Learn. Res.*, 6:503–556.
- Figueiredo, M. (2000). On gaussian radial basis function approximations: Interpretation, extensions, and learning strategies. *Pattern Recognition, International Conference on*, 2:618–621.
- Ghahramani, Z. and Jordan, M. (1994). Supervised learning from incomplete data via an em approach. In *Proceeding of Advances in Neural Information Processing Systems (NIPS'94)*, pages 120–127. San Mateo, CA: Morgan Kaufmann.
- Gordon, G. J. (1995). Stable function approximation in dynamic programming. In *ICML*, pages 261–268.
- Neal, R. and Hinton, G. (1998). A view of the em algorithm that justifies incremental, sparse, and other variants. In *Proceedings of the NATO Advanced Study Institute on Learning in graphical models*, pages 355–368, Norwell, MA, USA. Kluwer Academic Publishers.
- Nowlan, S. J. (1991). *Soft competitive adaptation: neural network learning algorithms based on fitting statistical mixtures*. PhD thesis, Pittsburgh, PA, USA.
- Ormonet, D. and Sen, S. (2002). Kernel-based reinforcement learning. *Machine Learning*, 49(2-3):161–178.
- Rasmussen, C. and Kuss, M. (2004). Gaussian processes in reinforcement learning. *Advances in Neural Information Processing Systems*, 16:751–759.
- Riedmiller, M. (2005a). Neural fitted Q iteration—first experiences with a data efficient neural reinforcement learning method. *Lecture notes in computer science*, 3720:317–328.
- Riedmiller, M. (2005b). Neural Reinforcement Learning to Swing-up and Balance a Real Pole. In *Proceedings of the 2005 IEEE International Conference on Systems, Man and Cybernetics*, volume 4, pages 3191–3196.
- Rottmann, A. and Burgard, W. (2009). Adaptive autonomous control using online value iteration with gaussian processes. In *Proceedings of the 2009 IEEE International Conference on Robotics and Automation (ICRA'09)*, pages 2106–2111.
- Sato, M.-A. and Ishii, S. (2000). On-line em algorithm for the normalized gaussian network. *Neural Comput.*, 12(2):407–432.
- Song, M. and Wang, H. (2005). Highly efficient incremental estimation of gaussian mixture models for online data stream clustering. In *Proceedings of SPIE: Intelligent Computing: Theory and Applications III*, pages 174–183, Orlando, FL, USA.
- Sutton, R. and Barto, A. (1998). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA.
- Watkins, C. and Dayan, P. (1992). Q-learning. *Machine Learning*, 8(3-4):279–292.

PARAMETER TUNING BY SIMPLE REGRET ALGORITHMS AND MULTIPLE SIMULTANEOUS HYPOTHESIS TESTING

Amine Bourki**, Matthieu Coulm**, Philippe Rolet*, Olivier Teytaud* and Paul Vayssière**

*TAO, Inria, Umr CNRS 8623, Univ. Paris-Sud, 91405 Orsay, France

**EPITA, 16 rue Voltaire, 94270 Le Kremlin-Bicêtre, France

{amine.bourki, mcoulm}@gmail.com, philippe.rolet@lri.fr

olivier.teytaud@inria.fr, paul.vayssiere@gmail.com

Keywords: Simple regret, Automatic parameter tuning, Monte-Carlo tree search.

Abstract: “Simple regret” algorithms are designed for noisy optimization in unstructured domains. In particular, this literature has shown that the uniform algorithm is indeed optimal asymptotically and suboptimal non-asymptotically. We investigate theoretically and experimentally the application of these algorithms, for automatic parameter tuning, in particular from the point of view of the number of samples required for “uniform” to be relevant and from the point of view of statistical guarantees. We see that for moderate numbers of arms, the possible improvement in terms of computational power required for statistical validation can’t be more than linear as a function of the number of arms and provide a simple rule to check if the simple uniform algorithm (trivially parallel) is relevant. Our experiments are performed on the tuning of a Monte-Carlo Tree Search algorithm, a great recent tool for high-dimensional planning with particularly impressive results for difficult games and in particular the game of Go.

1 INTRODUCTION

We consider the automatic tuning of new modules. It is quite usual, in artificial intelligence, to design a module, for which there are several free parameters. This is natural in supervised learning, optimization (Nannen and Eiben, 2007b; Nannen and Eiben, 2007a), control (Lee et al., 2009; Chaslot et al., 2009). We will here consider the particular case of Monte-Carlo Tree Search (Chaslot et al., 2006; Coulom, 2006; Kocsis and Szepesvari, 2006; Lee et al., 2009).

Consider a program, in which a new module with parameter $\theta \in \{1, \dots, K\}$ has been added. In the bandit literature, $\{1, \dots, K\}$ is referred to as the set of arms. Then, we’re looking for the best parameter $\theta \in \{1, \dots, K\}$ for some performance criterion; the performance criterion $L(\theta)$ is stochastic. We have a finite time budget T (also termed horizon), we can have access to T realizations of $L(\theta_1), L(\theta_2), \dots, L(\theta_T)$ and we then choose some $\hat{\theta}$. The game is as follows:

- The algorithm chooses $\theta_1 \in \{1, \dots, K\}$.
- The algorithm gets a realization r_1 distributed as $L(\theta_1)$.
- The algorithm chooses $\theta_2 \in \{1, \dots, K\}$.

- The algorithm gets a realization r_2 distributed as $L(\theta_2)$.
- ...
- The algorithm chooses $\theta_T \in \{1, \dots, K\}$.
- The algorithm gets a realization r_T distributed as $L(\theta_T)$.
- The algorithm chooses $\hat{\theta}$.
- The loss is $r_T = \max_{\theta} \mathbb{E}L(\theta) - \mathbb{E}L(\hat{\theta})$.

The performance measure is the simple regret (Bubeck et al., 2009), i.e. $r_T = \max_{\theta} \mathbb{E}L(\theta) - \mathbb{E}L(\hat{\theta})$, and we want to minimize it. Then main difference with noisy nonlinear optimization is that we don’t use any structure on the domain.

We point out the link with No Free Lunch theorems (NFL (Wolpert and Macready, 1997)), which claim that all algorithms are equivalent when no prior knowledge can be explored. Yet, there are some differences in the framework: NFL considers deterministic optimization, in which testing several times the same point is meaningless. We here consider noisy optimization, with a small search space: all the difficulty is in the statistical validation, for choosing which points in the search space should be tested more intensively.

Useful notations:

- $\#E$ is the cardinal of the set E ;
- $N_t(i)$ is the number of times the parameter i has been tested at iteration t , i.e.

$$N_t(i) = \#\{j \leq t; \theta_j = i\}.$$

- $\hat{L}_t(i)$ is the average reward for parameter i at iteration t , i.e.

$$\hat{L}_t(i) = \frac{1}{N_t(i)} \sum_{j \leq t; \theta_j = i} r_j.$$

(well defined if $N_t(i) > 0$)

Section 2 recalls the terminology of simple regret and discusses the relevance for Automatic Parameter Tuning (APT). Section 3 mathematically considers the statistical validation, which was not yet, to the best of our knowledge, considered for simple regret algorithms; we will in particular show that the dependency of the computational cost as a function of the number of tested parameter values is at best linear, and therefore it is not possible to do better than this linear improvement in terms of statistical validation - we will then switch to experimental analysis, and we'll show that the improvement is indeed improved by far less than a linear factor in our real world setting (section 4).

2 SIMPLE REGRET: STATE OF THE ART AND RELEVANCE FOR AUTOMATIC PARAMETER TUNING

We consider the case in which $L(\theta)$ is, for all θ , a Bernoulli distribution. (Bubeck et al., 2009) states that (i) the naive algorithm distributing θ_i uniformly among the possible parameters, i.e. $\theta_i = \text{mod}(i, K) + 1$ with mod the modulo operator, with $\hat{\theta} = \arg \max_i \hat{L}(i)$, has simple regret

$$\mathbb{E}r_T = O(\exp(-c \cdot T)) \quad (1)$$

for some constant c depending on the Bernoulli parameters (more precisely, on the difference between the parameters of the best arm and of the other arms). This is for $\hat{\theta}$ maximizing the empirical reward, i.e.

$$\hat{\theta} \in \arg \min_{\theta} \hat{L}_T(\theta)$$

and this is proved optimal.

If we consider distribution-free bounds (i.e. for a fixed T , we consider the supremum of $\mathbb{E}r_T$ for

all Bernoulli parameters), then (Bubeck et al., 2009) shows that, with the same algorithm,

$$\sup_{\text{distribution}} \mathbb{E}r_T = O(\sqrt{K \log K / T}), \quad (2)$$

where the constant in the $O(\cdot)$ is a universal constant; Eq. 2 is tight within logarithmic factors of K ; there's a lower bound for all algorithms of the form.

$$\sup_{\text{distribution}} \mathbb{E}r_T = \Omega(\sqrt{K/T}).$$

Importantly, the best known upper bounds for variants of UCB (Auer et al., 2002) are significantly worse than Eq. 1 (the simple regret is then only polynomially decreasing) and significantly worse than Eq. 2 (by a logarithmic factor of T) - see (Bubeck et al., 2009) for more on this.

However, it is clearly shown also in (Bubeck et al., 2009) that for small values of T , using a variant of UCB for choosing the θ_i and $\hat{\theta}$ is indeed much better than uniform sampling. The variant of UCB is as follows, for some parameter $\alpha > 1$:

$$\hat{\Theta}_t = \arg \max_i N_t(i).$$

$$\Theta_i = \text{mod}(i, K) + 1 \text{ if } i \leq K$$

$$\Theta_i = \arg \max_i \hat{L}_t(i) + \sqrt{\alpha \log(t-1) / N_{t-1}(i)} \text{ otherwise.}$$

Simple regret is a natural criterion when working on automatic parameter tuning. However, the theoretical investigations on simple regret did not answer the following question: how can we validate an arm selected by a simple regret algorithm when a baseline is present? In usual cases, for the application to parameter tuning, we know the score before a modification, and then we tune the parameters of the optimization: we don't only tune, we validate the tuned modification; this question is nonetheless central in many applications in particular when modifications are included automatically by the tuning algorithm (Nannen and Eiben, 2007b; Nannen and Eiben, 2007a; Hoock and Teytaud, 2010). We'll see in next sections that the naive solution, consisting in testing separately each arm, is not so far from being optimal.

3 MULTIPLE SIMULTANEOUS HYPOTHESIS TESTING IN AUTOMATIC PARAMETER TUNING

As pointed out above, a goal different from minimizing the simple regret consists in finding a good arm could be (i) finding a good arm if any (ii)

avoiding selecting a bad arm if there's no good arm (no arm which outperforms the baseline). We'll briefly show how to apply Multiple Simultaneous Hypothesis Testing (MSHT), and in particular its simplest and most well known variant termed the Bonferroni correction, to Automatic Parameter Tuning. MSHT(Holm, 1979; Hsu, 1996) is very classical in neuro-imagery(Pantazis et al., 2005), bioinformatics, tuning of optimizers(Nannen and Eiben, 2007b; Nannen and Eiben, 2007a).

MSHT consists in statistically testing several hypothesis in same time: for example, when 100 sets of parameters are tested simultaneously, then, whenever each set is tested with confidence 95%, and whenever all sets of parameters have no impact on the result, then with probability $1 - (1 - 0.05^{100}) \simeq 99.4\%$ at least one set of parameters will be validated. MSHT is aimed at correcting this effect, so that taking into account the multiplicity of tests we can have modified tests so that the overall risk remains lower than 5%.

Assume that we expect arms with standard deviation σ (we'll see that for our applications, σ is usually nearly known in advance; it can also be estimated dynamically during the process). Then, the standard Gaussian approximation says that with probability 90%¹, the difference between $\hat{L}_t(\theta)$ and $L_t(\theta)$ for arm θ is lower than $1.645\sigma/\sqrt{N_t(i)}$: with probability 90%,

$$|\hat{L}_t(\theta) - L_t(\theta)| \leq 1.645\sigma/\sqrt{N_t(i)}. \quad (3)$$

The constant 1.645 directly corresponds to the Gaussian probability distribution (the precise value is $\Phi^{-1}((1+0.9)/2) = 1.645$); a Gaussian standard distribution is ≤ 1.645 in absolute value with probability 90%. If we consider several tests simultaneously, i.e. we consider K arms, then Eq. 3 becomes Eq. 4: with probability 90%,

$$\forall \theta \in \{1, 2, \dots, K\} |\hat{L}_t(\theta) - L_t(\theta)| \leq t_K \sigma / \sqrt{N_t(i)} \quad (4)$$

where, with the so-called Bonferroni correction, $t_K = -\Phi^{-1}(0.05/K)$ where Φ is the normal cumulative distribution function². This is usually estimated with

$$\frac{\exp(-t_K^2)}{t_K \sqrt{2\pi}} = 0.05/K \quad (5)$$

and therefore if we expect improvements of size δ , we can only validate a modification with confidence 90%

¹The constant 90% is arbitrary; it means that we decide that results are guaranteed within risk 10%.

²Note that a tighter formula is $t_K = -\Phi^{-1}(1 - (1 - 0.05)^K)$; this holds thanks to independence of the different arms.

with n experiments per arm if t_K solving Eq. 5 verifies $t_K \sigma / \sqrt{n} \leq \delta$; a succinct equation for this is

$$s = \delta \sqrt{n} / \sigma \quad (6)$$

$$\frac{\exp(-s^2)}{s \sqrt{2\pi}} \leq 0.05/K \quad (7)$$

This shows that for other quantities fixed, n has a logarithmic dependency as a function of K .

A numerical application for $\delta = 0.02$, $K = 49$ and $\sigma = \frac{1}{2}$ is

$$s = 0.04 \sqrt{n}, \quad \frac{\exp(-s^2)}{s \sqrt{2\pi}} = 0.05/49.$$

which implies $n \geq 3219$; this implies that for our confidence interval, we require 3219 runs per arm (i.e. $\inf_{\theta} N_T(\theta) \geq 3219$). We'll see that this number is consistent with our numerical experiments later. Interestingly, with only one arm, i.e. $K = 1$, we get $n \geq 1117$; this is not so much better, and suggests that whatever we do, it will be difficult to get significant results with subtle techniques for pruning the set of arms: if there is only one arm, we can only divide the computational cost for this arm by $O(\log(K))$. In case of perfect pruning, n is also naturally multiplied by K (as all the computational power is spent on only one arm instead of K arms); this provides an additional linear factor, leading to a roughly linear improvement in terms of computational power as a function of the number of arms, in case of perfect pruning.

Bernstein Races

This paper is devoted to the use of simple regret algorithms to APT, compared to the most simple APT algorithm, namely uniform sampling (which is known asymptotically optimal for simple regret); Bernstein races are therefore beyond the scope of this paper. Nonetheless, as our results emphasize the success of uniform sampling (at least in some cases), we briefly discuss Bernstein races. In (Mnih et al., 2008; Hoock and Teytaud, 2010), Bernstein races were considered as tools for discarding statistically bad arms: this is equivalent to *Uniform*, except that tests as above are applied periodically, and statistically bad arms are discarded. This discards arms earlier than the uniform algorithm above which just checks the result at the end, but increases the quantity K involved in tests (as in Eqs. 6 and 7), *even if no arm can be rejected*. The fact that testing arms for discarding on the fly has a cost, whenever no arm is discarded, might be surprising at first view - it is a known effect that when multiple tests are performed, then the number of samples required for a same confidence rate on the result is much higher. This approach can therefore at

most divide the computational power by $K \log(K)$ before an arm is validated, and the computational power is indeed increases when no early discarding is possible. Nonetheless, this sound approach is probably the best candidate when the visualization is not crucial - *Uniform* can provide nice graphs as shown in the experimental section from <http://hal.inria.fr/inria-00467796/>.

4 EXPERIMENTAL VALIDATION: THE TUNING OF MOGO

Due to length constraints, the experimental section is reported to <http://hal.inria.fr/inria-00467796/>.

5 DISCUSSION

We have surveyed simple regret algorithms. They are noisy optimization algorithms, and they don't assume any structure on the domain. We compared *Uniform* (known as optimal for sufficiently large horizon, i.e. sufficiently large time budget) and *UCB* for automatic parameter tuning. Our results are as follows:

- **MSHT (even the Simple Bonferroni Correction) is relevant for Automatic Parameter Tuning.** It predicts how many computational power is required for *Uniform*; when the number K of tested sets of parameters depends on a discretization, MSHT can be applied for choosing the grain of the discretization. The *Uniform* approach combined with MSHT by Bonferroni correction might be the best approach when the computational power is large in front of K , thanks to its statistical guarantees, the easy visualization, the optimality in terms of simple regret. However, non-asymptotically, it is not optimal and the rule below is here for deciding the relevance of *Uniform* when K and T are known.
- **Choosing between the Naive Solution (*Uniform* sampling) and Sophisticated Algorithms.** The naive *Uniform* algorithm is provably optimal for large values of the horizon. We propose the following simple rule for choosing if it is worth using something else than the simple uniform sampling:

- Compute

$$s = \delta \sqrt{n} / \sigma.$$

where

- * δ is the amplitude of the expected change in reward;

- * σ is the expected standard deviation;
 - * n is the number of experiments you can perform for each arm with your computational power.
- Test if $\frac{\exp(-s^2)}{s\sqrt{2\pi}} \leq 0.05/K$ where K is the number of arms.
 - If yes, then uniform sampling is ok. Otherwise, you can try UCB-like algorithms (but, in that case, there's no statistical guarantee), or Bernstein races. At first view, our choice would be Bernstein races for an implementation aimed at automatically tuning and validating several modifications (as in (Hooock and Teytaud, 2010)) as soon as conditions above are not met by the computational power available; if the computational power available is strong enough, *Uniform* has nice visualization properties.
 - **What if Uniform Algorithms can't do it?** If K is not large, nothing can be much better than uniform; at most the required horizon can be divided by $K \log(K)$. What if K is large? *UCB* is probably much better when K is large. A drawback is that it does not include any statistical validation, and is not trivially parallel; therefore, classical algorithms derived far from the field of simple regret, like Bernstein races (Bernstein, 1924), might be more relevant. Bernstein races are close to the *Uniform* algorithm, except that they discard arms as early as possible (Mnih et al., 2008; Hooock and Teytaud, 2010) by performing statistical tests on the fly. A drawback is that Bernstein races do not provide a complete picture of the search space and of the fitness landscape as *Uniform*; also, if no arm can be discarded early, the horizon required for statistical validation is bigger than for *Uniform* as tests are performed during the run. Yet, Bernstein races might be the most elegant tool for doing better than *Uniform* as they adapt to various frameworks (Hooock and Teytaud, 2010): when many arms can be discarded easily, they will save up a lot of computational power.
 - **Results on our Application to MCTS.** For the specific application, the results were significant but moderate; however, it can be pointed out that many handcrafted modifications around Monte-Carlo Tree Search provide such small improvements of a few percents each. Moreover, as shown in (Hooock and Teytaud, 2010), improvements performed automatically by bandits can be applied incrementally, leading to huge improve-

ments once they are cumulated.

- **Comparing Recommendation Techniques: Most Played Arm is Better.** The empirically best arm and the most played arm in UCB are usually the same (this is not the case for various other bandit algorithms), and are much better than the “empirical distribution of play” technique. The most played arm and the empirical distribution of play obviously do not make sense for *Uniform*. Please note that it is known in other settings (see (Wang and Gelly, 2007)) that the most played arm is better (Wang and Gelly, 2007). MPA is seemingly a reliable tool in many settings.

A next experimental step is the automatic use of the algorithm for more parameters, or e.g. by extending automatically the neural network used in the Monte-Carlo Tree Search so that it takes into account more inputs: instead of performing one big modification, apply several modifications the one after the other, and tune them sequentially so that all the modifications can be visualized and checked independently. The fact that the small constant 0.1 was better in UCB is consistent with the known fact that tuned version of UCB (with p related to the variance) provides better results; using tuned-UCB might provide further improvements (Audibert et al., 2006).

ACKNOWLEDGEMENTS

This work has been supported by French National Research Agency (ANR) through COSINUS program (project EXPLO-RA No ANR-08-COSI-004), and grant No. ANR-08-COSI-007-12 (OMD project). It benefited from the help of Grid5000 for parallel experiments.

REFERENCES

- Audibert, J.-Y., Munos, R., and Szepesvari, C. (2006). Use of variance estimation in the multi-armed bandit problem. In *NIPS 2006 Workshop on On-line Trading of Exploration and Exploitation*.
- Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2/3):235–256.
- Bernstein, S. (1924). On a modification of chebyshev’s inequality and of the error formula of laplace. *Original publication: Ann. Sci. Inst. Sav. Ukraine, Sect. Math. I*, 3(1):38–49.
- Bubeck, S., Munos, R., and Stoltz, G. (2009). Pure exploration in multi-armed bandits problems. In *ALT*, pages 23–37.
- Chaslot, G., Hoock, J.-B., Teytaud, F., and Teytaud, O. (2009). On the huge benefit of quasi-random mutations for multimodal optimization with application to grid-based tuning of neurocontrollers. In *ESANN*, Bruges Belgium.
- Chaslot, G., Saito, J.-T., Bouzy, B., Uiterwijk, J. W. H. M., and van den Herik, H. J. (2006). Monte-Carlo Strategies for Computer Go. In Schobbens, P.-Y., Vanhoof, W., and Schwanen, G., editors, *Proceedings of the 18th BeNeLux Conference on Artificial Intelligence, Namur, Belgium*, pages 83–91.
- Coulom, R. (2006). Efficient selectivity and backup operators in monte-carlo tree search. In P. Ciancarini and H. J. van den Herik, editors, *Proceedings of the 5th International Conference on Computers and Games, Turin, Italy*.
- Holm, S. (1979). A simple sequentially rejective multiple test procedure. *scand. j. statistic.*, 6:65-70.
- Hoock, J.-B. and Teytaud, O. (2010). Bandit-based genetic programming. In *Accepted in EuroGP 2010*, LLNCS. Springer.
- Hsu, J. (1996). Multiple comparisons, theory and methods, chapman & hall/crc.
- Kocsis, L. and Szepesvari, C. (2006). Bandit based monte-carlo planning. In *15th European Conference on Machine Learning (ECML)*, pages 282–293.
- Lee, C.-S., Wang, M.-H., Chaslot, G., Hoock, J.-B., Rimmel, A., Teytaud, O., Tsai, S.-R., Hsu, S.-C., and Hong, T.-P. (2009). The Computational Intelligence of MoGo Revealed in Taiwan’s Computer Go Tournaments. *IEEE Transactions on Computational Intelligence and AI in games*.
- Mnih, V., Szepesvári, C., and Audibert, J.-Y. (2008). Empirical Bernstein stopping. In *ICML ’08: Proceedings of the 25th international conference on Machine learning*, pages 672–679, New York, NY, USA. ACM.
- Nannen, V. and Eiben, A. E. (2007a). Relevance estimation and value calibration of evolutionary algorithm parameters. In *International Joint Conference on Artificial Intelligence (IJCAI’07)*, pages 975–980.
- Nannen, V. and Eiben, A. E. (2007b). Variance reduction in meta-eda. In *GECCO ’07: Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 627–627, New York, NY, USA. ACM.
- Pantazis, D., Nichols, T. E., Baillet, S., and Leahy, R. (2005). A comparison of random field theory and permutation methods for the statistical analysis of MEG data. *Neuroimage*, 25:355–368.
- Wang, Y. and Gelly, S. (2007). Modifications of UCT and sequence-like simulations for Monte-Carlo Go. In *IEEE Symposium on Computational Intelligence and Games, Honolulu, Hawaii*, pages 175–182.
- Wolpert, D. and Macready, W. (1997). No Free Lunch Theorems for Optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82.

MASTERING INTELLIGENT CLOUDS

Engineering Intelligent Data Processing Services in the Cloud

Sergiy Nikitin, Vagan Terziyan and Michal Nagy

Industrial Ontologies Group, University of Jyväskylä, Mattilanniemi 1, Jyväskylä, Finland
{sergiy.nikitin, vagan.terziyan, minagy}@jyu.fi

Keywords: Agent Technology, Cloud Computing, Semantic Web, Cloud Services, Ubiware.

Abstract: Current Cloud Computing stack mainly targets three architectural layers: Infrastructure, Platform and Software. These can be considered as services for the respective layers above. The infrastructure layer is provided as a service for the platform layer and the platform layer is, in turn, a service for the Software layer. Agent platforms fit the “Platform as a service” layer within this stack. At the same time, innovative agent-oriented approaches to programming, open new possibilities for software design in the cloud. We introduce main characteristics of our pilot agent platform called UBIWARE and offer flexible servicing architecture within the cloud platform, where various components and systems can configure, run and reuse intelligent cloud services to provide higher degree of flexibility and interoperability for their applications.

1 INTRODUCTION

Fast development of network technologies has recently brought back to life business models with the “thin client” architecture. Powerful data centers connected to the internet via broadband networks can minimize IT-infrastructure of any company to a set of simple terminals with less demanding system requirements. All the software and data can reside on the data center side, making user access easy and location independent. The providers offer different payment schemes as “pay-per-use” or subscription-based, that seems to be advantageous compared to standard IT-infrastructure expenses. The approach has got a set of new features and a new branding name: “Cloud Computing” (Hayes, 2008; Foster, 2008).

Cloud management platforms provide API for management either through command line or a remote method calls. The API, however, is used mainly by system administrators, who take care of proper functioning of services within the cloud. Management of the cloud platform is considered as something that a system administrator should arrange and do. Usually administrators use batch files for managing routine tasks and resolving exceptional situations.

At the same time, more and more software architecting paradigms call for new approaches to software design and development, where software

components get a certain degree of self-awareness, when a component can sense its own state and act based on the state changes. The vision of Autonomic Computing (Kephart, 2003) proposes to handle the complexity of information systems by introducing self-manageable components, able to “run themselves.” The authors state, that self-aware components would decrease the overall complexity of large systems. The development of those may become a “nightmare of ubiquitous computing” due to a drastic growth of data volumes in information systems as well as heterogeneity of ubiquitous components, standards, data formats, etc. The Cloud Computing and Autonomic Computing paradigms will become complementary parts of global-scale information systems in the nearest future. Such a fusion sets the highest demands to the software architects because cloud platforms will have to provide self-management infrastructure for a variety of complex systems residing in the same cloud, separated virtually, but run physically on the same hardware. At the same time, the cloud platform itself may possess features of self-aware complex system. A variety of self-aware components of different nature (i.e. end-user oriented, infrastructure-oriented, etc.) will need a common mechanism for interoperability, as far as they may provide services to each other.

The vision of GUN – Global Understanding Environment (Terziyan, 2003, 2005; Kaykova et al.,

2005) has introduced a concept of “Smart Resource” and a notion of an environment where all resources can communicate and interact regardless of their nature. In GUN various resources can be linked to the Semantic Web-based environment via adapters (or interfaces), which include (if necessary) sensors with digital output, data structuring (e.g. XML) and semantic adapter components (e.g. XML to Semantic Web). Software agents are assigned to each resource and are assumed to be able to monitor data coming from the adapter about the state of the resource, make decisions on behalf of the resource, and to discover, request and utilize external help if needed. Agent technologies within GUN allow mobility of service components between various platforms, decentralized service discovery, utilization of FIPA communication protocols, and multi-agent integration/composition of services.

When applying the GUN vision, each traditional system component becomes an agent-driven “smart resource”, i.e. proactive and self-managing. This can also be recursive. For example, an interface of a system component can become a smart resource itself, i.e. it can have its own responsible agent, semantically adapted sensors and actuators, history, commitments with other resources, and self-monitoring, self-diagnostics and self-maintenance activities.

In this paper we introduce a flexible servicing architecture within the cloud platform, where various components and systems can configure, run and reuse intelligent cloud services to provide higher degree of flexibility and interoperability for their applications. We use our pilot agent platform developed in accordance with GUN vision called UBIWARE to show how cloud computing can expand platform functionality and at the same time how an agent platform can become a high-level service provisioning instrument in the cloud.

The paper is organized as follows: In the next Section we discuss architectures of state-of-the-art cloud computing platforms and explore the possibilities for the interoperability mechanisms. Section 3 presents the architecture of the semantic middleware agent platform and explores possible options of connectivity with the cloud. Section 4 describes the scenarios and the architecture of the agent-driven intelligent servicing platform for a cloud. In Section 5 we discuss related work and conclude in Section 6.

2 STATE OF THE ART IN CLOUD INTELLIGENCE PLATFORMS

Architecture of current Cloud Computing stack mainly targets three layers: Infrastructure, Platform and Software. These layers can be considered as services for the respective layers above. The infrastructure as a service (IaaS) is provided to the platform layer and the platform becomes a service (PaaS) for the software layer. And finally, the software as a service layer (SaaS) brings the topmost end-user web services to clients (see Figure 1).

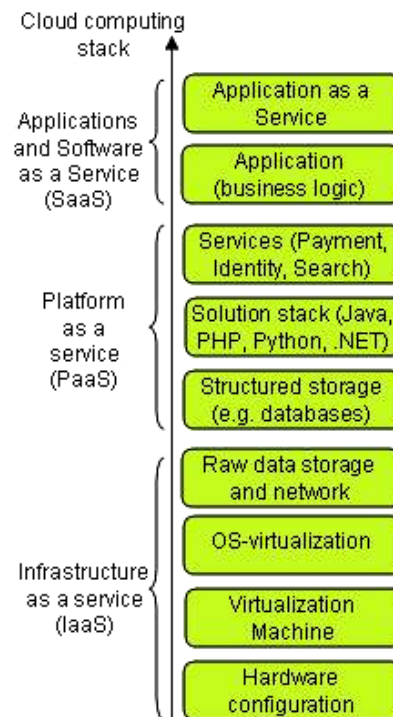


Figure 1: Cloud computing stack.

Cloud providers market niche is already a competitive field. Several big players are currently active in the market, e.g. Salesforce.com (SFDC), NetSuite, Oracle, IBM, Microsoft, Amazon EC2, Google Apps, etc. For a comprehensive survey of cloud computing systems see (Rimal et al., 2009). The services of the platform layer are in the scope of this work. In the next section we present a middleware platform and later introduce a new servicing approach in the cloud stack.

3 UBIWARE PLATFORM

UBIWARE has two main elements: an agent engine,

and S-APL – a Semantic Agent Programming Language (Katsonov and Terziyan, 2008) for programming of software agents within the platform.

The architecture of UBIWARE agent (Figure 2) consists of a *Live* behavior engine implemented in Java, a declarative middle layer, and a set of Java components – *Reusable Atomic Behaviors* (RABs).

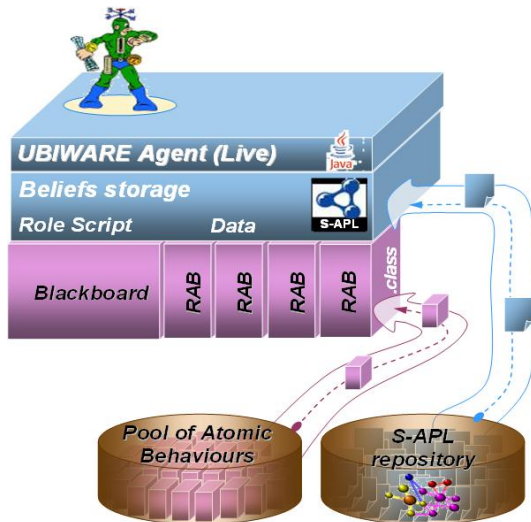


Figure 2: UBIWARE Agent.

RABs can be considered as sensors and actuators, i.e. components sensing or affecting the agent’s environment, but are not restricted to these. A RAB can also be a reasoner (data processor) if some of the logic needed is not efficient or possible to realize with the S-APL means, or if one wants to enable an agent to do some other kind of reasoning beyond the rule-based one. UBIWARE agent architecture implies that a particular UBIWARE-based software application will consist of a set of S-APL documents (data and behavior models) and a set of specific atomic behaviors needed for this particular application. Since reusability is an important UBIWARE concern, it is reasonable that the UBIWARE platform provides some of those ready-made.

Therefore, logically the UBIWARE platform, consists of the following three elements:

- The Live behavior engine
- A set of “standard” S-APL models
- A set of “standard” RABs

The extensions to the platform are exactly some sets of such “standard” S-APL models and RABs that can be used by the developers to embed into their applications certain UBIWARE features.

As Figure 2 shows, an S-APL agent can obtain the needed data and rules not only from local or

network documents, but also through querying S-APL repositories. Such a repository, for example, can be maintained by some organization and include prescriptions (lists of duties) corresponding to the organizational roles that the agents are supposed to play.

Technically, the implementation is built on top of the JADE – Java Agent Development Framework (Bellifemine et al. 2007), which is a Java implementation of IEEE FIPA specifications.

4 MASTERING INTELLIGENT CLOUD PLATFORM

Cloud computing providers offer various stack configurations with different sets of software and services inside. Theoretically, one can buy any configuration from the cloud provider; however this configuration will have nothing to do with the already running business logic of the customer. The application scenarios a customer wants to run will have to be adjusted. For example, consider a case, when a customer buys a virtual server with the MySQL database installed and a Java solution stack available. On top of this stack customer runs a workflow engine, e.g. BPEL-based. The user will have to install the engine, and then adjust local data storage settings (passwords, tables, queries). Then the process descriptions (BPEL files) should be adjusted to work with local settings. In some cases this process may be avoided if the cloud stack is identical to the customer’s environment, and if the all code was developed as portable. But what if the cloud stack slightly differs, but the prices are very attractive? Then customers may need to spend resources for solution code porting.

We propose architecture for a generic stack extension that allows users and platform providers to:

- Smoothly integrate with the infrastructure
- Build stack-independent solutions
- Automate reconfiguration of the solutions

The architecture is based on the UBIWARE platform architecture and extends cloud platform services with the standardized configurable intelligent models.

4.1 Agent-driven Servicing in the Cloud

Interoperability is stated as one of the challenges of the cloud computing paradigm. We believe that

adoption of the existing interoperability tools and solutions will become one of the major cloud computing research directions. Dummy platform API extension will just put the interoperability problem from the cloud provider to the client side. At the same time, the competitiveness of the cloud providers may depend on the simplicity of the integration with the client solutions and systems. Therefore, we foresee that embedded services offered by the cloud providers should be flexible and smart enough to handle client-specific model adjustments and configurations. We expand the understanding of the platform service to the smart proactive agent driven service. Such a service should not only be flexible and configurable in accordance with the customers' needs, it should also be prepared to resolve data- and API-level interoperability issues while being integrated with the client software.

Figure 3 shows the placement of the agent-driven extension in the cloud computing stack. From the user perspective the extension is still a service API but it offers an advanced functionality.

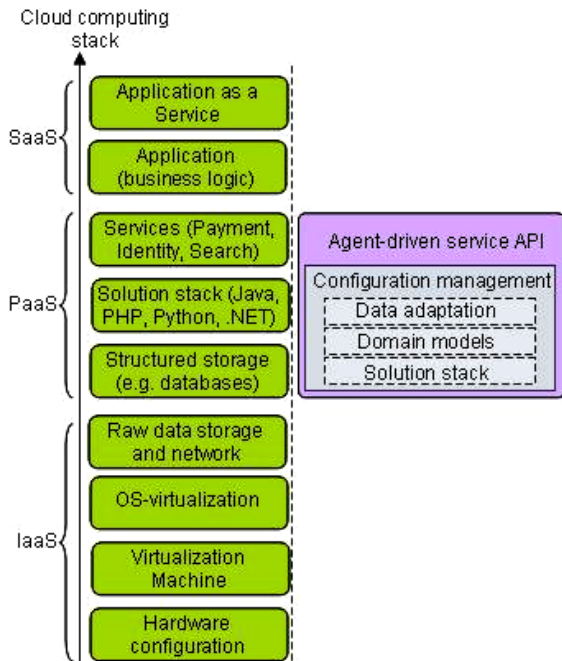


Figure 3: Placement of the agent-driven API.

The API shown above is a standalone middleware platform running either as a cloud facility, or embedded into the virtual machine instances as a platform extension. The detailed API content is shown of Figure 4.

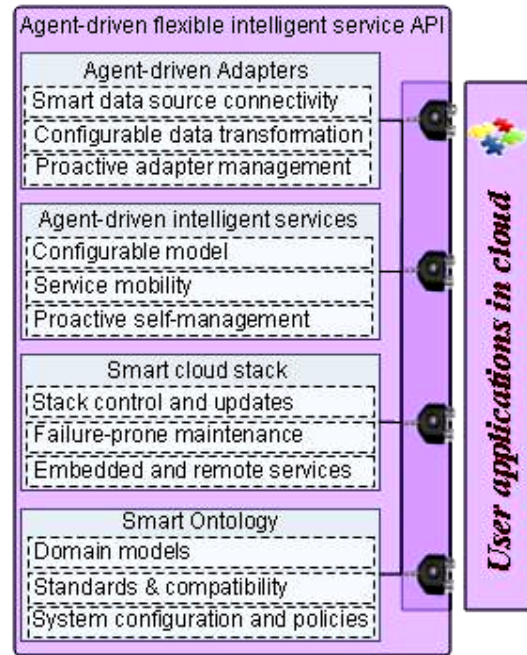


Figure 4: Agent-driven flexible platform service extension API.

The agent-driven adapters are software entities that facilitate data sources management. Adapters provide advanced data source connectivity functions (e.g. simplified database connectivity, file formats parsing, sensor data acquisition, etc.). Next, adapters handle the connectivity problem by providing the components for data transformation with configurable mapping functionality. The adapter becomes a proactive entity, i.e. it observes its state and takes actions based on the state and environment changes. The actions of the adapter may vary from simple fault messaging up to self-reconfiguration when an exceptional or fault situation occurs.

The services' API allows the user to run declarative models as services. The API provides a "model player API" for a particular domain-specific model definition language (the example of the API as well as the language will be discussed in the next Section). The model of the service being played is at the same time controlled by the dedicated agent that takes care of proper model functioning (e.g. load balancing and failures in the operation). Service agent may temporarily relocate the service executable code to another virtual machine instance to improve the performance in critical cases, thus the service becomes remote for its own original virtual machine instance. We also consider service API that has a local representative agent on each virtual machine, but the service execution is handled by the cloud provider (see Figure 5).

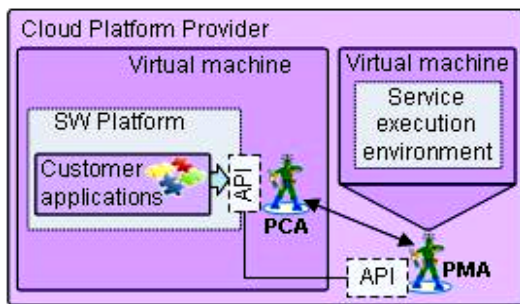


Figure 5: Service execution in cloud infrastructure.

In the Figure above the PCA stands for the Personal Customer Agent and PMA is a Platform Management Agent. The PCA may request the PMA agent to host a service execution (time period is a subject of contracting details) on a separate virtual machine to obtain e.g. higher performance, or for any other reason. At the same time the local API within the user’s virtual machine and/or platform will stay the same. The PCA agent will wrap and forward local API calls to the PMA agent. The difference of the architecture proposed from the standard remote method invocation is a control channel between agents that allows the service management layer to stay separated from the service consumption (service calls).

In the next Section we discuss how the web services from the data mining domain can be integrated into the infrastructure described above. Data mining services can be embedded as platform services into the cloud stack for particular domain-specific cloud configurations at the same time preserving features of configurability, mobility and self-awareness.

4.2 Mastering Data Mining Services into the Cloud

To model the data mining services we have to define a corresponding data mining domain ontology. The ontology will cover data mining methods as well as requirements for method inputs and respective outputs. The inputs and outputs should, in turn, refer to the data types. The data mining domain can not include all possible applications of its methods; therefore we should keep the granularity of the conceptualization and distinguish the data mining models and their application scenarios. For the purpose of this scenario we take two data mining techniques: cluster analysis and k-NN method (classification).

The efforts towards standardization of data mining techniques, methods and formats have been a

matter of discussion for the last ten years. One of the notable efforts is PMML language (PMML, 2009; Guazzelli et al., 2009). The language is a standard for XML documents which express instances of analytical models. In our work we take PMML as a reference model for the Data Mining Ontology and enhance both the model as well as the data with the semantic descriptions required to automate data mining methods application to the domain data. In this work we do not take into account the stage of information collection, preparation, etc. We assume that data is ready for data mining algorithm application. The PMML structure for model definition is composed of a set of elements that describe input data; model and outputs (Figure 6).

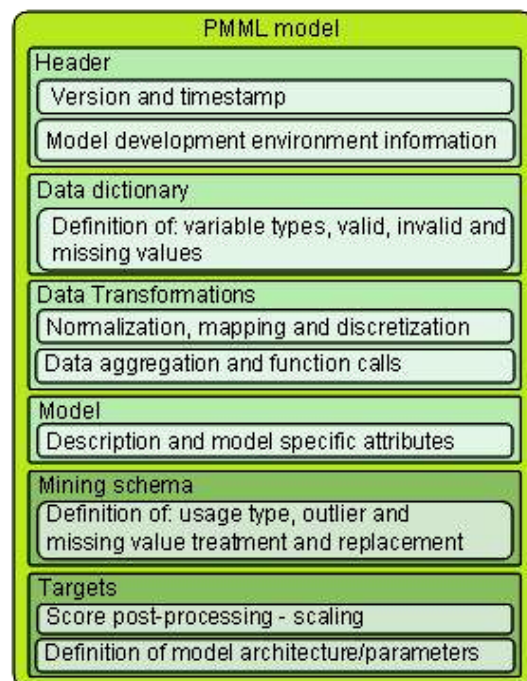


Figure 6: PMML model structure.

The PMML specification ver. 4.0 provides means for exhaustive model description, thus the model can be fully exported or imported without losses. Such model transportability gives huge opportunities for service orientation of the data mining methods. We also expect the PMML models reuse in the cloud computing domain in the nearest future. The specification of a software-independent descriptive data mining standard implies that Infrastructure and Platform layers of Cloud Computing stack are fully transparent for the data mining methods, i.e. the functional characteristics of the method-based services will be same for any stack configuration. The QoS, however, may vary

depending on the performance of the hardware and efficiency of platform software, therefore the additional control channel over the service configuration may be needed.

We have identified three main types of data mining services regardless of their application domain and have introduced a classification of them (Figure 7).

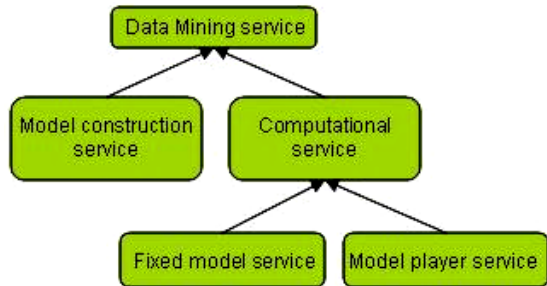


Figure 7: Upper ontology of data mining services.

We consider two major categories of data mining services:

- model construction services
- computational services

The model construction services produce a model (a semantic description) from the set of learning samples. In other words, input of such a service is learning data and conditions (for the neural network depending on its mode it can be a set of vectors plus e.g. initial network parameters). The output of the model construction service is a model with the parameters assigned (e.g. a neural network model, see Figure 8).

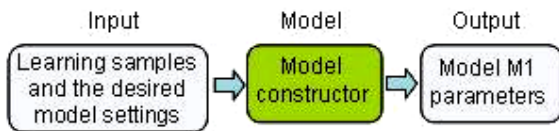


Figure 8: Model construction service.

The group of computational services can also be divided into two major groups:

- services with fixed model
- model player services

The services with fixed model define the format of the input and output as well as provide reference model description and the parameters that determine how the model is configured. For example, Figure 9 shows the definition of the neural network-based alarm classifier service for a paper machine.

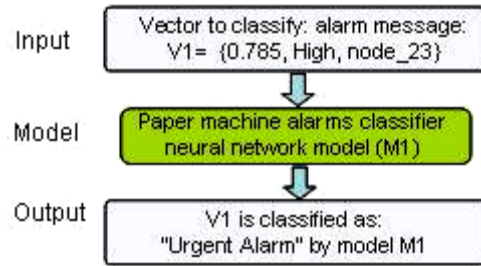


Figure 9: Neural network model in a classification service.

Usage of model player services has two stages: in the first stage the service accepts the service model as an input, and in the second stage, it can serve as a fixed model service (see Figure 10).

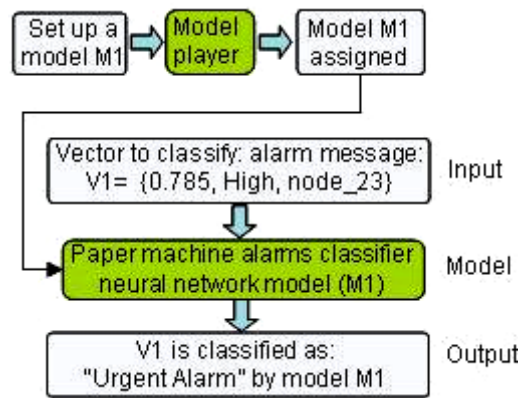


Figure 10: Model player service.

The true power of data mining services can be demonstrated in combination with the distributed querying (i.e. collecting learning or classification data), data mining model construction and further classification. The generic use case of such combination is shown on Figure 11.

The automated data collection process (first step in the use case) uses the Ontonuts technology and approach researched in the (Nikitin et al., 2009). The approach allows dynamic distributed query planning and execution, which we apply in this work to collect learning set data. The sources of the data may vary from databases, to csv-files and reside physically on different platforms and sites. The data collection and, hence, the querying implemented as a sequence of semantic data service calls orchestrated by a workflow management agent. Service orientation of data sources makes distributed querying a homogeneous part of other service-based workflow scenarios. The data collected (usually in form of a table of query results), may undergo preparatory steps to become a learning dataset. In this work we omit the procedure of normalization, or

other data transformations, however, they will be necessary and obligatory. We assume that all operations with the data are also wrapped as semantic services.

As soon as the learning set is ready, a desired model constructor should be chosen (step 2). The model constructor may require specific data preparation, therefore it is good to combine data preparation step with the model constructor service.

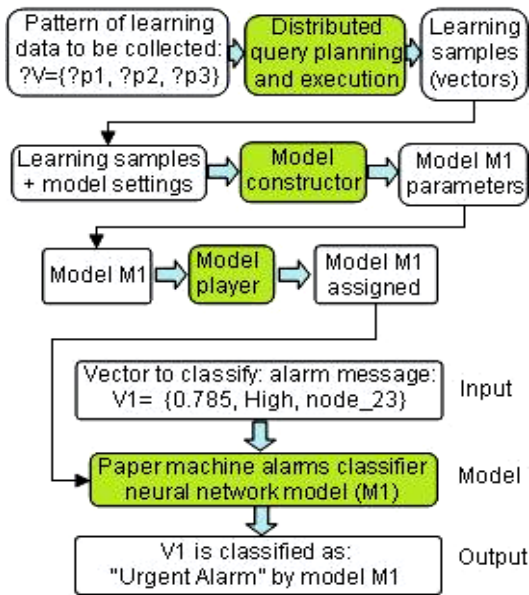


Figure 11: A use case scenario.

As an input, model constructor may require additional input parameters for model building. Those may be set as default, or, if other parameters were prepared, they should be supplied in the proper form. When a model is ready, we feed it to the model player service which is a platform service in terms of cloud computing, because it provides an infrastructure and software platform for service execution. As soon as our newly built model is deployed as a service, we can start classifying the data vectors, e.g. alarms coming from the paper machine.

The scenario described above may be dynamically reconfigured by the infrastructure agents. Some steps of the case (e.g. learning) may be temporarily moved to the separate execution environment (separate platform or virtual machine) to perform computationally expensive tasks.

The overall infrastructure of services should be highly proactive and responsive to the customer calls. Agents may monitor the execution and be ready to reconfigure their services in accordance with the current customer context.

5 RELATED WORK

The cloud platform solutions for business intelligence are gaining popularity. For example, Salesforce.com claims about 2 million success stories of its customers. The platform provides wide range of products (from infrastructure as a service, up to tailored web-based solutions for health care, retail and sales). The business intelligence tools are offered too. Nevertheless, the user has to adjust or prepare her/his software and data for the tools provided by the cloud. The advantage of the approach we offer is to empower any cloud platform with the intelligent adaptation mechanisms that would allow seamless data connectivity and integration. The architecture we offer is an extension to the cloud platform, not the platform itself. The data mining services with highly configurable parameters driven by the intelligent agents would simplify business intelligence integration, hence making adoption of cloud architecture easier for clients.

6 CONCLUSIONS

The research presented above describes specific application domain of intelligent services. We foresee that model player services will be a successful business case for the emerging paradigm of cloud computing. Pay-per-use principles combined with high computational capacities of cloud and standardized DM-models will be definitely an alternative to expensive business intelligence and statistics toolkits.

Another niche of data mining services in cloud computing can be model construction services. Such systems will drive innovations in data mining methods as well as applied data mining in certain domains. Such services will compete by introducing know-how and innovative tools and algorithms that bring add-values in e.g. predictive diagnostics or computational error estimation. This direction will lead to so-called “web intelligence” (Cercone et al.).

The role of UBIWARE platform in cloud computing emerges as a cross-cutting management and configuration glue for interoperability of future intelligent cloud services and client applications.

The main burden of UBIWARE will be management of consistency across different domain conceptualizations (Ontologies) and cross-domain middleware components. Fine-grained ontology modeling is still a challenge for research community and we predict that in the nearest future the domain

modeling will be task-driven, i.e. the domain model engineers may incorporate some standardized and accepted conceptualizations, whereas the whole ontology for solution will be tailor made. Tailored ontologies will require subsequent mapping mechanisms and additional efforts.

The advanced data integration mechanisms embedded into the cloud platform as services is also an interesting concept that may become an add value for competing cloud platforms. The easiness of integration into the cloud infrastructure should not be underestimated especially by enterprise-sized companies, where business processes and component interdependencies have reached an unprecedented level of complexity. We believe that autonomy and self-awareness of building blocks will be a key to the future design of information systems and cloud platforms.

ACKNOWLEDGEMENTS

This research has been supported by the UBIWARE project, funded by TEKES, and the industrial consortium of Metso Automation, Fingrid and Inno-W. The preparation of this paper was partially funded by the COMAS graduate school.

REFERENCES

- Bellifemine, F. L., Caire, G., and Greenwood, D., 2007. *Developing Multi-Agent Systems with JADE*. Wiley.
- Cercone, N.; Lijun Hou; Keselj, V.; Aijun An; Naruedomkul, K.; Xiaohua Hu, 2002. "From computational intelligence to Web intelligence," *Computer*, vol.35, no.11, pp. 72-76, Nov 2002.
- Foster, I.; Yong Zhao; Raicu, I.; Lu, S., 2008 "Cloud Computing and Grid Computing 360-Degree Compared," *Grid Computing Environments Workshop GCE '08*, vol., no., pp.1-10, 12-16 Nov. 2008.
- Guazzelli, A., Zeller, M., Lin, W. and Williams, G., 2009. PMML: An Open Standard for Sharing Models. *The R Journal*, Volume 1/1, May 2009.
- Hayes, B. 2008. Cloud computing. *Commun. ACM* 51, 7 (Jul. 2008), 9-11. DOI= <http://doi.acm.org/10.1145/1364782.1364786>
- Kaykova, O., Khriyenko, O., Kovtun, D., Naumenko, A., Terziyan, V., and Zharko, A., 2005. General Adaption Framework: Enabling Interoperability for Industrial Web Resources, In: *International Journal on Semantic Web and Information Systems*, Idea Group, Vol. 1, No. 3, pp.31-63.
- Kephart, J. O. and Chess, D. M., 2003. The vision of autonomic computing., *IEEE Computer*, 36(1):41–50.
- Nikitin S., Katasonov A., Terziyan V., 2009. Ontonuts: Reusable Semantic Components for Multi-Agent Systems, In: *Proceedings of the Fifth International Conference on Autonomic and Autonomous Systems (ICAS 2009)*, April 21-25, 2009, Valencia, Spain, IEEE CS Press, pp 200-207.
- PMML, 2009. Data Mining Group. PMML version 4.0. WWW, URL <http://www.dmg.org/pmml-v4-0.html>
- Rimal, B-P; Choi, E; Lumb, I, 2009, "A Taxonomy and Survey of Cloud Computing Systems," *INC, IMS and IDC, 2009. NCM '09. Fifth International Joint Conference on*, pp.44-51, 25-27 Aug. 2009.
- Terziyan, V., 2003. Semantic Web Services for Smart Devices in a "Global Understanding Environment", In: R. Meersman and Z. Tari (eds.), *On the Move to Meaningful Internet Systems 2003: OTM 2003 Workshops*, Lecture Notes in Computer Science, Vol. 2889, Springer-Verlag, pp.279-291.
- Terziyan, V., 2005. Semantic Web Services for Smart Devices Based on Mobile Agents, In: *International Journal of Intelligent Information Technologies*, Vol. 1, No. 2, Idea Group, pp. 43-55.

EVALUATING MAXIMUM TRANSMISSION UNRELIABILITY IN PERSISTENT CSMA PROTOCOL

Dariusz Kościelnik and Marek Miśkowicz

Department of Electronics, AGH University of Science and Technology, Al. Mickiewicza 30, Cracow, Poland
koscieln@agh.edu.pl, miskow@agh.edu.pl

Keywords: Carrier Sense Multiple Access, Performance Analysis, Analytical Modelling.

Abstract: The paper addresses the problem of evaluating the unreliability of transmission, undertaken by a given station, according to the persistent CSMA scheme. The unreliability of transmission is considered on the media access control level so it is defined by the probability that a given node participates in a collision. The presented results show that the maximum transmission unreliability is upper bounded by the persistence level (p), which is the main parameter of the protocol. The presented analysis is compared to the corresponding results for the non-persistent CSMA. As shown, both results are complementary because the maximum transmission unreliability in the non-persistent CSMA scheme is also bounded by the probability of choosing a single slot in the contention window.

1 INTRODUCTION

Although the carrier sense multiple access (CSMA) protocols have been introduced in the early 70s, due to their inherent flexibility and simplicity, they are in more advanced versions still widely used in contemporary networking, especially for wireless communication (e.g., Tay, Jamieson, Balakrishnan, 2004). In particular, the predictive CSMA protocol is employed in Local Operating Networks (LonWorks) commercial platform for sensor and control networking (Miśkowicz, Golański, 2006). On the other hand, the non-persistent CSMA scheme with a geometric distribution has been recently proposed for sensor networking (Tay, Jamieson, Balakrishnan, 2004; Miśkowicz, 2009a; Egea-López *et al.*, 2007).

The performance of the CSMA protocols have been investigated intensively for decades. The main criterion of performance analyses is evaluating the throughput-delay characteristics (Kleinrock, Tobagi, 1975; Lam, 1980), or the protocol energy efficiency in the context of wireless networking (Bruno, Conti, Gregori, E., 2002; Cali, Conti, Gregori, 2000).

Most of the performance analyses are based on classical approaches where the network load is assumed to contain an infinity number of stations that collectively produce (including both new and retransmitted packets) the traffic with Poisson arrivals (Kleinrock, Tobagi, 1975; Lam, 1980).

The Poisson model is an approximation of a large but finite population in which every station generates messages infrequently and each message can be successfully transmitted a long time before the station generates a next message.

The Poisson traffic model, based on the fundamental assumption of stochastically distributed independent stimuli, has been applied widely for performance evaluation of computer networks.

Since the assumption of lightly loaded network is invalid in many load scenaria in wireless local area networks (Bruno, Conti, Gregori, 2002; Cali, Conti, Gregori, 2000; Bianchi, 1998), and especially in networked sensor and control systems (Miśkowicz, 2009b), the other class of performance analyses deal with the load scenario of finite number of active stations in which every station may produce a significant portion of network traffic. In order to model the network operation under heavy load, it is assumed that all the stations are in the *asymptotic (saturation) conditions* where they have always a packet ready for transmission.

The present study deals with the performance analysis of the *persistent CSMA protocol* that belongs to one of generic CSMA schemes introduced in (Kleinrock, Tobagi, 1975). The paper contribution is the analytical proof that the unreliability of transmission undertaken by a given station, according to the persistent CSMA scheme, is

upper bounded by the persistence level (p), which is the main parameter of the protocol.

The unreliability of transmission is considered on the media access control level so it is defined by the probability that a given node participates in a collision. The presented analysis is compared to the corresponding results for the non-persistent CSMA. As interesting, both results are complementary because the maximum transmission unreliability in the non-persistent CSMA scheme is also bounded by the probability of choosing a single slot in the contention window (Miśkiewicz, Kościelnik, 2010).

The authors believe that the presented results contributes to better understanding of the persistent CSMA operation. To the best authors' knowledge, these results have not been yet published.

The performance analysis stated in the present paper belongs to the studies of persistent CSMA scheme for the network staying in the asymptotic conditions because the evaluation of the maximum transmission unreliability needs to feed a channel with heavy load.

2 ANALYTICAL MODEL OF PERSISTENT CSMA

2.1 Persistent CSMA Specification

The *persistent CSMA* scheme belongs to the slotted-CSMA protocol where the channel idle time is divided into fixed length intervals. All the stations in the network are synchronized and forced to start a transmission only at the beginning of a slot.

In the network that operates according to the persistent CSMA, when a station has a new message to transmit, it senses the channel. If the channel is detected to be idle, then it transmits a message with the probability p , while with probability $1-p$, it delays the message transmission to the next time slot. The slot duration is determined by the network propagation delay.

By a comparison, in the non-persistent CSMA, when the station senses the channel to be idle, it draws a number of a slot from a set of slots included in the *contention window*. The probability distribution of a random slot selection is uniform.

In the persistent CSMA protocol, the number of empty slots preceding a (successful or unsuccessful) transmission of a data packet is theoretically unbounded because the probability of starting transmission is defined by the geometric distribution where a success occurs with the probability p , and a

failure with the probability $(1-p)$. The mean number of trials undertaken by a given station equals $1/p$. On the other hand, in the non-persistent CSMA protocol, the maximum number of empty slots before (successful or unsuccessful) transmission of a data packet equals $(W-1)$, and the mean number $(W-1)/2$ where W is a number of slots in the contention window.

2.2 Collision Probability in Single Transmission Attempt

The probability $p_{coll(1)}^{(k)}$ that a certain station is involved in collision in the k th transmission attempt is defined by the product of the following probabilities:

$$p_{coll(1)}^{(k)} = p_{coll(11)}^{(k)} p_{coll(12)}^{(k)} \quad (1)$$

where $p_{coll(11)}^{(k)}$ is the probability that all the contending stations had not started the transmission in the previous $1, \dots, k-1$ transmission attempts, and $p_{coll(12)}^{(k)}$ is the probability that at least one from the $s=1, 2, 3, \dots$ contending stations apart from a selected station starts the transmission in the k th transmission attempt. The former probability $p_{coll(11)}^{(k)}$ is given by the formula:

$$p_{coll(11)}^{(k)} = \prod_{x=1}^{k-1} (1-p)^{s+1} = (1-p)^{(s+1)(k-1)} \quad (2)$$

The latter probability $p_{coll(12)}^{(k)}$ is defined as follows:

$$p_{coll(12)}^{(k)} = p \sum_{x=1}^s C_s^x p^x (1-p)^{s-x} \quad (3)$$

where $C_s^x = s! / [(s-x)! x!]$ is the binomial coefficient, and $s, s > 1$ is an integer.

Thus, the $p_{coll(1)}^{(k)}$ is given as:

$$p_{coll(1)}^{(k)} = p(1-p)^{(s+1)(k-1)+s} \left(\sum_{x=0}^s \left[C_s^x \left(\frac{p}{1-p} \right)^x \right] - 1 \right) \quad (4)$$

The formula (4) may be transformed as follows:

$$p_{coll(1)}^{(k)} = p(1-p)^{(s+1)(k-1)+s} \left(\left(\frac{1}{1-p} \right)^s - 1 \right) \quad (5)$$

because according to the Newton's generalized binomial theorem:

$$\sum_{x=0}^s C_x^s \left(\frac{p}{1-p} \right) = \left(\frac{p}{1-p} + 1 \right)^s \quad (6)$$

As follows from (3), the probability $p_{coll(12)}^{(k)}$ does not depend on k but only on the number of contenders s . On the other hand, the probabilities $p_{coll(11)}^{(k)}$ (see (2)), and consequently $p_{coll(1)}^{(k)}$ also (see (5)), is a decreasing function of the number transmission attempt k .

In Fig. 1, the plot of the probability $p_{coll(1)}^{(k)}$ versus the number of transmission attempt k for selected numbers of the contending stations $s = \{1, 3, 5\}$ for $p = 1/16$ according to (5) is shown.

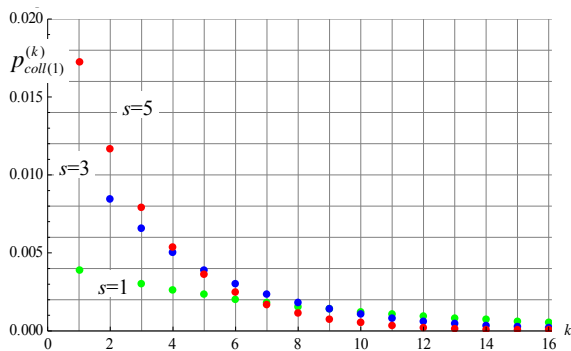


Figure 1: The probability $p_{coll(1)}^{(k)}$ that a certain station is involved in collision in the k th transmission attempt versus the number k for various population sizes of the contending stations s according to (5) for $p = 1/16$ and $s = \{1, 3, 5\}$.

In Fig. 2(a,b), the plots of the probability $p_{coll(1)}^{(k)}$ versus the number of the contending stations s in selected transmission attempts $k; k = \{1, 2, 5, 9\}$ according to (5) are presented.

As seen in Fig. 2a, the probability $p_{coll(1)}^{(k)}$ that a given station participates in collision for $k \geq 2$ is a unimodal function of s that is maximized for a certain number of contending stations $s_0^{(k)}$. The $s_0^{(k)}$ is a decreasing function of $k, k \geq 2$. Instead, for $k=1$, the $p_{coll(1)}^{(k)}$ increases strictly with s and approaches a horizontal asymptote $p_{coll(1)}^{(k)} = p$ as may be analytically derived (Fig. 2b):

$$\lim_{s \rightarrow \infty} p_{coll(1)}^{(1)} = \lim_{s \rightarrow \infty} (1-p)^s p \cdot \left(\left(\frac{p}{1-p} + 1 \right)^s - 1 \right) = p \quad (7)$$

In particular, $p_{coll(1)}^{(k)}$ reaches 0.0625 for $p = 1/16 = 0.0625$ for high s as seen in Fig. 2b.

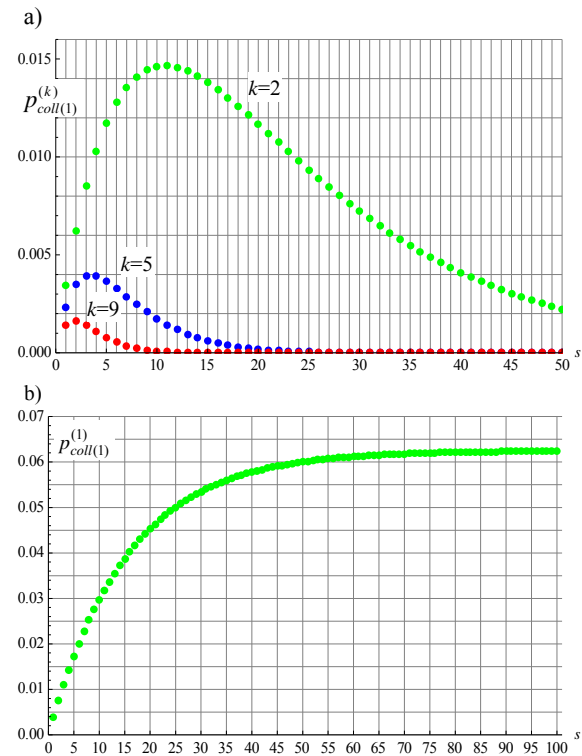


Figure 2: The probability $p_{coll(1)}^{(k)}$ versus the population of the contending stations s for $p = 1/16 = 0.0625$ and selected transmission attempts $k = \{2, 5, 9\}$ (a), and $k = \{1\}$ (b).

3 MAXIMUM TRANSMISSION UNRELIABILITY IN PERSISTENT CSMA

As stated, the probability $p_{coll(1)}^{(k)}$ that a certain station is involved in collision in the k th transmission attempt is defined by the formula (5). The total probability $p_{coll(1)}^{(1 \div k)}$ that a certain station participates in collision at most at the k th transmission attempt is defined as a sum:

$$p_{coll(1)}^{(1 \div k)} = \sum_{x=1}^k p_{coll(1)}^{(x)} \quad (8)$$

where $p_{coll(1)}^{(k)}$ is defined by (5).

By setting (5) to (8):

$$p_{coll(1)}^{(1+k)} = \sum_{x=1}^k p(1-p)^{(s+1)(x-1)+s} \left(\left(\frac{1}{1-p} \right)^s - 1 \right) \quad (9)$$

The probability $p_{coll(1)}^{(1+k)}$ in the formula (9) is defined as a geometric series with the first term equal to $p[1-(1-p)^s]$, and the ratio equal to $(1-p)^{s+1}$ so it might be computed as:

$$p_{coll(1)}^{(1+k)} = \frac{p \cdot (1-(1-p)^s) \left(1-(1-p)^{(s+1)k} \right)}{1-(1-p)^{s+1}} \quad (10)$$

The plot of the probability $p_{coll(1)}^{(1+k)}$ versus k according to (10) is shown in Fig. 3 for $p=1/16$ and $s=\{1,2,5,10\}$.

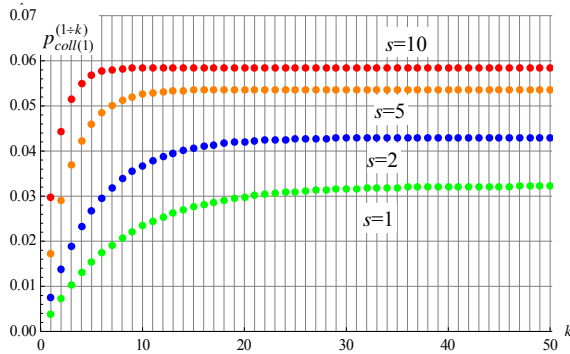


Figure 3: The probability $p_{coll(1)}^{(1+k)}$ according to (10) for $p=1/16$ and $s=\{1,2,5,10\}$.

As seen in Fig. 3, each curve approaches a horizontal asymptote with growing number of transmission attempt k . These asymptotes corresponding to the limits:

$$p_{coll(1)} = \lim_{k \rightarrow \infty} p_{coll(1)}^{(1+k)} \quad (11)$$

for various s and denoted by $p_{coll(1)}$ defines the probability of collision in any attempt in a given transmission cycle.

By setting (10) to (11):

$$p_{coll(1)} = \frac{p \cdot (1-(1-p)^s)}{1-(1-p)^{s+1}} \quad (12)$$

As follows from (12), the $p_{coll(1)}$ depends both on the p value and the number of contending stations s . The plots of $p_{coll(1)}$ versus the persistence level p for various numbers of contending stations s is presented in Fig. 4.

As seen in Fig. 4, the probability $p_{coll(1)}$ grows with increasing p but it is at the same time smaller than p for any number of contending stations s . This conclusion may be also derived analytically on the basis of (12) as follows:

$$p_{coll(1)} = \frac{p \cdot (1-(1-p)^s)}{1-(1-p)^{s+1}} < p \quad (13)$$

because $\frac{1-(1-p)^s}{1-(1-p)^{s+1}} < 1$ for $s \geq 1$, and furthermore:

$$\lim_{s \rightarrow \infty} p_{coll(1)} = \lim_{s \rightarrow \infty} \frac{p \cdot (1-(1-p)^s)}{1-(1-p)^{s+1}} = p \quad (14)$$

Thus, the probability $p_{coll(1)}$ of collision in any attempt in a given transmission cycle is upper bounded by the persistence level p regardless of the number of contending stations s .

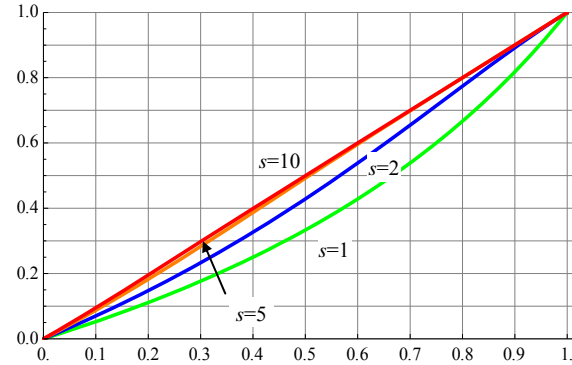


Figure 4: Plots of $p_{coll(1)}$ vs. the persistence level p for $s=\{1,2,5,10\}$.

In Fig. 5, the plots of the $p_{coll(1)}$ versus s for $p=\{1/4, 1/16, 1/64\}$ are presented. Finally, in Fig. 6, the $p_{coll(1)}$ versus s and p in 3-D plot are illustrated.

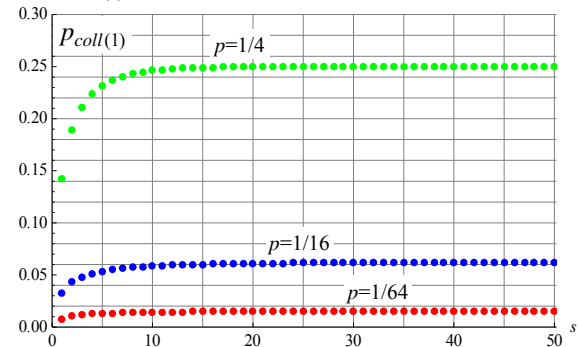


Figure 5: Plots of the $p_{coll(1)}$ versus s for $p=\{1/4, 1/16, 1/64\}$.

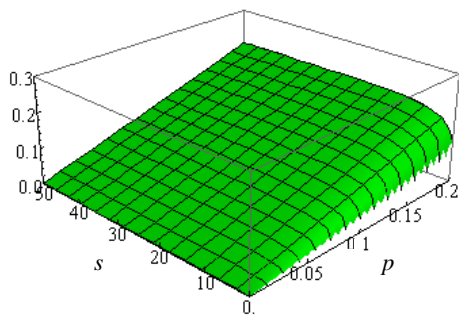


Figure 6: The 3-D plot of $p_{coll(1)}$ versus s and p .

By a comparison, the probability that a given station participates in collision for non-persistent CSMA with a number of W contending slots equals $1/W$ and is independent of the number of contenders (Koscielnik, Miskowicz, 2010).

4 CONCLUSIONS

We compare the maximum transmission unreliability in the non-persistent CSMA and persistent CSMA for the same average number of contention slots in both schemes. In the persistent CSMA, the latter equals simply $1/p$. In the non-persistent CSMA, the contention window is constant in each transmission cycle and equals W slots.

As proved in (Koscielnik, Miskowicz, 2010), the maximum probability of participating in a collision in the non-persistent CSMA scheme is upper bounded by $1/W$, that is, by the probability of a selection of a single slot in the transmission attempt.

On the other hand, as follows from the present paper, maximum probability of participating in a collision by a given station, according to the persistent CSMA scheme, is upper bounded by the persistence level (p), which a main parameter of the protocol. Thus, the complementary results defined by (14) are valid for the persistent CSMA scheme.

REFERENCES

Tay, Y. C., Jamieson, K., Balakrishnan, H., 2004. "Collision-minimizing CSMA and its applications to wireless sensor networks", *IEEE Journal on Selected Areas in Communication*, vol.22, pp.1048-1057.

Miśkiewicz, M., Golański R., 2006. LON technology in wireless sensor networking applications, *Sensors*, vol. 6, no. 1, pp. 30-48.

Miśkiewicz, M., 2009. Average channel utilization of CSMA with geometric distribution under varying

workload, *IEEE Transactions on Industrial Informatics*, vol. 6, no. 2, pp.123-131, 2009.

Egea-López, E., Vales-Alonso, J. Martínez-Sala, A. S., Bueno-Delgado, M. V., M. V. García-Haro, J., 2007. Performance evaluation of non-persistent. CSMA as anti-collision protocol for active RFID tags", *Lecture Notes in Computer Science LNCS 4517*, pp. 279-289.

Kleinrock, L., Tobagi, F.A, 1975. Carrier sense multiple-access modes and their throughput-delay characteristics, *IEEE Transactions on Communication*, vol. COM-23, no. 12, pp. 1400-1416.

Lam, S.S., 1980. A carrier sense multiple access protocol for local networks, *Computer Networks*, vol. 4, no. 1, pp. 21-32.

Bruno, R., Conti, M., Gregori, E., 2002. Optimization of efficiency and energy consumption in p-persistent CSMA-based wireless LANs, *IEEE Transactions on Mobile Computing*, vol. 1, no. 1, pp. 10-31.

Cali, F., Cont, M., Gregori, E., 2000. Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit, *IEEE/ACM Transactions on Networking*, vol. 8, no. 6, pp. 785-799.

Miśkiewicz, M., 2009. Access delay in LonTalk MAC protocol, *Computer Standards & Interfaces*, vol. 31, no. 3, pp. 548-556.

Bianchi, G., 1998. IEEE 802.11—Saturation throughput analysis, *IEEE Communications Letters*, vol. 2, no. 12, pp. 318-320.

Miśkiewicz, M., Kościelnik, D., 2010. On the upper bound of transmission unreliability in memoryless backoff contention, *Proceedings of IEEE International Workshop on Factory Communication Systems WFCS 2010*, Nancy, France, May 2010.

STRUCTURE-PRESERVING ALGORITHMS FOR DISCRETE-TIME ALGEBRAIC MATRIX RICCATI EQUATIONS

Vasile Sima

National Institute for Research & Development in Informatics, 8-10 Bd. Mareşal Averescu, Bucharest, Romania
vsima@ici.ro

Keywords: Linear-quadratic Optimization, Optimization Algorithms, Structure-preserving Algorithms.

Abstract: Structure-preserving algorithms for solving discrete-time algebraic matrix Riccati equations are presented. The proposed techniques extract the stable deflating subspaces for extended, inverse-free symplectic matrix pencils. The algorithms are based on skew-Hamiltonian/Hamiltonian pencils derived by an extended Cayley transformation, which only involves matrix additions and subtractions. The structure-preserving approach has the potential to avoid the numerical difficulties which are encountered for a traditional, non-structured solution, returned by the currently available software tools.

1 INTRODUCTION

Consider the *continuous-time algebraic Riccati equation (CARE)*,

$$0 = Q + A^H X E + E^H X A - E^H X W X E$$

and the *discrete-time algebraic Riccati equation (DARE)*,

$$E^H X E = Q + A^H X A - A^H X B (R + B^H X B)^{-1} B^H X A,$$

where $A, E, W, Q \in \mathbb{C}^{n \times n}$, $B \in \mathbb{C}^{n \times m}$, $R \in \mathbb{C}^{m \times m}$, $W = W^H$, $Q = Q^H$, $R = R^H$, E is nonsingular, and $W := B R^{-1} B^H$. More general equations are obtained by replacing $E^H X B$ and $A^H X B$ above by $L + E^H X B$ and $L + A^H X B$, respectively, where $L \in \mathbb{C}^{n \times m}$. The real case is obtained by replacing \mathbb{C} by \mathbb{R} , and the conjugate-transpose operator H by the transpose operator T .

In applications, usually the *stabilizing solution* X_* is required, hence, e.g., for DARE, $\lambda E - (A - B(R + B^H X_* B)^{-1} B^H X_* A)$ is a (Schur) stable matrix pencil, i.e., $\Lambda(A - B(R + B^H X_* B)^{-1} B^H X_* A, E) \subset \mathbb{C}^- := \{z \in \mathbb{C} : |z| < 1\}$, where $\Lambda(M)$ denotes the spectrum of a matrix or pencil M .

CAREs and DAREs arise in many applications, such as, stabilization and linear-quadratic regulator problems, Kalman filtering, linear-quadratic Gaussian (H_2 -) optimal control problems, computation of (sub)optimal H_∞ controllers, model reduction techniques based on stochastic, positive or bounded real LQG balancing, factorization procedures for transfer functions (here, usually $E \neq I_n$).

There are several basic approaches for solving algebraic Riccati equations (AREs):

1. Treat an ARE as a nonlinear system of equations using *Newton's method (with line search)*.
2. Use the connection to *Hamiltonian eigenproblem*.

The second approach for CARE, with $E = I_n$, is based on the identity

$$\begin{bmatrix} A & -W \\ -Q & -A^H \end{bmatrix} \begin{bmatrix} I_n \\ X \end{bmatrix} = \begin{bmatrix} I_n \\ X \end{bmatrix} (A - WX),$$

hence, if X is stabilizing, then $\Lambda(A - WX) = \Lambda(H) \cap \mathbb{C}^-$, where H is the first matrix in the above formula, and $\mathbb{C}^- := \{z \in \mathbb{C} : \Re(z) < 0\}$. Consequently, the columns of $[I_n \ X^T]^T$ span the *stable* invariant subspace of the Hamiltonian matrix H . Therefore, it is possible to compute the stable H -invariant subspace via eigendecomposition or block-Schur factorization,

$$U^{-1} H U = \begin{bmatrix} H_{11} & H_{12} \\ 0 & H_{22} \end{bmatrix}, \quad U = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix},$$

and the solution is given by $X = U_{21} U_{11}^{-1}$.

If R is ill-conditioned, it is advisable to use extended matrix pencils, for better accuracy (Bender and Laub, 1987a; Bender and Laub, 1987b; Lancaster and Rodman, 1995; Mehrmann, 1991; Van Dooren, 1981):

– *extended pencil for CARE*:

$$N - \lambda M = \begin{bmatrix} A & 0 & B \\ Q & A^H & L \\ L^H & B^H & R \end{bmatrix} - \lambda \begin{bmatrix} E & 0 & 0 \\ 0 & -E^H & 0 \\ 0 & 0 & 0 \end{bmatrix};$$

– extended pencil for DARE:

$$N - \lambda M = \begin{bmatrix} A & 0 & B \\ Q & -E^H & L \\ L^H & 0 & R \end{bmatrix} - \lambda \begin{bmatrix} E & 0 & 0 \\ 0 & -A^H & 0 \\ 0 & -B^H & 0 \end{bmatrix}.$$

If $[U_1^T \ U_2^T \ U_3^T]^T$ spans the stable deflating subspace of $N - \lambda M$, then $X_* = U_2(EU_1)^{-1}$. The feedback gain matrix for the linear-quadratic optimal regulator can be computed directly via $G = U_3U_1^{-1}$.

If R is nonsingular, $E = I_n$, and $L = 0$, the above pencils can be reduced to $2n \times 2n$ Hamiltonian and symplectic pencils, respectively, by removing the sub-pencils with infinite eigenvalues (Paige and Van Loan, 1981; Pappas et al., 1980; Mehrmann, 1991). A pencil $N - \lambda M$ is *Hamiltonian* if $NJM^H = -MJN^H$, and it is *symplectic* if $NJN^H = MJM^H$, where

$$J := \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix}.$$

The general pencils inherit most of the spectral properties of the corresponding reduced Hamiltonian or symplectic pencils.

The pencils above have much structure, which should be exploited in order to improve the numerical properties of the Riccati solvers. The approach we follow is to transform the discrete-time problem to an equivalent continuous-time problem, and use the newly developed skew-Hamiltonian/Hamiltonian eigensolvers for the latter problem, suitably extended.

2 EQUIVALENCE OF PENCILS IN CONTINUOUS-TIME AND DISCRETE-TIME PROBLEMS

A block column permutation (and sign change) gives, equivalently:

– extended pencil for CARE:

$$\lambda \begin{bmatrix} 0 & E & 0 \\ -E^H & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & A & B \\ A^H & Q & L \\ B^H & L^H & R \end{bmatrix};$$

– extended pencil for DARE:

$$\lambda \begin{bmatrix} 0 & E & 0 \\ -A^H & 0 & 0 \\ -B^H & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & A & B \\ -E^H & Q & L \\ 0 & L^H & R \end{bmatrix}.$$

These pencils are special cases of the following *block structured C-type* and *D-type pencils* (Xu, 2006):

$$\lambda \mathcal{E}_C - \mathcal{A}_C = \lambda \begin{bmatrix} 0 & \tilde{F} \\ -\tilde{F}^H & 0 \end{bmatrix} - \begin{bmatrix} 0 & \tilde{G} \\ \tilde{G}^H & \tilde{D} \end{bmatrix}, \quad (1)$$

and

$$\lambda \mathcal{E}_D - \mathcal{A}_D = \lambda \begin{bmatrix} 0 & F \\ -G^H & 0 \end{bmatrix} - \begin{bmatrix} 0 & G \\ -F^H & D \end{bmatrix}, \quad (2)$$

respectively, where $F, G, \tilde{F}, \tilde{G} \in \mathbb{C}^{n,q}$, $q = n + m$, and $D, \tilde{D} \in \mathbb{C}^{q,q}$ are *Hermitian*, i.e., $D = D^H$, $\tilde{D} = \tilde{D}^H$.

These pencils have important *spectral properties*: C-type: symmetry about $\Re(z) = 0$, i.e., pairs $(\lambda, -\tilde{\lambda})$; D-type: symmetry about $|z| = 1$, i.e., pairs $(\lambda, \tilde{\lambda}^{-1})$.

An equivalence transformation between the C-type and D-type pencils can be established starting from the *Cayley transformation*, $\mathbf{c} : \mathbb{C} \cup \{\infty\} \rightarrow \mathbb{C} \cup \{\infty\}$, defined by

$$\mu = \mathbf{c}(\lambda) = (\lambda - 1)(\lambda + 1)^{-1}; \quad \mathbf{c}(-1) = \infty, \quad \mathbf{c}(\infty) = 1.$$

Specifically, the *generalized Cayley transformation* for matrix pairs is given by

$$(\mathcal{F}, \mathcal{B}) = \mathbf{c}(\mathcal{E}, \mathcal{A}) = (\mathcal{A} + \mathcal{E}, \mathcal{A} - \mathcal{E}). \quad (3)$$

Let

$$(\tilde{\mathcal{E}}, \tilde{\mathcal{A}}) := \mathbf{c}(\mathcal{E}_D, \mathcal{A}_D).$$

Then, an eigenvalue pair $(\lambda, \tilde{\lambda}^{-1})$ of $\lambda \mathcal{E}_D - \mathcal{A}_D$ is transformed to an eigenvalue pair $(\mu, -\bar{\mu})$ of $\lambda \tilde{\mathcal{E}} - \tilde{\mathcal{A}}$, with $\mu = \mathbf{c}(\lambda)$, $-\bar{\mu} = \mathbf{c}(\tilde{\lambda}^{-1})$.

Unfortunately, $\lambda \tilde{\mathcal{E}} - \tilde{\mathcal{A}}$ has not the same block structure as $\lambda \mathcal{E}_C - \mathcal{A}_C$, and it cannot be put into the continuous-time setting. This inconvenience can be removed using the Cayley transformation followed by a *drop/add transformation* (Xu, 2006):

$$(\mathcal{E}_C, \mathcal{A}_C) = \mathbf{t}(\mathcal{E}_D, \mathcal{A}_D),$$

where $\mathbf{t}(\cdot) = \mathbf{d}(\mathbf{c}(\cdot))$, and \mathbf{d} corresponds to dropping/adding D in the \mathcal{E} part.

The Cayley plus drop/add transformation is suggestively represented by the following *t transformation diagram*:

$$\begin{aligned} \lambda \mathcal{E}_D - \mathcal{A}_D &= \lambda \begin{bmatrix} 0 & F \\ -G^H & 0 \end{bmatrix} - \begin{bmatrix} 0 & G \\ -F^H & D \end{bmatrix} \\ &\quad \mathbf{c} \downarrow \uparrow \mathbf{c}^{-1} \\ \lambda \tilde{\mathcal{E}} - \tilde{\mathcal{A}} &= \lambda \begin{bmatrix} 0 & \tilde{F} \\ -\tilde{F}^H & D \end{bmatrix} - \begin{bmatrix} 0 & \tilde{G} \\ \tilde{G}^H & D \end{bmatrix} \\ &\quad \text{drop } D \text{ from } \tilde{\mathcal{E}} \downarrow \uparrow \text{ add } D \text{ to } \tilde{\mathcal{E}} \\ \lambda \mathcal{E}_C - \mathcal{A}_C &= \lambda \begin{bmatrix} 0 & \tilde{F} \\ -\tilde{F}^H & 0 \end{bmatrix} - \begin{bmatrix} 0 & \tilde{G} \\ \tilde{G}^H & \tilde{D} \end{bmatrix} \end{aligned}$$

where $\tilde{F} := G + F$, $\tilde{G} := G - F$, $\tilde{D} = D$. It is worth mentioning that the \mathbf{t} transformation involves matrix additions and subtractions only.

Only regular pencils are considered in the sequel. A pencil $\lambda \mathcal{E} - \mathcal{A}$ is *regular* if \mathcal{E} and \mathcal{A} are square and $\det(\gamma \mathcal{E} - \mathcal{A}) \neq 0$ for some $\gamma \in \mathbb{C}$. A *necessary regularity condition* is: if the C-type and D-type pencils of order $n + q$ are regular, then

$$q - \text{rank } \tilde{D} \leq n \leq q,$$

where $\widehat{D} = \widetilde{D}$ and $\widehat{D} = D$, for C-type and D-type pencils, respectively.

The relation between the eigen-structure of $\lambda\mathcal{E} - \mathcal{A}$ and $\lambda\mathcal{F} - \mathcal{B}$, $(\mathcal{F}, \mathcal{B}) = \mathbf{t}(\mathcal{E}, \mathcal{A})$, can be summarized as follows (see, e.g., (Mehrmann, 1991; Xu, 2006)):

(i) $\lambda\mathcal{E} - \mathcal{A}$ is regular if and only if (iff) $\lambda\mathcal{F} - \mathcal{B}$ is regular.

(ii) $\lambda \in \Lambda(\mathcal{E}, \mathcal{A})$ iff $\mu = \mathbf{c}(\lambda) \in \Lambda(\mathcal{F}, \mathcal{B})$, and λ and μ have the same geometric, partial, and algebraic multiplicities.

(iii) If $\lambda\mathcal{E} - \mathcal{A}$ is regular, then, $\mathcal{R}_\lambda = \mathcal{R}_\mu$, $\mathcal{L}_\lambda = \mathcal{L}_\mu$, $\mu = \mathbf{c}(\lambda)$, where \mathcal{R}_λ and \mathcal{L}_λ are the *right* and *left deflating subspaces* corresponding to eigenvalue(s) λ .

The C-type pencil (1) is *skew-Hermitian/Hermitian*, i.e., $\mathcal{E}_C^H = -\mathcal{E}_C$, $\mathcal{A}_C^H = \mathcal{A}_C$, and it has the following main eigen-structure properties:

(i) $\lambda \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$ iff $-\bar{\lambda} \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$, and λ and $-\bar{\lambda}$ have the same geometric, partial, and algebraic multiplicities.

(ii) $\mathcal{R}_\lambda = \mathcal{L}_{-\bar{\lambda}}$ and $\mathcal{L}_\lambda = \mathcal{R}_{-\bar{\lambda}}$.

(iii) U is a basis matrix of a right deflating subspace of $\lambda\mathcal{E} - \mathcal{A}$ corresponding to $\lambda\mathcal{S} - T$ iff U is a basis matrix of a left deflating subspace corresponding to $\lambda(-S^H) - T^H$.

The eigenvalue pairing $(\lambda, -\bar{\lambda})$ **does not hold** for λ with $\Re(\lambda) = 0$, since then $\lambda = -\bar{\lambda}$. But for such an eigenvalue, $\mathcal{R}_\lambda = \mathcal{L}_\lambda$. This also holds for $\lambda = \infty$.

The regular D-type pencil (2) has the following main eigen-structure properties (Mehrmann, 1991; Xu, 2006):

(i) Nonzero finite eigenvalues come in pairs $(\lambda, \bar{\lambda}^{-1})$, and $\lambda, \bar{\lambda}^{-1}$ have the same geometric, partial, and algebraic multiplicities.

(ii) $\text{span } U = \mathcal{R}_\lambda$, $\text{span } V = \mathcal{L}_\lambda$ iff $\text{span } \widehat{V} = \mathcal{R}_{\bar{\lambda}^{-1}}$, $\text{span } \widehat{U} = \mathcal{L}_{\bar{\lambda}^{-1}}$, where $\text{span } X$ denotes the subspace spanned by the columns of the matrix X , and

$$U = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}, \quad V = \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} \in \mathbb{C}^{n+q, \ell},$$

$$\mathcal{E}_D U T = \mathcal{A}_D U, \quad S^H V^H \mathcal{E}_D = V^H \mathcal{A}_D, \quad (4)$$

for $T, S \in \mathbb{C}^{\ell, \ell}$ with $\Lambda(T) = \Lambda(S^H) = \{\lambda\}$, $\lambda \neq 0$ (with algebraic multiplicity ℓ), and

$$\widehat{U} = \begin{bmatrix} U_1 T \\ U_2 \end{bmatrix}, \quad \widehat{V} = \begin{bmatrix} V_1 S \\ V_2 \end{bmatrix},$$

$$\mathcal{E}_D \widehat{V} S^{-1} = \mathcal{A}_D \widehat{V}, \quad T^{-H} \widehat{U}^H \mathcal{E}_D = \widehat{U}^H \mathcal{A}_D. \quad (5)$$

Moreover, $\det V^H \mathcal{E}_D U \neq 0$ iff $\det \widehat{U}^H \mathcal{E}_D \widehat{V} \neq 0$.

(iii) $U = [U_1^T \ U_2^T]^T$ is a basis matrix of a right deflating subspace (left deflating subspace) of $\lambda\mathcal{E}_D - \mathcal{A}_D$ corresponding to $T \in \mathbb{C}^{p, p}$ nonsingular, iff $\widehat{U} =$

$[(U_1 T)^T \ U_2^T]^T$ is a basis matrix of a left deflating subspace (right deflating subspace) of $\lambda\mathcal{E}_D - \mathcal{A}_D$ corresponding to T^{-H} .

(iv) If $0 \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$ with algebraic multiplicity ℓ_0 , then $\infty \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$ with algebraic multiplicity greater than or equal to ℓ_0 .

(v) The formulas for the relations between the basis matrices of right/left deflating subspace for $\lambda = 0$ or ∞ are more complicated than for the case $\lambda \neq 0, \infty$. The sizes of the submatrices depend on the algebraic multiplicities of $0 \in \Lambda(G^H, F^H)$ and $0 \in \Lambda(F, G)$.

The eigenvalue pairing $(\lambda, \bar{\lambda}^{-1})$ **does not hold** for $|\lambda| = 1$, since then $\lambda = \bar{\lambda}^{-1}$. But for such an eigenvalue, U in (4) is a basis matrix of \mathcal{R}_λ iff \widehat{U} in (5) is a basis matrix of \mathcal{L}_λ .

Eigenvalues 0 and ∞ are **paired in a weak sense**, since the algebraic multiplicity of ∞ may be greater than or equal to the algebraic multiplicity of 0, and \mathcal{R}_0 and \mathcal{L}_0 are only related to certain subspaces of \mathcal{L}_∞ and \mathcal{R}_∞ , respectively.

The *equivalence relation between D-type and C-type pencils* is shown below.

Assume $(\mathcal{E}_C, \mathcal{A}_C) = \mathbf{t}(\mathcal{E}_D, \mathcal{A}_D)$ and that $\lambda\mathcal{E}_D - \mathcal{A}_D$ (or $\lambda\mathcal{E}_C - \mathcal{A}_C$) is regular. Then,

(i) $\lambda \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$, $\lambda \neq -1, \infty$, iff $\mu = \mathbf{c}(\lambda) \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$, $\mu \neq \infty, 1$, and λ and μ have the same geometric, partial, and algebraic multiplicities.

(ii) $\text{span } U = \mathcal{R}_\lambda^D$, $\text{span } V = \mathcal{L}_\lambda^D$ iff $\text{span } \widetilde{U} = \mathcal{R}_\mu^C$, $\text{span } \widetilde{V} = \mathcal{L}_\mu^C$, where the superscript C or D refers to (1) or (2), respectively, U and V satisfy (4) for $T, S \in \mathbb{C}^{\ell, \ell}$ with $\Lambda(T) = \Lambda(S^H) = \{\lambda\}$ (with algebraic multiplicity ℓ), and

$$\widetilde{U} = \begin{bmatrix} U_1(I+T) \\ 2U_2 \end{bmatrix}, \quad \widetilde{V} = \begin{bmatrix} V_1(I+S) \\ 2V_2 \end{bmatrix},$$

$$\mathcal{E}_C \widetilde{U} \widetilde{T} = \mathcal{A}_C \widetilde{U}, \quad \widetilde{S}^H \widetilde{V}^H \mathcal{E}_C = \widetilde{V}^H \mathcal{A}_C,$$

where $\widetilde{T} = \mathbf{c}(T)$, $\widetilde{S} = \mathbf{c}(S)$, $\Lambda(\widetilde{T}) = \Lambda(\widetilde{S}^H) = \{\mu\}$, $\mu = \mathbf{c}(\lambda)$. Moreover, $\det V^H \mathcal{E}_D U \neq 0$ iff $\det \widetilde{V}^H \mathcal{E}_C \widetilde{U} \neq 0$.

(iii) If $-1 \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$, with algebraic multiplicity ℓ_{-1} , then $\infty \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$, with algebraic multiplicity greater than or equal to ℓ_{-1} . Suppose also $-1 \in \Lambda(G^H, F^H)$, with algebraic multiplicity r_1 . Let

$$U = \begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix} \in \mathbb{C}^{n+q, \ell_{-1}}, \quad U_{11} \in \mathbb{C}^{n, r_1},$$

$$\mathcal{E}_D U T = \mathcal{A}_D U, \quad \text{rank } \mathcal{E}_D U = \ell_{-1},$$

$$T = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix} \in \mathbb{C}^{\ell_{-1}, \ell_{-1}}, \quad T_{11} \in \mathbb{C}^{r_1, r_1},$$

with $\Lambda(T) = \{-1\}$. If U is a basis matrix of \mathcal{R}_{-1}^D , then the columns of

$$\widetilde{U} = \begin{bmatrix} 2U_{11} & U_{12}(T_{22} + I) \\ 0 & 2U_{22} \end{bmatrix}$$

span an ℓ_{-1} -dimensional (right and left) deflating subspace of $\lambda\mathcal{E}_C - \mathcal{A}_C$ corresponding to eigenvalue ∞ .

(iv) Let ℓ_{-1} , ℓ_0 , and ℓ_∞ be the algebraic multiplicities of the eigenvalues $-1, 0, \infty \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$ and $\tilde{\ell}_1, \tilde{\ell}_\infty$ the algebraic multiplicities of the eigenvalues $1, \infty \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$. Then, $\ell_0 = \tilde{\ell}_1$ and

$$\tilde{\ell}_\infty = \ell_\infty - \ell_0 + \ell_{-1}, \quad \ell_\infty = \tilde{\ell}_\infty - \ell_{-1} + \tilde{\ell}_1.$$

Specifically, with \mathbf{t} , $\infty \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$ comes from $-1 \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$ (with multiplicity ℓ_{-1}) and $\infty \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$ (with multiplicity $\ell_\infty - \ell_0$).

If $-1 \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$ (i.e., $0 \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$), then $1 \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$, and it comes from $\infty \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$. Only part of $\infty \in \Lambda(\mathcal{E}_D, \mathcal{A}_D)$ is transformed into 1, to match $-1 \in \Lambda(\mathcal{E}_C, \mathcal{A}_C)$.

3 DEFLATING SUBSPACES FOR SKEW-HAMILTONIAN/HAMILTONIAN PENCILS

The *structure-preserving algorithms and software* are more advanced for CAREs, based on deflating subspaces for skew-Hamiltonian/Hamiltonian pencils. Extensions of the *HAPACK approach* are currently under development. In the sequel, the pencils $\lambda M - N$ will be represented in the numerically better form $\alpha M - \beta N$, with $\lambda = \alpha/\beta$ (possibly ∞).

Since the structured algorithms for skew-Hamiltonian/Hamiltonian pencils work on problems with even size, a basic idea is to embed the matrix pencil, adding $k \geq 0$ fictitious controls, so that $m+k$ is even. The solution of the optimal control problem corresponding to CARE, hence to

$$\alpha\mathcal{E}_c - \beta\mathcal{A}_c = \alpha \begin{bmatrix} E & 0 & 0 \\ 0 & -E^H & 0 \\ 0 & 0 & 0 \end{bmatrix} - \beta \begin{bmatrix} A & 0 & B \\ Q & A^H & L \\ L^H & B^H & R \end{bmatrix},$$

is unchanged for k new controls, with $\tilde{B} = 0_{n \times k}$, $\tilde{R} = I_k$, and D replaced by block-diag(D, \tilde{R}), with

$$D := \begin{bmatrix} Q & L \\ L^H & R \end{bmatrix}.$$

Partition, with $\ell = (m+k)/2$, $B_i \in \mathbb{C}^{n \times \ell}$, $L_i \in \mathbb{C}^{n \times \ell}$, $R_{ij} \in \mathbb{C}^{\ell \times \ell}$, $i, j = 1, 2$,

$$\begin{bmatrix} B & \tilde{B} \end{bmatrix} = \begin{bmatrix} B_1 & B_2 \end{bmatrix}, \quad \begin{bmatrix} Q & L & 0 \\ L^H & R & 0 \\ 0 & 0 & \tilde{R} \end{bmatrix} = \begin{bmatrix} Q & L_1 & L_2 \\ L_1^H & R_{11} & R_{12} \\ L_2^H & R_{21} & R_{22} \end{bmatrix}.$$

Reordering the variables and equations, the following

skew-Hamiltonian/Hamiltonian pencil is obtained

$$\alpha\mathcal{E}_c^e - \beta\mathcal{A}_c^e = \alpha \begin{bmatrix} E & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & E^H & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} - \beta \begin{bmatrix} A & B_1 & 0 & B_2 \\ L_2^H & R_{12}^H & B_2^H & R_{22} \\ -Q & -L_1 & -A^H & -L_2 \\ -L_1^H & -R_{11} & -B_1^H & -R_{12} \end{bmatrix}. \quad (6)$$

Let $\alpha\mathcal{S} - \beta\mathcal{H}$ be skew-Hamiltonian/Hamiltonian, i.e., $(\mathcal{S}\mathcal{J})^H = -\mathcal{S}\mathcal{J}$, $(\mathcal{H}\mathcal{J})^H = \mathcal{H}\mathcal{J}$. For some cases, including in linear-quadratic optimization applications, \mathcal{S} is given in a factored form, the so-called *skew-Hamiltonian Cholesky factorization*, defined by $\mathcal{S} = \mathcal{J}\mathcal{Z}^H\mathcal{J}^T\mathcal{Z}$. For instance, in (6) with $\mathcal{S} = \mathcal{E}_c^e$,

$$\mathcal{Z} = \begin{bmatrix} I_n & 0 & 0 & 0 \\ 0 & I_\ell & 0 & 0 \\ 0 & 0 & E^H & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

A skew-Hamiltonian matrix having such a factorization is said to be *\mathcal{J} -semidefinite*.

Some properties of skew-Hamiltonian/Hamiltonian pencils are summarized below (Benner et al., 2002):

- (i) Real skew-Hamiltonian matrices are \mathcal{J} -semidefinite.
- (ii) The structure of skew-Hamiltonian/Hamiltonian matrix pencils is preserved under \mathcal{J} -congruence:

$$\alpha\mathcal{S} - \beta\mathcal{H} \rightarrow \mathcal{J}\mathcal{Y}^H\mathcal{J}^T(\alpha\mathcal{S} - \beta\mathcal{H})\mathcal{Y},$$

for \mathcal{Y} nonsingular.

- (iii) A skew-Hamiltonian matrix \mathcal{S} of order $2n$ is \mathcal{J} -semidefinite (\mathcal{J} -definite) iff $\iota\mathcal{J}\mathcal{S}$ has at most (exactly) n positive and at most (exactly) n negative eigenvalues, where $\iota := (-1)^{1/2}$.

- (iv) If \mathcal{S} is skew-Hamiltonian (\mathcal{H} is Hamiltonian) and there is \mathcal{Y} nonsingular, such that

$$\begin{aligned} \mathcal{J}\mathcal{Y}^H\mathcal{J}^T\mathcal{S}\mathcal{Y} &= \begin{bmatrix} S_{11} & S_{12} \\ 0 & S_{11}^H \end{bmatrix} \\ (\mathcal{J}\mathcal{Y}^H\mathcal{J}^T\mathcal{H}\mathcal{Y}) &= \begin{bmatrix} H_{11} & H_{12} \\ 0 & -H_{11}^H \end{bmatrix} \end{aligned}$$

with $S_{11}, S_{12} (H_{11}, H_{12}) \in \mathbb{C}^{n \times n}$, then $\mathcal{S} (\iota\mathcal{H})$ is \mathcal{J} -semidefinite.

- (v) Let $\alpha\mathcal{S} - \beta\mathcal{H}$ be regular skew-Hamiltonian/Hamiltonian with v pairwise distinct, nonzero finite eigenvalues $\iota\alpha_i$, of algebraic multiplicity p_i , and associated right deflating subspace Q_i , $i = 1 : v$; let p_∞, Q_∞ , be defined similarly for eigenvalue ∞ . The following statements are equivalent:

(a) There exists a nonsingular matrix \mathcal{Y} , such that

$$\mathcal{Y} \mathcal{Y}^H \mathcal{Y}^T (\alpha \mathcal{S} - \beta \mathcal{H}) \mathcal{Y} = \alpha \begin{bmatrix} S_{11} & S_{12} \\ 0 & S_{11}^H \end{bmatrix} - \beta \begin{bmatrix} H_{11} & H_{12} \\ 0 & -H_{11}^H \end{bmatrix}, \quad (7)$$

where S_{11} and H_{11} are upper triangular, S_{12} skew-Hermitian, and H_{12} Hermitian.

(b) There exists a unitary matrix Q , such that (7) holds for \mathcal{Y} replaced by Q .

(c) $Q_k^H \mathcal{J} \mathcal{S} Q_k$ is congruent to a $p_k \times p_k$ copy of \mathcal{J} , $k = 1, 2, \dots, v$; $Q_\infty^H \mathcal{J} \mathcal{H} Q_\infty$ is congruent to a $p_\infty \times p_\infty$ copy of \mathcal{J} .

(vi) *Factored version:* Let $\alpha \mathcal{S} - \beta \mathcal{H}$ be a skew-Hamiltonian/Hamiltonian pencil with nonsingular \mathcal{J} -semidefinite skew-Hamiltonian part $\mathcal{S} = \mathcal{J} \mathcal{Z}^H \mathcal{J}^T \mathcal{Z}$. If any of the equivalent statements above holds, then there is a unitary matrix Q and a unitary symplectic matrix \mathcal{U} , such that

$$\begin{aligned} \mathcal{U}^H \mathcal{Z} Q &= \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & Z_{22} \end{bmatrix}, \\ \mathcal{J} Q^H \mathcal{J}^T \mathcal{H} Q &= \begin{bmatrix} H_{11} & H_{12} \\ 0 & -H_{11}^H \end{bmatrix}, \end{aligned}$$

where Z_{11} , Z_{22}^T , and H_{11} are $n \times n$ upper triangular.

(vii) If \mathcal{H} is also nonsingular \mathcal{J} -semidefinite, i.e., $\mathcal{H} = \mathcal{J} \mathcal{W}^H \mathcal{J}^T \mathcal{W}$, then

$$\begin{aligned} \mathcal{U}^H \mathcal{Z} Q &= \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & Z_{22} \end{bmatrix}, \\ \mathcal{U}^H \mathcal{W} Q &= \begin{bmatrix} W_{11} & W_{12} \\ 0 & W_{22} \end{bmatrix}, \end{aligned}$$

where Z_{11} , Z_{22}^T , W_{11} , and W_{22}^T are $n \times n$ upper triangular.

(viii) *Factored version, real skew-Hamiltonian/skew-Hamiltonian case:* Let $\alpha \mathcal{S} - \beta \mathcal{N}$ be a real regular skew-Hamiltonian/skew-Hamiltonian pencil with $\mathcal{S} = \mathcal{J} \mathcal{Z}^T \mathcal{J}^T \mathcal{Z}$. Then, there is an orthogonal matrix Q and an orthogonal symplectic matrix \mathcal{U} , such that

$$\begin{aligned} \mathcal{U}^T \mathcal{Z} Q &= \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & Z_{22} \end{bmatrix}, \\ \mathcal{J} Q^T \mathcal{J}^T \mathcal{N} Q &= \begin{bmatrix} N_{11} & N_{12} \\ 0 & N_{11}^T \end{bmatrix}, \end{aligned}$$

where Z_{11} , Z_{22}^T are upper triangular, N_{11} is upper quasi-triangular, and $N_{12} = -N_{12}^T$. Moreover,

$$\mathcal{J} Q^T \mathcal{J}^T (\alpha \mathcal{S} - \beta \mathcal{N}) Q = \alpha \begin{bmatrix} Z_{22}^T Z_{11} & Z_{22}^T Z_{12} - Z_{12}^T Z_{22} \\ 0 & Z_{11}^T Z_{22} \end{bmatrix} - \beta \begin{bmatrix} N_{11} & N_{12} \\ 0 & N_{11}^T \end{bmatrix}$$

is a \mathcal{J} -congruent skew-Hamiltonian/skew-Hamiltonian matrix pencil.

Comments:

1. The result (viii) above is used to compute the *struc-*

tured Schur form of order $4n$ for a **complex** skew-Hamiltonian/Hamiltonian pencil.

2. The periodic QZ algorithm is used.

3. Algorithms for eigenvalue reordering and deflating subspace computation are available.

Below is a summary about the related software:

- Fortran and MATLAB software for eigenvalues and deflating subspaces are under development.
- Both real and complex cases are considered.
- Factored or unfactored versions are covered.
- Auxiliary routines for problems (of even order) with (quasi-)triangular structure are included.
- Optimized kernels for problems of order 2, 3, or 4, called by the general solvers, are available.

To compute or reorder the eigenvalues, the computations begin with an initial reduction, called *generalized symplectic URV decomposition*, whose *factored version* is defined as follows (Benner et al., 2007):

Given a real skew-Hamiltonian/Hamiltonian $2n \times 2n$ pencil $\alpha \mathcal{T} \mathcal{Z} - \beta \mathcal{H}$, orthogonal matrices Q_1 , Q_2 and orthogonal symplectic matrices \mathcal{U}_1 , \mathcal{U}_2 are determined, such that

$$\begin{aligned} Q_1^T \mathcal{T} \mathcal{U}_1 &= \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix}, \\ \mathcal{U}_2^T \mathcal{Z} Q_2 &= \begin{bmatrix} Z_{11} & Z_{12} \\ 0 & Z_{22} \end{bmatrix}, \\ Q_1^T \mathcal{H} Q_2 &= \begin{bmatrix} H_{11} & H_{12} \\ 0 & H_{22} \end{bmatrix}, \end{aligned}$$

where T_{11} , T_{22}^T , Z_{11} , Z_{22}^T , and H_{11} are upper triangular, and H_{22}^T is upper quasi-triangular. The matrices \mathcal{U}_1 and \mathcal{U}_2 are stored compactly (the first n rows only), since, for $i = 1, 2$,

$$\mathcal{U}_i = \begin{bmatrix} U_{i1} & U_{i2} \\ -U_{i2} & U_{i1} \end{bmatrix}.$$

4 NUMERICAL RESULTS

This section presents some preliminary numerical results. These results have been obtained on a portable Intel Dual Core computer at 2 GHz, with 2 GB RAM, and relative machine precision $\varepsilon \approx 1.11 \times 10^{-16}$, using Windows XP (Service Pack 2) operating system, Intel Visual Fortran 11.1 compiler, and MATLAB 7.8.0.347 (R2009a). The matrices

$$\mathcal{S} = \begin{bmatrix} A & B \\ C & A^H \end{bmatrix}, \quad \mathcal{H} = \begin{bmatrix} D & E \\ F & -D^H \end{bmatrix},$$

where $A, B, C, D, E, F \in \mathbb{C}^{n \times n}$, have been initially generated with MATLAB commands of the form

```
list
```

```
A = 10*rand(n)-5 + (10*rand(n)-5)*1i;
```

where `rand` is the uniform (0,1) random generator, and `1i` is the MATLAB notation for the purely imaginary unit, i . Then, the B , C , E , and F matrices have been transformed using the formulas

$$\begin{aligned} B &:= B - B^H, & B &:= B/2; & C &:= C - C^H, \\ E &:= E + E^H, & E &:= E/2; & F &:= F + F^H, \end{aligned}$$

to become skew-Hermitian, and Hermitian, respectively. Therefore, the pencil $\lambda S - \mathcal{H}$ is skew-Hamiltonian/Hamiltonian.

The order n took the values $n = 100, 200, \dots, 800$. For each order $n \leq 500$, 10 problems have been solved, and the means of the results are reported. For larger n values, one problem has been solved for each n . The generalized eigenvalues computed by a structure-preserving algorithm have been compared with those delivered by the standard QZ algorithm, optimally implemented in the MATLAB function `eig`.

Fig. 1 presents the ratios of the mean CPU times, in seconds, i.e., the speed-up factor of the structured algorithm, in comparison with the standard algorithm.

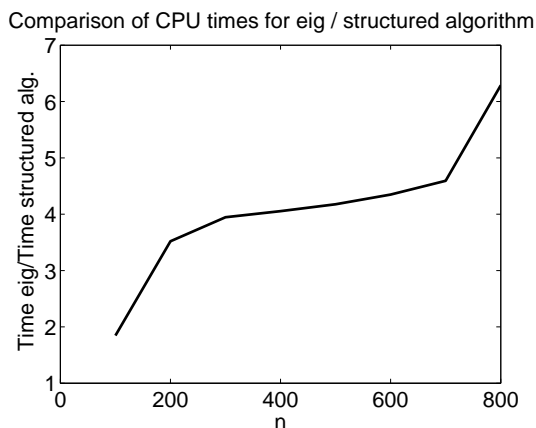


Figure 1: Ratios between the CPU times needed by the MATLAB function `eig` and the structure-preserving algorithm for randomly generated complex skew-Hamiltonian/Hamiltonian pencils of order $2n$.

The deviation from symmetry of the eigenvalues computed by `eig` has also been computed as the difference between the vector of eigenvalues $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_{2n}]^T$ and a permutation of the elements of the vector $-\bar{\lambda}$, chosen so that the elements with the same indices in the two vectors be as close as possible. The largest norm has been $4 \cdot 10^{-10}$, and the smallest norm has been $1.90 \cdot 10^{-12}$. The norms should theoretically be 0.

5 CONCLUSIONS

Main issues related to the structure-preserving algorithms for solving discrete-time algebraic matrix Riccati equations are summarized. Stable deflating subspaces for extended, inverse-free symplectic matrix pencils, are computed. Algorithms based on skew-Hamiltonian/Hamiltonian pencils derived by an extended Cayley transformation, which only involves matrix additions and subtractions, are considered. The preliminary results are encouraging.

ACKNOWLEDGEMENTS

The work was partially supported by the German Research Foundation (DFG) and The MathWorks, Inc. The collaboration with Peter Benner and Matthias Voigt from TU Chemnitz is highly acknowledged.

REFERENCES

- Bender, D. J. and Laub, A. J. (1987a). The linear-quadratic optimal regulator for descriptor systems. *IEEE Trans. Automat. Contr.*, AC-32(8):672–688.
- Bender, D. J. and Laub, A. J. (1987b). The linear-quadratic optimal regulator for descriptor systems: Discrete-time case. *Automatica*, 23(1):71–85.
- Benner, P., Byers, R., Losse, P., Mehrmann, V., and Xu, H. (2007). Numerical solution of real skew-Hamiltonian/Hamiltonian eigenproblems. Technical report, Technische Universität Chemnitz, Chemnitz.
- Benner, P., Byers, R., Mehrmann, V., and Xu, H. (2002). Numerical computation of deflating subspaces of skew Hamiltonian/Hamiltonian pencils. *SIAM J. Matrix Anal. Appl.*, 24(1):165–190.
- Lancaster, P. and Rodman, L. (1995). *The Algebraic Riccati Equation*. Oxford University Press, Oxford.
- Mehrmann, V. (1991). *The Autonomous Linear Quadratic Control Problem. Theory and Numerical Solution*, volume 163 of *Lect. Notes in Control and Information Sciences*. Springer-Verlag, Berlin.
- Paige, C. and Van Loan, C. F. (1981). A Schur decomposition for Hamiltonian matrices. *Lin. Alg. Appl.*, 41:11–32.
- Pappas, T., Laub, A. J., and Sandell, N. R. (1980). On the numerical solution of the discrete-time algebraic Riccati equation. *IEEE Trans. Automat. Contr.*, AC-25(4):631–641.
- Van Dooren, P. (1981). A generalized eigenvalue approach for solving Riccati equations. *SIAM J. Sci. Stat. Comput.*, 2(2):121–135.
- Xu, H. (2006). On equivalence of pencils from discrete-time and continuous-time control. *Lin. Alg. Appl.*, 414:97–124.

CONTROL OF NETWORKED SYSTEMS CONTAINING MULTIPLE AGENTS

Jose B. Cruz Jr.^{1,2}, Gregory Tangonan², Raymond R. Tan³, Nathaniel Libatique²
Fabian M. Dayrit² and Alvin B. Culaba³

¹*The Ohio State University, Columbus, U.S.A.*

²*Ateneo De Manila University, Manila, Philippines*

³*De La Salle University, Manila, Philippines*

jbacruz@ieee.or, {g raymond.tan, alvin.culaba}@dlsu.edu.ph, goriot@mac.com, {nlibatique, fdayrit}@ateneo.edu

Keywords: Multi-agent Systems, Game Theory, Complex Systems, Networked Control Systems.

Abstract: This position paper illustrates the use of a natural framework for the modelling, analysis, and design of engineering systems that involve two or more controllers, each of which has an associated objective function. Such systems arise when ordinary single controller systems are networked through communication links so that the information available to each controller may contain aspects of the other systems' states and the optimization of each objective function is no longer decoupled from each other. Single controller optimization is no longer directly applicable. Appropriate to the study of such systems is the theory of games that has been developing in mathematics, economics, and engineering for the past 60 years. There are extensive applications in economics, but in engineering the applications are scarce. In recent years, there has been great attention to global problems such as the negative environmental impact of energy use, and global warming. These problems arise from complex systems with multiple controllers. Among the approaches for dealing with the problems, there should be one on a total systems approach with a game theory base. A natural framework for this is the subject of this policy paper.

1 INTRODUCTION

In this position paper we establish the benefits and advantages of explicitly including multiple agents in the modelling and control of networked engineering systems, when in fact, multiple agents are present in the application systems. The agents are not necessarily cooperating in a team and not necessarily antagonistic against each other, although in some applications they might be cooperating as a team. Some global problems with significant technological components are (a) integration of renewable energy sources (such as solar, geothermal, wind, hydro, and biological) with the traditional fossil energy sources, to reduce negative impact on the environment, (b) recycling of wastewater to produce clean water, to conserve scarce fresh water resources, (c) mitigating damages due to disasters such as typhoons, hurricanes, floods, and earthquakes. These national and international problems are also examples of complex systems. Complexity arises because of large numbers of smaller systems that are networked

together, and total system behaviour is not easily inferred from individual behaviours of the component systems. These complex systems are characterized by the presence of many stakeholders, starting from the national government, to provincial/state governments, private enterprises of suppliers, industry associations, and large blocks of consumers. The stakeholders generally have policies that translate to actions affecting the system. Notwithstanding announced plans to the contrary, the complex systems typically evolve piece-meal, and unexpected and undesirable effects are addressed piece-meal. Finally, complex systems have numerous time lags throughout and stability is a crucial issue that could lead to a total collapse if not addressed properly.

A specific scenario for a networked system is the following: we can project a boom in ethanol production in some countries, even exporting of ethanol (after meeting local needs for ethanol), using sugar cane. For the same country can also project a return to the export of sugar as well. The sugar industry is terribly inefficient and unkind to labour -

six months of work and down for 6 months. The ethanol industry needs sugar cane 12 months of the year. The options for farmers have just changed - what is their way of optimizing their land and/or labour? The government needs to figure out how to provide a good price of sugar for consumers, to protect the growing ethanol market, and to encourage investment by distillers moving into the ethanol industry through incentives such as tax breaks. This ecosystem is extremely rich in control problems to consider. Various stakeholders or agents need to understand various solution concepts and different perspectives. A decision support system that reflects the interactions of all the stakeholders in the modelling and control strategies would be highly useful.

The presence of multiple stakeholders in complex systems can be studied through mathematical game theory. Much of the literature is for static systems with very few exceptions (Nash 1950, Nash 1951, von Neumann and Morganstern 1947, von Stackelberg 1952). Dynamics can be studied using standard methods in mathematics and engineering. Yet, there are no general computer-based decision aids as tools for the design and analysis of dynamic complex systems. To be sure the challenge is not trivial and it requires significant research effort.

The bulk of control theory pertains to dynamic systems with a single controller (Bellman 1957, Pontryagin, Boltyanskii, Gamkrelidze, Mischenko 1962). With the emergence of networked control systems (Wang and Liu 2008) whereby previously separate individual subsystems with their individual controls are linked through communication networks, the information available at each subsystem through the links may generally contain aspects of the other systems. If a subsystem controller were to optimize a performance criterion associated with that subsystem, the performance criterion may contain variables pertaining to the other subsystems because of the network links and a dilemma of how to proceed is encountered.

There have been design approaches that ignore the presence of the links in the network. Subsequently the systems are analyzed to check robustness against the neglected connections. In the case of two interconnected control systems, worst case scenarios have been assumed in the design, in the sense that the controller of the other subsystem is assumed to make the performance criterion of the system as bad as possible while the controller of the system maximizes the performance of the resulting worst case. The theory of maxmin (or minmax)

optimization is applied. In many applications the performance criteria of the two control systems are not opposite of each other, so that the minmax design is overly conservative. Furthermore, when the system is dynamic and the controls involve feedback, the structure of the feedback control of the other system needs to be known, in order to perform a correct minmax optimization with the correct total system dynamics. Thus the minmax approach could be problematic unless the other system including its feedback structure is modelled properly.

When the agents are cooperating, the individual performance criteria may be combined as a single convex linear combination using the Pareto-optimal concept, and once again the theory for single controller systems may be used. In applications, there is the additional task of choosing the weights in the convex linear combination. A special case arises when the performance criteria are identical and the choice of weights is immaterial. This is the team-optimal problem.

Unlike engineering systems, economic systems are modelled, analyzed, and optimized using multiple agents. In fact in the ideal case, there is an infinite number of consumers and an infinite number of suppliers interacting in a market. However, the bulk of the literature is for static systems with very few exceptions. The bulk of the literature in the field of operations research with respect to multiple agents is likewise for static systems.

2 NETWORKED STATIC CONTROL SYSTEMS

To focus on the effects of multiple agents in a networked control system, let us initially consider static systems to eliminate one dimension of complexity induced by dynamics.

Furthermore, even the static area could be utilized to great advantage in dealing with large complex systems. Suppose we have two networked control systems of producers of renewable energy. The productions are modelled by

$$x_1 = f_1(u_1) \quad (1)$$

$$x_2 = f_2(u_2) \quad (2)$$

where x_1 and x_2 are quantities of renewable energy produced by Companies 1 and 2 respectively, and u_1 and u_2 are resources (controls) used to produce x_1 and x_2 respectively. The functions f_1 and f_2 are monotonically increasing so their unique inverses exist. The renewable energy products are sold in a

market and the price P is determined from a supply curve that relates price to total demand X

$$P = m_1 X + b_1 \quad (3)$$

and the total demand X is equal to the total supply

$$X = x_1 + x_2 \quad (4)$$

The parameters m_1 and b_1 are given and $m_1 > 0$, $b_1 < 0$. The costs for producing the renewable energy products are

$$C1 = g_1(x_1) \quad (5)$$

$$C2 = g_2(x_2) \quad (6)$$

where g_1 and g_2 are nonlinear functions. Each company wants to maximize its profit, which is revenue minus cost. For Company 1 the profit is

$$Profit1 = Px_1 - g_1(x_1) \quad (7)$$

$$Profit2 = Px_2 - g_2(x_2) \quad (8)$$

It is not a simple matter for Company 1 to maximize its profit Profit1 with respect to x_1 because P in Profit1 contains x_2 , which is not under its control. Similarly for Company 2, it is not a simple matter to maximize Profit2 with respect to x_2 because P contains x_1 .

This illustrates the intrinsic difference between a single controller problem and a problem with multiple controllers or multiple agents, such as the example above where the two static control systems are networked through the market mechanism where their outputs are sold. From the point of view of single controller theory, for example in the design of u_1 , Company 1 may simply assume a value for u_2 or x_2 and proceed to maximize Profit1 with respect to x_1 . Except for some very lucky choice of u_2 by Company 1, when Company 2 chooses u_2 using an assumed value of u_1 , Company 2 will obtain a value of u_2 different from the one assumed by Company 1, posing a dilemma for both companies.

Next let us consider each company's worst-case design whereby Company 1 assumes that Company 2 chooses u_2 to minimize Profit1. Then Company 1 maximizes Profit1 resulting in its maxmin (or worst-case) design. Similarly Company 2 may proceed to calculate its worst-case design. When both apply their worst-case designs, their resulting profits will be generally higher but in any case no worse than the worst-case profits they previously calculated. The pair of separately calculated worst-case controls will generally not lead to the worst case for either company. Still, in general their designs would be conservative because the companies are not trying to

destroy each other by making each other's profit as small as possible.

In the theory of games that applies to systems with multiple agents, there are many solution concepts that go beyond single controller optimization. For example, one may consider the Nash equilibrium concept (Nash 1950, Nash 1951) whereby when (u_1, u_2) is a Nash solution pair and Company 1 chooses a control u_1 that is different from u_1^N , but Company 2 still uses u_2^N , the resulting profit of Company 1 can not be higher than that when both use their Nash controls. There is also a Stackelberg (von Stackelberg 1952) or Leader-Follower solution concept whereby one subsystem controller is dominant or more powerful than the others. The leader's control is announced in advance and all other controllers know what the leader's control is before they choose their own controls. In the case of Stackelberg games, it is of particular interest to determine the role of the dominant player in inducing desirable behaviour from low-level players through incentives or disincentives. The implications of such mechanisms are clearly evident for situations in which, for example, it is desired to determine government policies to facilitate environmentally beneficial behaviour from the private sector (e.g., Aviso et al 2010). The Stackelberg hierarchy may have more than two levels.

3 NETWORKED DYNAMIC CONTROL SYSTEMS

A system that is more general than the class considered in Section 2 is one where the individual control systems are dynamic. If the systems are discrete-time the individual models may be of the form

$$x_i(k+1) = f_i(x_i(k), u_i(k), k) \quad (9)$$

$i = 1, \dots, N, k = 0, \dots, M$, where x_i is the state vector of system i , with dimension n_i ; u_i is the control vector of system i , with dimension m_i , and k is discrete time with integer values from 0 to M . The vector functions f_i are mappings from the spaces of the arguments to the space of x_i and n_i, m_i, N and M are given. The network connections may be modelled by an algebraic equation

$$G(x_1, \dots, x_n) = 0 \quad (10)$$

where G may be a scalar or vector of a given dimension. Associated with each system i , may be a scalar performance index or cost function

$$J_i = C_i(x_i(M)) + \sum_{j=0}^{M-1} L_i(x_i(j), u_i(j)) \quad (11)$$

If J_i represents total cost for the entire horizon from 0 to M , then $C_i(x_i(M))$ represents the incremental cost at the terminal time, and $L_i(x_i(j), u_i(j))$ represents the incremental cost during time j . As in the static case, there will be a dilemma in a simple dynamic optimization of J_i , in Equation (11) with respect to the control vector $u_i(j)$ sequence because the control may be in feedback form and even if only $x_i(j)$ is used for feedback at time j , x_i is not independent of the other system states because of the network coupling modelled by Equation (10).

A more general effect of the network connection represented by Equation (10) may be a change in the individual control system model from Equation (9) to Equation (12)

$$x_i(k + 1) = f_i(x_1(k), \dots, x_N(k), u_i(k)) \quad (12)$$

$$i = 1, \dots, N, k = 0, \dots, M$$

and the constraint represented by Equation (10) may remain.

For systems that are modelled as continuous-time processes a typical description in state variable form is given by the vector differential equation

$$dx_i(t) / dt = f_i(x_1(t), \dots, x_N(t), u_i(t)) \quad (13)$$

where x_i is the state vector of system i , with dimension n_i ; u_i is the control vector of system i , with dimension m_i , and t is continuous time with values in the interval $[0, T]$, and T is a specified real number. Instead of the cost function in Equation (11) the continuous time version is an integral analogous to the sum in Equation (11)

$$J_i = \phi_f(x_i(T)) + \int_0^T L_i(x_i(t), u_i(t), t) dt \quad (14)$$

Because of the links in the network there may be a constraint as in Equation (10). The standard dynamic optimization of the integral cost functional with respect to the vector functions $u_i(t)$ for t in the interval $[0, T]$ poses a dilemma because the functional may depend on the states of the other subsystems through the constraint in Equation (10).

In general, the direct application of dynamic optimization for single controllers becomes a problematic issue. The field of dynamic game theory

offers potential benefits in the design and analysis of such systems (Isaacs 1955, Basar and Cruz 1982, Basar and Olsder 1998, Starr and Ho 1969a, Starr and Ho 1969b, Chen and Cruz 1972, Simaan and Cruz 1973a-c, Cruz 1975, Castanon and Athans 1976). Macroeconomics has completely adopted concepts from dynamic game theory. For multiple agent dynamic engineering systems the application of multi-agent models and equilibrium theories of dynamic games would be beneficial also.

4 ILLUSTRATIVE EXAMPLE

In this section we consider a single, simplified composite energy system with only one state variable (x_i) but two decision-makers, each with a control variable. We consider only finite states of zero, 1, and 2 and finite controls 0 and 1, and two time stages. For each controller there will be an associated incremental cost at each time stage and a total cost for the two time stages. We will analyze the network using some of the solution concepts in dynamic game theory.

Table 1: State Transitions in Period 1. (Simaan and Cruz 1973b).

	Controllers' Decisions			
	(0, 0)	(0, 1)	(1, 0)	(1, 1)
$x_0 = 1$	$x_1 = 1$	$x_1 = 2$	$x_1 = 0$	$x_1 = 1$

Controller 1 is assumed to be the upstream agricultural sector of a biofuel supply chain, similar to that considered in Cruz, Tan, Culaba, and Balacillo 2009, while Controller 2 is the downstream sector comprised of the biofuel processing sector. In each time period, each controller is faced with the option of expanding ($u = 0$) or maintaining ($u = 1$) current production capacity. The composite system is described by a trivalent state variable which indicates upstream (agricultural) deficit ($x = 0$), balanced production ($x = 1$), or upstream surplus ($x = 2$). This is a biofuel supply chain interpretation of the numerical example that appeared in Starr and Ho 1969 a,b and Simaan and Cruz 1973a,b. Tables 1 and 2, which are based on the game described in Simaan and Cruz 1973b, show the possible state transitions arising from decisions in this stylized energy system. As each controller is faced with a binary decision in a given time period, over the entire planning horizon each will have four possible decisions, namely, (0, 0), (0, 1), (1, 0) and (1, 1). For open-loop control structure, i.e., the controls are

functions of time (stage) only, each of the two controllers have four possible decision sequences, and the two-stage game may be expressed in bi-matrix form as in Table 3, wherein the first and second entries are the cumulative costs borne by the two controllers over the entire time horizon. See Simaan and Cruz 1973b for the incremental costs.

Table 2: State Transitions in Period 2 (Simaan and Cruz 1973b).

	Controllers' Decisions			
	(0, 0)	(0, 1)	(1, 0)	(1, 1)
$x_1 = 0$	$x_2 = 2$	$x_2 = 1$	$x_2 = 0$	$x_2 = 0$
$x_1 = 1$	$x_2 = 1$	$x_2 = 2$	$x_2 = 0$	$x_2 = 1$
$x_1 = 2$	$x_2 = 2$	$x_2 = 2$	$x_2 = 1$	$x_2 = 0$

Table 3: Cumulative Cost for Open-Loop Bi-Matrix Game (Simaan and Cruz, 1973b).

		Controller 2			
		(0, 0)	(0, 1)	(1, 0)	(1, 1)
Controller 1	(0, 0)	8, 8	11, 6**	10, -2	11, 0
	(0, 1)	6, 4	12, 3	7, 2	12, 4
	(1, 0)	5, 12	20, 15	5, 11	8, 9*
	(1, 1)	6, 5***	16, 7	3, 7	9, 6
*Nash equilibrium					
**Open-loop Stackelberg equilibrium with Controller 2 as leader					
***Open-loop Stackelberg equilibrium with Controller 1 as leader					

If we assume that neither the upstream nor downstream sectors of the energy supply chain dominate the game, the system naturally tends toward the Nash equilibrium as indicated in Table 3. In this case, each decision maker identifies his rational reaction, which is the response that minimizes his cost for any possible action by the other player. The Nash equilibrium is the intersection of the rational reactions of the two decision makers. They both commit to an open-loop sequence at the beginning of the horizon. The upstream sector maintains production capacity in the first time period, and expands production in the second period, while the downstream sector maintains its capacity throughout. As a result, the energy system is at a state of surplus farm production at the end of the time horizon analyzed. Note that this state is reached without any centralized direction, and emerges purely from the self-interested action of the two agents.

The energy system evolves differently if either

sector were dominant. For instance, if the downstream (fuel processing) sector acted as the leader, the system reaches the open-loop Stackelberg equilibrium indicated in Table 3. In this scenario, the dominant decision-maker selects his action so as to minimize his cost, having anticipated that the follower's response is the latter's rational reaction as in the Nash case. It would commit in advance that it would increase production capacity in the first period, but maintain it in the second period. The agricultural sector, the follower, would increase production in both periods. The energy supply chain thus also reaches a state of surplus upstream production capacity (i.e., excess biofuel feedstock) as in the Nash game, except that the cost burden of the farmers would have increased while those of the processors would have gone down. Note that the leader's Stackelberg solution can be no worse than his corresponding Nash solution (Simaan and Cruz, 1973a,b).

A similar analysis can be made for the case wherein the upstream sector dominates and acts as leader. In this case, an alternative Stackelberg solution is reached, as shown in Table 3, with the supply chain ending in a state of deficit in upstream production capacity. Note that in this case, both controllers incur lower cumulative costs than they do in either of the two previous scenarios. Thus, from the system-level standpoint, this solution is superior for the given transitions and payoffs.

For a closed loop structure the sectors have 16 decision choices that depend on time and the state, see Simaan and Cruz (1973b). In particular, Simaan and Cruz (1973b) showed through the examples that the leader solutions violate Bellman's principle of optimality (Bellman, 1957). In economics, this violation is known as time-inconsistency (Kydland and Prescott 1977).

5 CONCLUSIONS

In this position paper we provide a discussion of the need to use modelling and control methods that are more appropriate than those for single controller systems when we have a networked system of control systems whereby the individual systems that are networked have their individual controls and individual objective functions. This need arises because the network that may involve communication links induces interaction and complicates the choice of control strategies for the various subsystems. There are methodologies that could be applied now for multi-agent systems but

there remains further need for research to address issues such as estimation, adaptation, stability, and robustness, to name a few. Global complex systems such as reduction of the external costs of negative environmental impacts of the use of various energy sources, mitigation of natural disasters, and consideration of global warming in technological planning, are prime areas where these networked control systems methods could be beneficially applied.

REFERENCES

- Aviso, K., Tan, R. R., Culaba, A. B., and Cruz, J. B., Jr., 2010, *Bi-Level Fuzzy Optimization Approach for Water Exchange in Eco-Industrial Parks*, Process Safety and Environmental Protection, Vol. 88, pp. 31-40.
- Basar, Tamer and Cruz, Jose B., Jr., 1982, *Concepts and Methods in Multi-Person Coordination and Control*, Optimization and Control of Dynamic Operational Research Models, Ed. S.G.Tzafestas. Amsterdam: North-Holland, 455-462.
- Basar, T. and Olsder, G. J., 1998, *Dynamic Noncooperative Game Theory*. New York, NY: Academic Press, revised ed.
- Basar, T. and Selbuz, H., 1979, *Closed-loop Stackelberg strategies with applications in the optimal control of multilevel systems*, IEEE Trans. on Automatic Control, vol. AC-24, no. 2, pp. 166-179.
- Bellman, R., 1957, *Dynamic Programming*. Princeton, NJ: Princeton University Press.
- Castastanon, D. and Athans, M., 1976, *On stochastic dynamic Stackelberg strategies*, Automatica, vol. 12, pp. 177-183.
- Chen, C. I. and Cruz, J. B., Jr., 1972, *Stackelberg solution for two person games with biased information patterns*, IEEE Trans. on Automatic Control, vol. AC-17, pp. 791-797.
- Cruz, J. B., Jr., *Survey of Nash and Stackelberg Equilibrium Strategies in Dynamic Games*, Annals of Economic and Social Measurement, Vol. 4, No. 4, 1975, pp. 339-344.
- Cruz, J. B., Jr., 1978, *Leader-follower strategies for multilevel systems*, IEEE Trans. on Automatic Control, vol. AC-23, no. 2, pp. 244-255.
- Cruz, J. B., Jr., Tan, R. R. Culaba, A. B., and Ballacillo, J.A., 2009, *A Dynamic Input-Output Model for Nascent Bioenergy Supply Chains*, Applied Energy, Vol.86, Supplement 1, pp. 86-94.
- Cruz, Jose B. Jr. and Tan, Xiaohuan, 2009, *Dynamic Noncooperative Game Models for Deregulated Electricity Markets*, Nova Publishers, New York NY.
- Isaacs, R. P., 1955, *Differential Games: a Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization*. New York: John Wiley and Sons, 1955.
- Kydland, F. E. and Prescott, Edward C., *Rules Rather Than Discretion: The Inconsistency of Optimal Plans*, The Journal of Political Economy, Vol. 85 No. 3 (June 1977), pp. 473-492.
- Nash, J., 1950 *Equilibrium points in N-person games*, in Proc. of the National Academy of Sciences of the United States of America, vol. 36, pp. 48-49.
- Nash, J., 1951, *Noncooperative games*, Annals of Mathematics, vol. 54, pp. 286-295.
- Pontryagin, L. S., Boltyanskii, V. G., Gamkrelidze, R. V., and Mishchenko, E. F., 1962, *The Mathematical Theory of Optimal Processes*. New York, NY: Interscience Publishers, 1962.
- Simaan, M. and Cruz, J. B., Jr., 1973, *On the Stackelberg strategy in nonzero-sum games*, Journal of Optimization Theory and Application, vol. 11, no. 5, pp. 533-555.
- Simaan, M. and Cruz, J. B., Jr., 1973, *Additional aspects of the Stackelberg strategy in nonzero-sum games*, Journal of Optimization Theory and Application, vol. 11, no. 6, pp. 613-626.
- Simaan, M. and Cruz, J. B., Jr., 1973, *A Stackelberg solution for games with many players*, IEEE Trans. on Automatic Control, vol. AC-18, no. 3, pp. 322-324.
- Starr, A. W. and Ho, Y. C., 1969, *Nonzero Sum Differential Games*, Journal of Optimization Theory and Applications, Vol. 3, No. 3.
- Starr, A. W. and Ho, Y. C., 1969, *Further Properties of Nonzero Sum Differential Games*, Journal of Optimization Theory and Applications, Vol. 3, No. 4.
- von Neumann, J. and Morgenstern, O., 1947, *Theory of Games and Economic Behavior*. Princeton, NJ: Princeton University Press, 2nd ed.
- von Stackelberg, H., 1952, *The Theory of the Market Economy*. Oxford: Oxford University Press, English translated ed., 1952.
- Wang, Fei-yue and Liu, Derong, 2008, *Networked Control Systems: Theory and Applications*, Springer.

USING MODELICA MODELLING LANGUAGE FOR PHYSICAL PLANT PARAMETERS EVALUATION AND OPTIMIZATION

A Case Study

Eurico Seabra and José Machado

Department of Mechanical Engineering, CT2M Research Center, University of Minho, 4800-058 Guimarães, Portugal
 {eseabra, jmachado}@dem.uminho.pt

Keywords: Modeling, Simulation, Modelica Modeling Language, Hybrid Plants, Dependable Controllers.

Abstract: Modelica Modeling language is powerful and suitable for modeling mechatronic systems, being possible to interact different technological aspects and deal, simultaneously with different technologies (mechanical, electrical, pneumatic, hydraulic,...). In this paper it is discussed, in a case study, the possibility of using this language for modeling an automation system (controller and plant) in closed loop behavior and also in defining some parameters of the automation system in order to optimize some behavior aspects of the system as, for instance, the time cycle of the automation system. Some aspects related with controllers dependability are also discussed and it is showed how Modelica modeling language can help controllers' designers improving controllers dependability, when are used Simulation Techniques.

1 INTRODUCTION

There is a rapidly increasing use of computer simulations in industry to optimize products, to reduce product development costs and time by design optimization, and to train operator. Whereas in the past it was considered sufficient to simulate subsystems separately, the current trend is to simulate increasingly complex physical systems composed of subsystems from multiple domains.

In such a complex industrial process, simulation tools are extremely useful since they can contribute to higher product quality and production efficiency in several ways. For example, modifications in a plant could be tested (both statistically and dynamically) in advance in a simulator saving much of the trial and error procedure that is used nowadays; the optimization of plant behavior parameters can be performed too. Besides, a dynamic simulator of the plant and of its control would allow for a thorough study of different control strategies, and would be an efficient way to tune controllers for new equipments. Finally, a simulation tool can also be a way of training not only the operators but also the production engineers and technicians. Some tools have been developed in order to simulate the behavior of automation systems (figure 1).

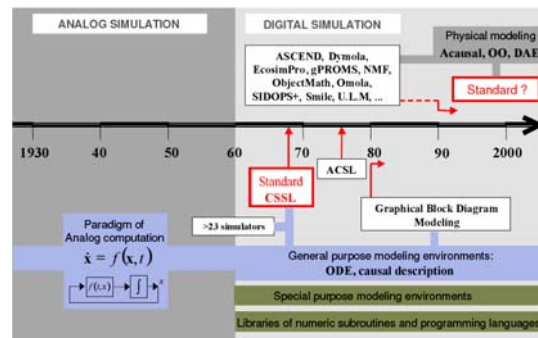


Figure 1: Evolution of modeling and simulation tools.

Graphical block diagram modeling is widely used in control engineering (Karaynakis, 1995). Some examples of languages and environments supporting this paradigm are Matlab/Simulink (Matlab, 2010), MATRIXX/SystemBuild (Matrixx, 2010), HYBRISIM (Mosterman, 2002) and ACSL Graphics Modeller (MGA Software 1996). Block diagram modeling paradigm might be considered as a heritage of analog simulation (Aström et al. 1998).

On the other hand, object-oriented modeling languages and compilers supporting the physical modeling paradigm have become available since the 1990's decade. This is driven by demands from users to be able to simulate complex multi-domain models.

In this paper it is presented a study and shown how Modelica modeling language can be used to optimize plant behavior parameters in order to guarantee the good and desired behavior for the system, in the shorter time cycle, combined with other aspects like energy consumption, for example.

To achieve the proposal goal, the section 2 is devoted to the presentation of Modelica modeling language and the Dymola Simulation environment; section 3 presents the case study that is the base for our study; section 4 discusses the mathematical modeling of the plant. Further, section 5 presents the Modelica model of the system (controller model coupled with plant model); section 6 discusses the obtained results concerning the defined plant behavior parameters to study and, finally, section 7, presents some conclusions and future works, in this field.

2 MODELICA AND DYMOLA

In the few years of research in modeling and simulation, the concept of object-oriented modeling has achieved a big relevance. Several works have demonstrated how objected oriented concepts can be successfully employed to support hierarchical structuring, reuse and evolution of large and complex models independent from the application domain and specialized graphical formalism.

To handle complex models, the reuse of standard model components is a key issue. But in order to exchange models between different packages an unified language is needed. Modelica is an object-oriented, general-purpose modeling language that is under development in an international effort to introduce an expressive standardized modeling language, see (Elmqvist and Mattson, 1997) (Fritzson and Vadim, 1998). Modelica supports object-oriented modeling using inheritance concepts taken from computer languages such as Simula and C++. It also supports non-causal modeling, meaning that model's terminals do not necessarily have to be assigned an input or output role. In fact, in the last few years it has been proved in several cases that non-causal simulation techniques perform much better than the ordinary object-oriented tools.

Modelica is a powerful programming language where equations are used for modeling of the physical phenomena. No particular variable needs to be solved for manually because the software Dymola (Dymola software, 2010) has enough information to decide that automatically. This is an important property of Dymola to enable handling of large

models having more than hundred thousand equations. Modelica supports several formalisms: ordinary differential equations (ODE), differential-algebraic equations (DAE), bond graphs, finite state automata, Petri nets, etc.

3 CASE STUDY DESCRIPTION

The case study that is proposed as base for this work is inspired on the benchmark system proposed by (Kowalewski *et al.* 2001).

Figure 2 illustrates an example of an evaporator system, which consists of two tanks, where an aqueous solution suffers transformations. In the first tank that solution should acquire a certain concentration through the heating of the solution using an electrical resistance (H1) which provokes the steam formation.

Associated to the tank1 (figure 2) exist a condenser (C) responsible for the condensation of the steam that however it was formed. The cooling, in that condenser, it is done through the circulation of a cooling liquid (whose flow is measured by sensor FIS) that passes through the cooling circuit (if open the valve V13).

Associate to the tank1 there are a group of sensors: level sensors (maximum (LIS1) and minimum (LI1)), temperature sensor (acceptable maximum (TIS1)); sensor of conductivity (QIS) that is to indicate the desired concentration; they also exist several actuators: filling valve of the tank1 (V12), drain valve (V16) and emptying valve (V15), that it is also the filling valve of the tank2.

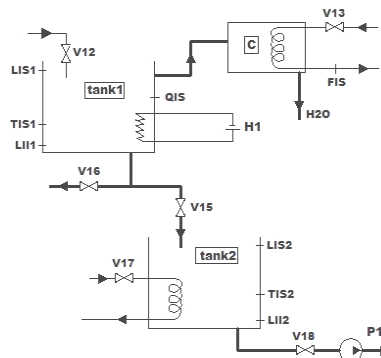


Figure 2: Scheme of the entire evaporator system.

In the normal operation mode, the system works as follows.

The tank1 should be previously filled to its superior level with an aqueous solution by opening valve V12. When the tank1 is full, the heating

system is switch on and also, in simultaneous, the cooling system of the condenser by opening valve V13. When it is formed steam, this condenses in the condenser C. When the concentration desired in the tank1 is reached, there are switch off the heating system and the cooling system of the condenser. Continuously the solution flows from tank1 into tank2, and it must be guaranteed that the tank2 is empty. The transfer of the solution to the tank2 is for a powder-processing operation that is not, here, described. For that powder-processing operation, there is necessary to heat the solution to avoid possible crystallization, and for that there are two approaches: it can heat until the temperature sensor of the tank2 indicates that the desired temperature was reached; or it can heat up for a certain time. Finally, the tank2 is emptied by the pump P1, if the valve V18 be opened.

On the other hand, in the possible failure operation mode, the system works as follows.

A possible failure scenario of the system happens when the cooling fluid flow in the condenser be to low (detected by sensor FIS). This implicates the increase of pressure and temperature in condenser C and tank1, if the heating system keep switch on (solution steam). It is necessary to guarantee that the pressure in the condenser C doesn't exceed a maximum value to avoid its explosion. For that, it should be guaranteed that the heating in the tank1 is switch off before the open of the safety valve (V16).

For this situation of failure operation, it should switch off the resistance H1 the more quickly possible, but tends in account that the solution doesn't crystallize, then that we are before a critical time. To switch off the resistance H1 they are considered two possibilities: through a time after sensor FIS to have detected reduced flow; or through the sensor of temperature TIS1 (due to the pressure and temperature are parameters that are directly related).

There are evidences that should be guaranteed, as for instance that the tanks should never overflow. After the failure situation occurs, all of the valves should be immediately closed.

3.1 Controller Specification

In order to guarantee the desired behavior, the controller specification was developed according to IEC 60848 SFC specification.

Table 1: Input/Output variables of the controller.

Inputs	Outputs
LIS1 – Superior level of the tank1	V12 – Solution entrance of the tank1
LI1 – Inferior level of the tank1	V13 – Cooling of the condenser
QIS – Electrical conductivity of the solution in tank1 (concentration)	V15 – Valve of solution passage of the tank1 for the tank2
TAlarm– Maximum solution temperature in tank1 (sensor TIS1)	V16 – Drain of the tank 1
LIS2 – Superior level of tank2	V17 – Heating of the tank2
LI2 – Inferior level of tank2	V18 – Emptying of the tank2
TIS2 – Solution temperature in tank2	P1 – Emptying pump of the tank2
FIS – Cooling solution flow of the condenser C	H1 – Heating Resistance of the tank1

The input and output variables of the controller which controls the process in closed-loop are presented and described in table 1.

The SFC specification of the controller behavior (normal and failure modes) is presented in figures 3 and 4.

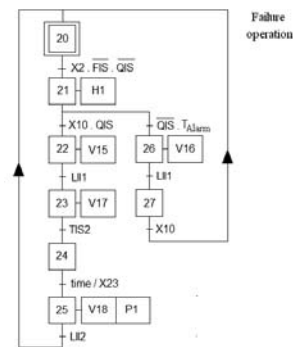


Figure 3: SFC specification of the Controller – Normal Operation Mode.

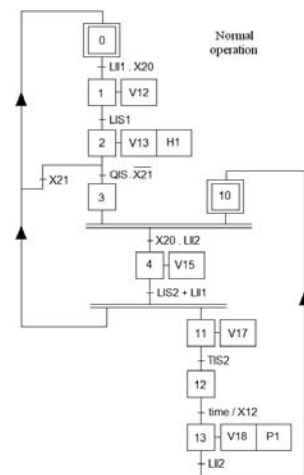


Figure 4: SFC specification of the Controller – Failure Operation Mode.

The controller specification was directly translated to Modelica modeling language, more specifically to the library for hierarchical state machines StateGraph (Otter *et al.* 2005).

4 PLANT MATHEMATICAL MODELLING

The next table presents the mathematical equations that model the system.

The plant modelling has two goals (table 2): first to assure that the controller specification is adequate for the intended system behaviour and, second, to minimize the cycle time for repetitive automation systems processes. In this paper there are discussed the two of them: to be sure that the system behaves as expected – without leading to dangerous situations - and to maximize the productivity of the process that it implicates the maximization of the number of batches.

Due to discrete switching between the two different continuous systems (T1 and T2), which happens not only at the stage transitions, by changing the position of the on/off valves (V15 and V18), but also in stage 2 for boiling water point, this developed model is of hybrid nature. The main required parameters and algebraic equations are presented in detail in the table 2.

The setting of alarm temperature T_{Alarm} is chosen correctly to accomplish the following two opposed very important properties: On the one hand it must be low enough to avoid a dangerous temperature and pressure values, and on the other hand it has to be sufficient high so that temperature T does not fall below a crystallization temperature before liquid level in tank1 (H_1) becomes zero.

5 MODELICA MODEL OF THE SYSTEM

Due to the described potentialities, it was developed a global model of the evaporator system, already presented in the previous sections. The plant was modeled as the controller using the Dymola software and the object-oriented programming language Modelica (Fritzson and Vadim, 1998, Elmqvist and Mattson, 1997). Additionally, to model the controller, it was used the library for hierarchical state machines StateGraph (Otter *et al.* 2005), which are included in the Dymola software.

Table 2: System description (differential and algebraic equations).

Stage 1 Heating while T2 is drained	$(dQ/dt) = Q_{Heat} - Q_{Loss} - Q_{Evap}$ $(dQ/dt) = d(T.c_{p,L} \cdot (m_L + m_V)) / dt$ $\frac{dH_1}{dt} = 0; \frac{dH_2}{dt} = -K_1 \sqrt{H_2}$ $Q_{Loss} = kA \cdot (T - T_e)$ $Q_{Evap} = (dm_V / dt) \cdot \Delta h_{ev}; K_1 = (A_R / A_2) \cdot \sqrt{2g}$ $p = a_0 + a_1 T + a_2 T^2$ (boiling pressure, dissolve substance ignored) $pV_V = (m_V / M_L) R_m T; \Delta h_{ev} = b_1 + b_2 T$ $m_{total} = m_L + m_V = 6 \text{ kg}$ (total mass of fluid), Q_{Heat} (heat supply rate) $V_V = 0.02 \text{ m}^3$ (vapor volume, assumed to be constant), $kA = 24 \text{ W / K}$ (heat loss flow per Kelvin)
Stage 2 Cooling while T2 is drained	$(dQ/dt) = -Q_{Loss} - Q_{Evap}$ $\frac{dH_1}{dt} = 0; \frac{dH_2}{dt} = -K_1 \sqrt{H_2}$ $T < 373 \text{ K}: (dQ/dt) = d(T.c_{p,L} \cdot m_L) / dt$ $Q_{Evap} \cong 0$ $T > 373 \text{ K}, p > 1 \text{ bar}:$ $(dQ/dt) = d(T.c_{p,L} \cdot (m_L + m_V)) / dt;$ $Q_{Evap} = (dm_V / dt) \cdot \Delta h_{ev}; Q_{Loss} = kA \cdot (T - T_e)$ $kA = 22.5 \text{ W / K}$ (heat loss flow per Kelvin) Note: In this stage it will be used the same algebraic equations and parameters as in stage 1.
Stage 3 Cooling while T1 is drained	$(dQ/dt) = -Q_{Loss}$ $\frac{dH_1}{dt} = -K_2 \sqrt{H_1}; \frac{dH_2}{dt} = -K_1 \sqrt{H_1}$ $(dQ/dt) = c_{p,L} \cdot (dT / m_L) / dt; Q_{Loss} = kA \cdot (T - T_e)$ $K_2 = (A_R / A_1) \cdot \sqrt{2g}; m_L = \rho_L H_1 A_1$ $A = A_1 + \pi \cdot D H_1$ $k = 150 \text{ W / K / m}^2$ (heat loss transfer coefficient), $A_1 = 0.03 \text{ m}^2, A_2 = 0.06 \text{ m}^2$ (cross-sectional area T1 and T2)
Variables	state: T (temperature in T1), H_1, H_2 (liquid heights, tanks considered empty when $H_{1/2} \leq 0.0017 \text{ m}$) algebraic: m_L (liquid mass), m_V (vapor mass), Δh_{ev} (evaporation enthalpy), p (pressure), A (heat loss area)
Additional parameters	$A_1 = 0.03 \text{ m}^2, A_2 = 0.06 \text{ m}^2$ (cross-sectional areas of T1 and T2), $A_R = 2 \cdot 10^{-5} \text{ m}^2$ (pipe cross-sectional area) $a_0 = 9.3 \cdot 10^6 \text{ N / m}^2,$ $a_1 = -5.28 \cdot 10^4 \text{ N / m}^2 / \text{K}^2,$ $a_2 = 75.4 \text{ N / m}^2 / \text{K}^2$ (a_0, a_1, a_2 pressure constants), $b_1 = 3.294 \cdot 10^6 \text{ J / kg}, b_2 = -2.78 \cdot 10^3 \text{ J / kg / K}$ (enthalpy constant), $c_{p,L} = 4220 \text{ J / kg / K}$ (liquid heat capacity), $D = 0.2 \text{ m}$ (diameter of T1), $g = 9.81 \text{ m / s}^2$ (gravity constant), $M_L = 0.018 \text{ kg / mol}$ (molecular weight of liquid), $\rho_L = 970 \text{ kg / m}^3$ (liquid density), $R_m = 8.314 \text{ J / kg / mol}$ (molecular gas constant), $T_e = 283 \text{ K}$ (environment temperature)

Related with the plant part, it was modeled the filling source, the tank1 and tank2, the heater (H1), the condenser and the valves. For that, it were used the parameters and algebraic equations presented in the table 2.

Figure 5 shows the global modelica model of the system, being highlighted the two main parts, the physical part (plant) on the left, and the controller on the right. On the other hand, the controller model was developed according the SFC specifications (see figures 3 and 4).

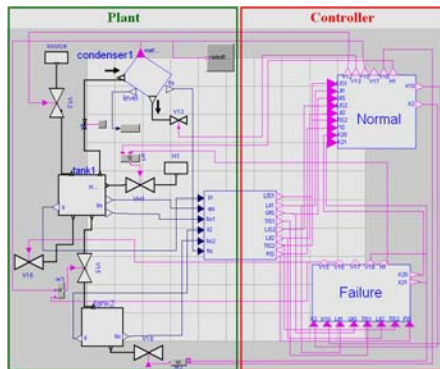


Figure 5: Global modelica model of the evaporator system.

Also, due to the reason of it being specified a discrete controller to control the hybrid plant, it was necessary to implement an appropriate interface, that translate the analogue outputs signals of the plant (tanks levels, temperatures, concentration,...) digital signals, that can be used as inputs of the discrete controller.

6 SIMULATION RESULTS

In this section, there are presented results of simulations that were accomplished with the purpose of studying the dynamical behavior of the hybrid models described in the previous sections in order to maximize the productivity of the evaporator process, in terms, of their energy efficiency and batches times.

Moreover, these simulations can be seen as a “system preliminary analysis” to check if the system behaves in agreement to a given specification for a particular case, like as, a given a initial state of the process and a given control program. However, it must to be enhanced that this is not verification in the strict sense, since it relies on the appropriate selection of the considered cases.

In order to perform the hybrid model simulation with different heating power’s it was necessary to define the parameters, start and stop time of the simulation, the interval output length or number of output intervals and the integration algorithm. In the present work, in all simulations performed, the Dass algorithm (Basu *et al.* 2006) with 10000 output intervals was used.

The first simulations performed was devoted to verify if the SFC of the controller system (see figures 3 and 4) modeled with Modelica language with the library for hierarchical state machines StateGraph simulated correctly the evaporator system.

Figures 6 and 7 show the results of the first two simulations, respectively, relating to the normal operation and failure operation modes for the level tanks. The failure mode it is consequence of the occurrence of the condenser malfunction during the production cycle that it originates that the solution temperature in the tank1 reach the alarm temperature pre-defined (390K).

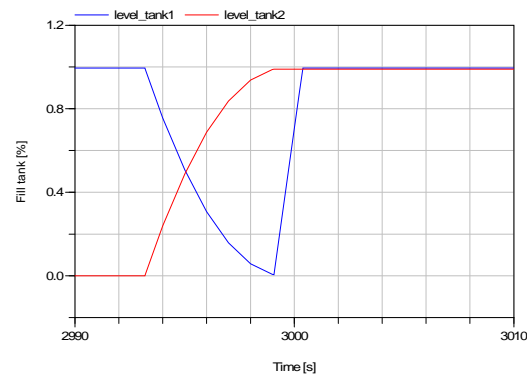


Figure 6: Level tanks in function of time in normal operation mode of the evaporator system.

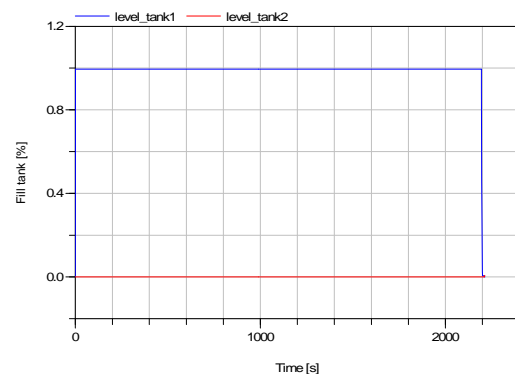


Figure 7: Level tanks in function of time in failure operation mode of the evaporator system.

Observing Figure 6 it can be concluded that the normal operation mode is properly simulated by the developed program, since the two main properties that are important to prove are confirmed, for instance, the drainage of the solution present in the tank 1 only to happen when the tank2 is empty and also the filling of the tank1 to happen soon after this to be empty. On the other hand, observing figure 7, it can be also concluded that the failure operation mode is properly simulated, given that is proven that the tank1 is drained through the safety valve (V16 – see figure 2) because it is seen that the tank2 remains empty.

After being concluded that the normal and failure operation behavior is properly simulated by the proposed program they were performed other simulations in order to obtain the relationship between several physical plant parameters that can obtain the best ratio between the number of batches and the supply energy costs.

This manner, among of several physical variables of the process (see table 2) it was chosen the heat supply rate (Q_{Heat}) because it is the most relevant variable, that determine the rate of the steam formation (this condenses in the condenser C) and correspondingly, the time in that the solution present in the evaporator (tank1) is prepared to be drained (desired concentration reached).

The solution concentration (C) is obtained by the following equation:

$$C = (C_0 \cdot m_L) / (m_L - m_V) \tag{1}$$

Where, C_0 is the initial concentration, m_L is the liquid mass and m_V is the vapour mass. In addition, in all of the performed simulations, it was assumed concentration values of 0.01000 and 0.01003, respectively, initial and final.

Figures 8 and 9 illustrate the behavior of the model given in the table 2 for heat supply rate (Q_{Heat}) of 3000W, respectively for the vapour mass and concentration.

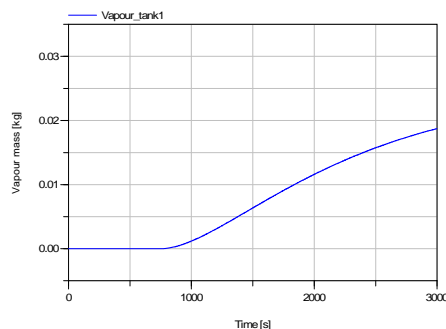


Figure 8: Vapour mass in function of time with a heat supply rate of 3000W.

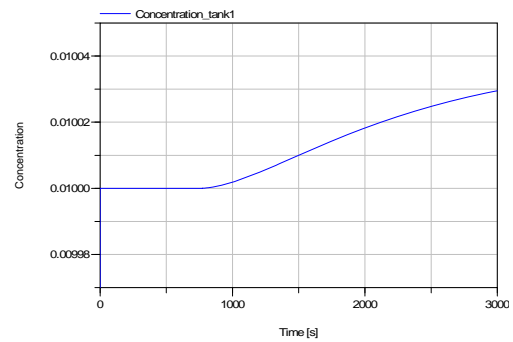


Figure 9: Concentration in function of time with a heat supply rate of 3000W.

In a general way, the results presented in the figures 8 and 9 allow to conclude that the concentration behavior is properly simulated by the proposed program.

In particular, analyzing figure 8 it can be stated that the boiling water point (373K) it is reached after having elapsed about 800s and after this time the vapour mass increases continually as it was foreseen with the increase of the temperature.

On other hand, observing figure 9, it can be verified that the time in that the solution present in the tank1 reaches the final concentration (0.01003), and this way prepared to be drained to tank2, is about 3000s.

In order to be possible to generalize the batches optimization, that it implicates the productivity maximization of the evaporator system, it is essential to know the optimized relation between the heat supply rate and the time for the solution reaches the desired concentration in the tank1 (evaporator).

Figure 10 illustrates the time for the solution reaches the desired final concentration in function of heat supply rate, as example, from 3000 to 10000W.

Analyzing figure 10, it can be concluded that the increase of the heat supply rate originates a very significant decrease on the required time for the solution reaches the final concentration. It can be highlighted that the more accentuated time reductions happens in the interval from 3000 to 20000W.

This manner, in agreement with the simulations results presented, it can be concluded that the heat supply rate of 20000W, could be the most appropriate to obtain the best optimization between the number of batches and the supply energy costs, considering the values of the physical variables of the evaporator system presented in table 2.

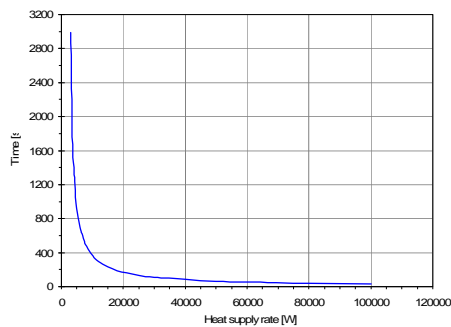


Figure 10: Time for the solution present in the tank1 reaches the final concentration in function of heat supply rate.

7 CONCLUSIONS AND FUTURE WORK

The simulation used to evaluate the controller and plant behavior has been developed and proposed in this paper.

The present research proved to be successful using the Modelica programming Language to obtain a plant model and using it, in a closed-loop behavior, with the controller model.

Some parameters and functional aspects of the system have been simulated in order to define a set of values of different variables that make the system dependable and safe avoiding dangerous situations, and more efficient, when studied some critical plant behavior parameters.

The study of critical plant behavior parameters (like presented in this paper) can be performed using Modelica in order to obtain simulation models of complex systems.

As future work the authors believe that, with auxiliary calculations, it will be possible, using simulation strategies, to define optimal values for the different variables, in order to obtain, by one hand, a safe system behavior and, by other hand, to optimize the time cycle of Automation repetitive systems taking into account the critical steps of their functioning.

ACKNOWLEDGEMENTS

This research project is carried out in the context of the SCAPS Project supported by FCT, the Portuguese Foundation for Science and Technology, and FEDER, the European regional development fund, under contract POCI/EME/61425/2004 that

deals with safety control of automated production systems.

REFERENCES

- Aström, K. J., Elmquist, H. & Mattsson, S. E., 1998. *Evolution of continuous time modeling and simulation*, in 'Proceedings of the 12th European Simulation Multiconference', Manchester, UK, pp. 9–18.
- Basu S., Pollack R., Roy M., 2006. *Algorithms in Real Algebraic Geometry*. In Springer (Eds), Algorithms and Computation in Mathematics, (10), 2nd edition.
- Dymola software, 2010. Available in <http://www.3ds.com/products/catia/portfolio/dymola> (consulted in Mar 05th, 2010).
- Elmqvist E., Mattson S., 1997. In *ESS'97, An Introduction to the Physical Modelling Language Modelica. Proceedings of the 9th European Simulation Symposium*. Passau, Germany.
- Fritzon, P., Vadim E., 1998. In *ECOOP'98: Modelica, a general object-oriented language for continuous and discrete event system modeling and simulation. 12th European Conference on Object-Oriented Programming*. Brussels, Belgium.
- Karayanakis, N. M., 1995. *Advanced System Modelling and Simulation with Block Diagram Languages*, CRC Press, Inc.
- Kowalewski S., Stursberg O. and Bauer N., 2001. *An Experimental Batch Plant as a Test Case for the Verification of Hybrid Systems*. European Journal of Control. vol. 7, n_4, pp. 400-415.
- Matlab, 2010. Matlab Available in <http://www.Mathworks.com> (consulted in March 05th, 2010).
- MATRIX, 2010. MATRIX Available in <http://www.ni.com/matrix> (consulted in March 05th, 2010).
- MGA Software, 1996. ACSL Graphic Modeller - Version 4.1, MGA Software.
- Mosterman P. J., 2002. *HYBRISIM - a modeling and simulation environment for hybrid bond graphs. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*. Vol. 216, N 1, pp 35-46.
- Otter M., Årzén K., Dressler I., 2005. StateGraph - A Modelica Library for Hierarchical State Machines. *Modelica 2005 Proceedings*, 2005.

POSTERS

FUZZY FREQUENCY RESPONSE FOR UNCERTAIN DYNAMIC SYSTEMS

Carlos Cesar Teixeira Ferreira and Ginalber Luiz de Oliveira Serra
Federal Institute of Education, Science and Technology of Maranhão (IFMA)
Av. Getúlio Vargas, 04, Monte Castelo, 65025-001, São Luís, MA, Brazil
{ccteixeira, ginalber}@ifma.edu.br

Keywords: Takagi-Sugeno fuzzy control, Uncertain dynamic systems, Frequency response analysis.

Abstract: This paper focuses on the Fuzzy Frequency Response: Definition and Analysis for Uncertain Dynamic Systems. In terms of transfer function, the uncertain dynamic system is partitioned into several linear sub-models and it is organized into Takagi-Sugeno (TS) fuzzy structure. The main contribution of this approach is demonstrated, from a *Theorem*, that fuzzy frequency response is a boundary in the magnitude and phase Bode plots. Low and high frequency analysis of fuzzy dynamic model is obtained by varying the frequency ω from zero to infinity.

1 INTRODUCTION

The design of control systems is currently driven by a large number of requirements posed by increasing competition, environmental requirements, energy and material costs, the demand for robust and fault-tolerant systems. These considerations introduce extra needs for effective process control techniques. In this context, the analysis and synthesis of compensators are completely related to each other. In the analysis, the characteristics or dynamic behaviour of the control system are determined. In the design, the compensators are obtained to attend the desired characteristics of the control system from certain performance criteria. Generally, these criteria may involve disturbance rejection, steady-state errors, transient response characteristics and sensitivity to parameter changes in the plant.

Test input signals is one way to analyse the dynamic behaviour of real world system. Many test signals are available, but a simple and useful signal is the sinusoidal wave form because the system output with a sinusoidal wave input is also a sinusoidal wave, but with a different amplitude and phase for a given frequency. This frequency response analysis describes how a dynamic system responds to sinusoidal inputs in a range of frequencies and has been widely used in academy, industry and considered essential for robust control theory (Serra and Ferreira, 2010).

The frequency response methods were developed during the period 1930 – 1940 by Harry

Nyquist (1889 – 1976) (Nyquist, 1932), Hendrik Bode (1905 – 1982) (Bode, 1940), Nathaniel B. Nichols (1914 – 1997) (James et al., 1947) and many others. Since, frequency response methods are among the most useful techniques and available to analyse and synthesise the compensators. In (Jr, 1973), the U.S. Navy obtains frequency responses for aircraft by applying sinusoidal inputs to the autopilots and measuring the resulting position of the aircraft while the aircraft is in flight. In (Lascu et al., 2009), four current controllers for selective harmonic compensation in parallel Active Power Filters (APFs) have been compared analytically in terms of frequency response characteristics and maximum operational frequency. Most real systems, such as circuit components (inductor, resistor, operational amplifier, etc.) are often formulated using differential/integral equations with uncertain parameters (Kolev, 1993). The uncertain about the systems arises from aging, temperature variations, etc. These variations do not follow any of the known probability distributions and are most often quantified in terms of boundaries. The classical methods of frequency response do not explore these boundaries for uncertain dynamic systems. To overcome this limitation, this paper proposes the definition of Fuzzy Frequency Response (FFR) and its application for analysis of uncertain dynamic systems.

2 FORMULATION PROBLEM

This section presents some essentials concepts for the formulation and development of this paper *Fuzzy Frequency Response for Uncertain Dynamic Systems*.

2.1 Uncertain Dynamic Systems

In terms of transfer function, the general form of an uncertain dynamic systems is given by Eq. 1, as depicted in Fig. 1.

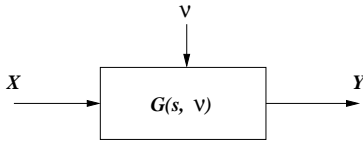


Figure 1: TS fuzzy model.

$$G(s, v) = \frac{Y(s, v)}{X(s)} = \frac{b_\alpha(v)s^\alpha + b_{\alpha-1}(v)s^{\alpha-1} + \dots + b_\alpha(v)s^\alpha + b_1(v)s + b_0(v)}{s^\beta + a_{\beta-1}(v)s^{\beta-1} + \dots + a_1(v)s + a_0(v)} \quad (1)$$

where: $X(s)$ and $Y(s, v)$ represents the input and the output of uncertain dynamic systems; $a_*(v)$ and $b_*(v)$ are the varying parameters; $v(t)$ is the time varying scheduling variable; s is the Laplace operator; α and β are the orders of the numerator and denominator of the transfer function, respectively (with $\beta \geq \alpha$). The scheduling variable v belongs to a compact set $v \in V$, with its variation limited by $|\dot{v}| \leq d^{\max}$, with $d^{\max} \geq 0$. This formulation is very efficient and the fuzzy frequency response of (1) can be used for stability analysis and robust control design.

2.2 Takagi-Sugeno Fuzzy Dynamic Model

The inference system TS, originally proposed in (Takagi and Sugeno, 1985), presents in the consequent a dynamic functional expression of the linguistic variables of the antecedent. The i $\left| \begin{matrix} i=1,2,\dots,l \end{matrix} \right.$ -th rule, where l is the rules numbers, is given by

$$\begin{aligned} & \text{Rule}^{(i)} : \\ & \text{IF } \tilde{x}_1 \text{ is } F_{\{1,2,\dots,p_{\tilde{x}_1}\}|\tilde{x}_1}^i \text{ AND } \dots \text{ AND } \tilde{x}_n \text{ is } F_{\{1,2,\dots,p_{\tilde{x}_n}\}|\tilde{x}_n}^i \\ & \text{THEN } y_i = f_i(\tilde{\mathbf{x}}) \end{aligned} \quad (2)$$

where the total number of rules is $l = p_{\tilde{x}_1} \times \dots \times p_{\tilde{x}_n}$. The vector $\tilde{\mathbf{x}} = [\tilde{x}_1, \dots, \tilde{x}_n]^T \in \mathfrak{R}^n$ containing the linguistics variables of antecedent, where T

represents the operator for transpose matrix. Each linguistic variable has its own discourse universe $\mathcal{U}_{\tilde{x}_1}, \dots, \mathcal{U}_{\tilde{x}_n}$, partitioned by fuzzy sets representing its linguistics terms, respectively. In i -th rule, the variable $\tilde{x}_{\{1,2,\dots,n\}}$ belongs to the fuzzy set $F_{\{\tilde{x}_1, \dots, \tilde{x}_n\}}^i$ with a membership degree $\mu_{F_{\{\tilde{x}_1, \dots, \tilde{x}_n\}}^i}$ defined by a membership function $\mu_{F_{\{\tilde{x}_1, \dots, \tilde{x}_n\}}^i} : \mathfrak{R} \rightarrow [0, 1]$, with $\mu_{F_{\{\tilde{x}_1, \dots, \tilde{x}_n\}}^i} \in \{\mu_{F_{1|\{\tilde{x}_1, \dots, \tilde{x}_n\}}^1}, \mu_{F_{2|\{\tilde{x}_1, \dots, \tilde{x}_n\}}^1}, \dots, \mu_{F_{p|\{\tilde{x}_1, \dots, \tilde{x}_n\}}^1}\}$, where $p_{\{\tilde{x}_1, \dots, \tilde{x}_n\}}$ is the partitions number of the discourse universe associated with the linguistic variable $\tilde{x}_1, \dots, \tilde{x}_n$. The output of the TS fuzzy dynamic model is a convex combination of the dynamic functional expressions of consequent $f_i(\tilde{\mathbf{x}})$, without lost of generality for the bidimensional case, as illustrated in Fig. 2, given by Eq. 3.

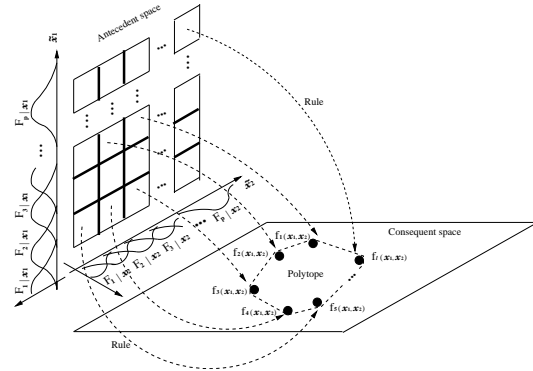


Figure 2: Fuzzy dynamic model: A TS model can be regarded as a mapping from the antecedent space to the space of the consequent parameters one.

$$y(\tilde{\mathbf{x}}, \gamma) = \sum_{i=1}^l \gamma_i(\tilde{\mathbf{x}}) f_i(\tilde{\mathbf{x}}) \quad (3)$$

where γ is the scheduling variable of the TS fuzzy dynamic model. It can be observed that the TS fuzzy dynamic system, which represents any uncertain dynamic model, may be considered as a class of systems where $\gamma_i(\tilde{\mathbf{x}})$ denotes a decomposition of linguistic variables $[\tilde{x}_1, \dots, \tilde{x}_n]^T \in \mathfrak{R}^n$ for a polytopic geometric region in the consequent space from the functional expressions $f_i(\tilde{\mathbf{x}})$.

3 FUZZY FREQUENCY RESPONSE (FFR): DEFINITION

This section will present how a TS fuzzy model of an uncertain dynamic system responds to sinusoidal inputs, which in this paper is proposed as the definition of fuzzy frequency response. The response of a TS fuzzy model to a sinusoidal input of frequency ω_1

in both amplitude and phase, is given by the transfer function evaluated at $s = j\omega_1$, as illustrated in Fig. 3.

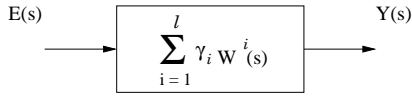


Figure 3: TS fuzzy transfer function.

For this TS fuzzy model,

$$Y(s) = \left[\sum_{i=1}^l \gamma_i W^i(s) \right] E(s) \quad (4)$$

Consider $\sum_{i=1}^l \gamma_i W^i(j\omega)$ a complex number for a given ω , as

$$\begin{aligned} \sum_{i=1}^l \gamma_i W^i(j\omega) &= \\ &= \left| \sum_{i=1}^l \gamma_i W^i(j\omega) \right| e^{j\phi(\omega)} = \left| \sum_{i=1}^l \gamma_i W^i(j\omega) \right| \angle \phi(\omega) = \\ &= \left| \sum_{i=1}^l \gamma_i W^i(j\omega) \right| \angle \arctan \left[\sum_{i=1}^l \gamma_i W^i(j\omega) \right] \end{aligned} \quad (5)$$

Then, for the case that the input signal $e(t)$ is sinusoidal, that is,

$$e(t) = A \sin \omega_1 t \quad (6)$$

the output signal $y_{ss}(t)$, in the steady state, is given by

$$y_{ss}(t) = A \left| \sum_{i=1}^l \gamma_i W^i(j\omega) \right| \sin [\omega_1 t + \phi(\omega_1)] \quad (7)$$

As result of the fuzzy frequency response definition, it is proposed the following theorem:

Theorem 3.1. Fuzzy frequency response is a region in the frequency domain, defined by the consequent sub-models and from the operating region of the antecedent space.

Proof. Considering that the parameter $v(t)$ is uncertain and can be represented by linguistic terms, once known its discourse universe, as shown in Fig. 4, the activation degrees, $h_i(\tilde{v})|^{i=1,2,\dots,l}$, are also uncertain, since it depends of the dynamic system:

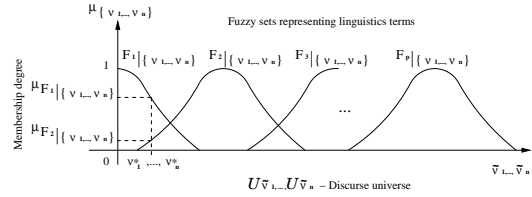


Figure 4: Functional description of the linguistic variables: linguistic terms, discourse universes and membership degrees.

$$h_i(\tilde{v}) = \mu_{F_{\tilde{v}_1}^i}^i \star \mu_{F_{\tilde{v}_2}^i}^i \star \dots \star \mu_{F_{\tilde{v}_n}^i}^i \quad (8)$$

where $\tilde{v}_{\{1,2,\dots,n\}}^i \in \mathcal{U}_{\tilde{v}_{\{1,2,\dots,n\}}}$, respectively, and \star is a fuzzy logic operator.

So, the normalized activation degrees $\gamma_i(\tilde{v})|^{i=1,2,\dots,l}$, are also uncertain, as shown in:

$$\gamma_i(\tilde{v}) = \frac{h_i(\tilde{v})}{\sum_{r=1}^l h_r(\tilde{v})} \quad (9)$$

This normalization implies

$$\sum_{k=1}^l \gamma_k(\tilde{v}) = 1 \quad (10)$$

The output of the TS fuzzy model is a weighted sum of the consequent functional expression, e.g., a linear convex combination of the local functions $f_i(\tilde{v})$, and is given by

$$y(\tilde{v}) = \sum_{i=1}^l \gamma_i(\tilde{v}) f_i(\tilde{v}) \quad (11)$$

Let $F(\tilde{v})$ a vectorial space of transfer functions with degree $\leq l$ and $f^1(s), f^2(s), \dots, f^l(s)$ transfer functions which belongs to this vectorial space. A transfer function $f(s) \in F(\tilde{v})$ must be a linear convex combination of the vectors $f^1(s), f^2(s), \dots, f^l(s)$. So

$$f(s) = \gamma_1 f^1(s) + \gamma_2 f^2(s) + \dots + \gamma_l f^l(s) \quad (12)$$

$$f(s) = \sum_{i=1}^l \gamma_i(\tilde{v}) f_i(\tilde{v}) \quad (13)$$

The TS fuzzy model must attend the polytope property. So, the sum of the normalized activation degree must be equal to 1, as shown in Eq (10). To satisfy this property, each rule must be singly activated. This condition is called boundary condition. In this way, the following results are obtained:

If just the rule 1 is activated, it has ($\gamma_1 = 1, \gamma_2 = 0, \gamma_3 = 0, \dots, \gamma_l = 0$). Hence,

$$f(s) = 1f^1(s) + 0f^2(s) + \dots + 0f^l(s) = f^1(s) \quad (14)$$

From (5), it has

$$f(j\omega) = |f^1(j\omega)| \angle f^1(j\omega) \quad (15)$$

If just the rule 2 is activated, it has ($\gamma_1 = 0, \gamma_2 = 1, \gamma_3 = 0, \dots, \gamma_l = 0$). Hence,

$$f(s) = 0f^1(s) + 1f^2(s) + \dots + 0f^l(s) = f^2(s) \quad (16)$$

From (5), it has

$$f(j\omega) = |f^2(j\omega)| \angle f^2(j\omega) \quad (17)$$

If just the rule l is activated, it has ($\gamma_1 = 0, \gamma_2 = 0, \gamma_3 = 0, \dots, \gamma_l = 1$). Hence,

$$f(s) = 0f^1(s) + 0f^2(s) + \dots + 1f^l(s) = f^l(s) \quad (18)$$

From (5), it has

$$f(j\omega) = |f^l(j\omega)| \angle f^l(j\omega) \quad (19)$$

Note that $|f^1(j\omega)| \angle f^1(j\omega)$ and $|f^l(j\omega)| \angle f^l(j\omega)$ define a boundary region. Under such circumstances, it seems plausible that the fuzzy frequency response for uncertain dynamic systems converges to a boundary in the frequency response, as shown in Fig.5.

4 CONCLUSIONS

The Fuzzy Frequency Response: Definition and Analysis for Uncertain Dynamic Systems is proposed in this paper. It was shown that the fuzzy frequency response is a region in the frequency domain, defined by the consequent linear sub-models $G^i(s)$, from operating regions of the uncertain dynamic system, according to the proposed *Theorem 3.1*. This formulation is very efficient and can be used for robust stability analysis and control design for uncertain dynamic systems.

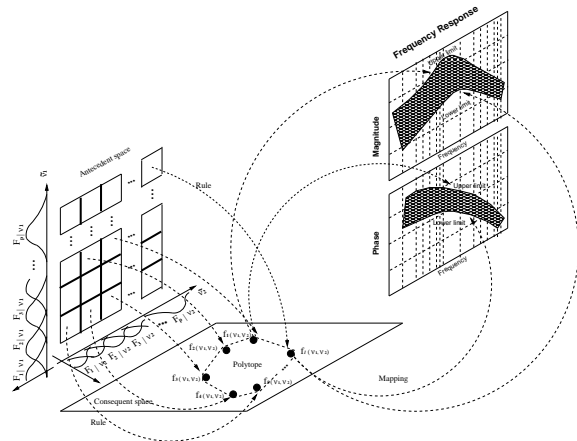


Figure 5: Fuzzy frequency response: mapping from the consequent space to the region in the frequency domain.

ACKNOWLEDGEMENTS

The authors wish to express their gratitude for FAPEMA and CAPES by support of this research.

REFERENCES

- Bode, H. W. (1940). Feedback amplifier design. *Bell Systems Technical Journal*, 19:42.
- James, H. M., Nichols, N. B., and Phillips, R. S. (1947). *Theory of servomechanisms*. McGraw-Hill, MIT Radiation Laboratory Series. New York.
- Jr, A. P. S. (1973). Determination of aircraft response characteristics in approach/landing configuration for microwave landing system program. In *Report FT-61R-73, Naval Air Test Center*. Patuxent River, MD.
- Kolev, L. V. (1993). Interval methods for circuit analysis. *Singapore: World Scientific*.
- Lascu, C., Asiminoaei, L., Boldea, I., and Blaabjerg, F. (2009). Frequency response analysis of current controllers for selective harmonic compensation in active power filters. *IEEE Transactions on Industrial Electronics*, 56(2).
- Nyquist, H. (1932). Regeneration theory. *Bell Systems Technical Journal*.
- Serra, G. L. O. and Ferreira, C. C. T. (2010). Fuzzy frequency response: Definition and analysis for nonlinear dynamic systems. *IEEE International Symposium on Industrial Electronics*.
- Takagi, T. and Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst. Man. Cyber*, 15:116–132.

HARDWARE-SOFTWARE CODESIGN OF FUZZY CONTROL SYSTEMS USING FPGAS

E. del Toro

*Centro de Investigaciones de Microelectrónica (CIME), CUJAE, Ciudad de La Habana, Cuba
ernesto@imse-cnm.csic.es*

S. Sánchez-Solano

*Instituto de Microelectrónica de Sevilla (CNM-CSIC), Sevilla, Spain
santiago@imse-cnm.csic.es*

M. Brox

*Dpto. Arquitectura de Computadores, Electrónica y Tecnología Electrónica, Univ. Córdoba, Spain
mbrox@uco.es*

A. J. Cabrera

*Dpto. Automática y Computación, CUJAE, Ciudad de La Habana, Cuba
alex@electronica.cujae.edu.cu*

Keywords: Fuzzy Control, Hardware/Software Codesign, FPGA.

Abstract: This paper describes a hardware/software codesign strategy for fuzzy control systems implementation using FPGAs. The main contribution of the paper consists of a methodology for joint development of hardware and software components intended for rapid and verifiable design of a fuzzy control system. The design flow combines specific tools for fuzzy inference systems included in the *XFuzzy* environment, simulation and modelling tools from Matlab and FPGA synthesis, and implementation tools provided by Xilinx. The advantages of this proposal are described in section 4 as it is used for the control system development of an autonomous vehicle.

1 INTRODUCTION

Fuzzy logic provides a mathematical framework to deal with the uncertainty and the imprecision typical of the human reasoning system. One of its main characteristics is the capability to describe the behaviour of a complex system in a linguistic way (Zadeh, 1973). Unlike classical logic systems, fuzzy logic aims to model approximated reasoning modes that play a significant role in the human ability to make rational decision without using precise mathematical models. These advantages have led to an increase in the number of applications using fuzzy logic controller (Ross, 2004).

A great number of different design proposals, which range from software implementation to complete hardware design, have been reported in the last year (Baturone et al., 2000). The level of

complexity attained by many of the current applications of control systems requires designing the fuzzy inference modules (FIM) as components to be included in a bigger system that, not only will be able to apply the control responses, but also to interface to other systems, reconfigure itself to different states, and perform other tasks not related to the fuzzy inference process. In these systems, common tasks and configuration may be realized by the software part using a general purpose processor, while time consuming functions must be implemented by means of specialized hardware (Cabrera et al., 2004).

The progress in integrated circuits manufacturing technologies allows the integration of complex control systems on a single chip. Also, the resources available in current FPGA families can be used to implement a system on a programmable chip (SoPC). However, to benefit from these

technological advances, new design methodologies and powerful CAD tools must be developed to cut down the development cycle of new products and make them more competitive in market terms.

A fuzzy control system design methodology is described in this paper. The codesign strategy and the basic components of the control systems are introduced in Section 2. In Section 3, the design flow and the tools are described. An application of the proposed methodology is explained in Section 4. Finally, the main contributions and future goals are resumed in Section 5.

2 CONTROL SYSTEMS CODESIGN MODULES

The proposed HW/SW codesign methodology for development of a fuzzy control system as a SoPC combines the use of a general purpose processor, available as IP-module for FPGA, connected to specific fuzzy IP-modules that allow fuzzy inference acceleration.

The processor used is MicroBlaze, which is a 32-bit RISC processor soft core optimized for implementation in Xilinx FPGAs. The system architecture of MicroBlaze consists of several buses that allows using multiple interfaces to connect peripherals.

The main characteristics of the fuzzy module used in this design are the efficient use of resources, low power and high speed. In order to accomplish these characteristics it is important to remark the use of simplified defuzzification methods, the limited overlap degree of input membership functions and the implementation of a processing strategy that evaluates only the active rules (Sánchez-Solano et al., 2007).

3 DESIGN FLOW TOOLS

The proposed design flow combines the use of specific tools for development of fuzzy systems from the *XFuzzy* environment, modelling and simulation using Matlab, and Xilinx EDK for synthesis and implementation in FPGAs. According to the proposed methodology, the development of a fuzzy control system will be implemented at different stages which are described in next sections.

3.1 FIMs Design using Xfuzzy

The *Xfuzzy* environment has been developed to ease the design of fuzzy systems by starting from linguistic and/or numerical knowledge to final implementing them as hardware and/or software components. It provides a wide set of new featured tools which offer Graphical User Interfaces to ease the design flow at the stages of description, verification and synthesis. It can be also used for extracting fuzzy rules from numerical data and includes tuning and simplification facilities (López et al., 1998).

The first stage of the aforementioned methodology aims at functional description and verification of the fuzzy inference modules. The FIMs may be described in *Xfuzzy* using a hierarchical architecture that combines fuzzy modules (for implementation of fuzzy rules bases) and crisp modules (to perform arithmetic and logic functions).

Knowledge bases may be generated directly via *xfedit* or using identification and supervised learning tools, like *xfdm* and *xfsl*, with training data. *xfplot* may be used for functional verification. In addition, a closed loop simulation may be done with *xfsim*, using the fuzzy module in connection with a high-level model of the plant.

Once concluded the specification stage, a tool named *xfsg* is used to perform hardware synthesis. This tool generates the required files for the next stage.

3.2 Synthesis and Verification using SysGen

Using the SysGen library (Xilinx, 2008b), *XFuzzyLib* is generated as a specific library that provides basic modules for implementation of fuzzy controllers. *XFuzzyLib* library contains basic building blocks and other module descriptions including different connectives and defuzzification methods. See Figure 1 for description of an inference module.

Automatic translation between the fuzzy inference description and the Simulink module is made using the files generated by the above mentioned *xfsg*. These files are a Simulink module describing the fuzzy system and a Matlab file that contains the definition of size and functionality of FIM components.

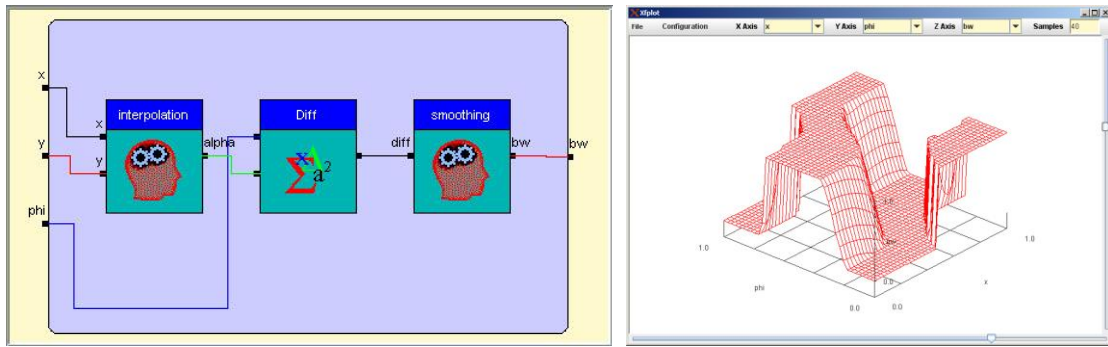


Figure 1: Description of a fuzzy inference module in *Xfuzzy* (left) and associated control surface (right).

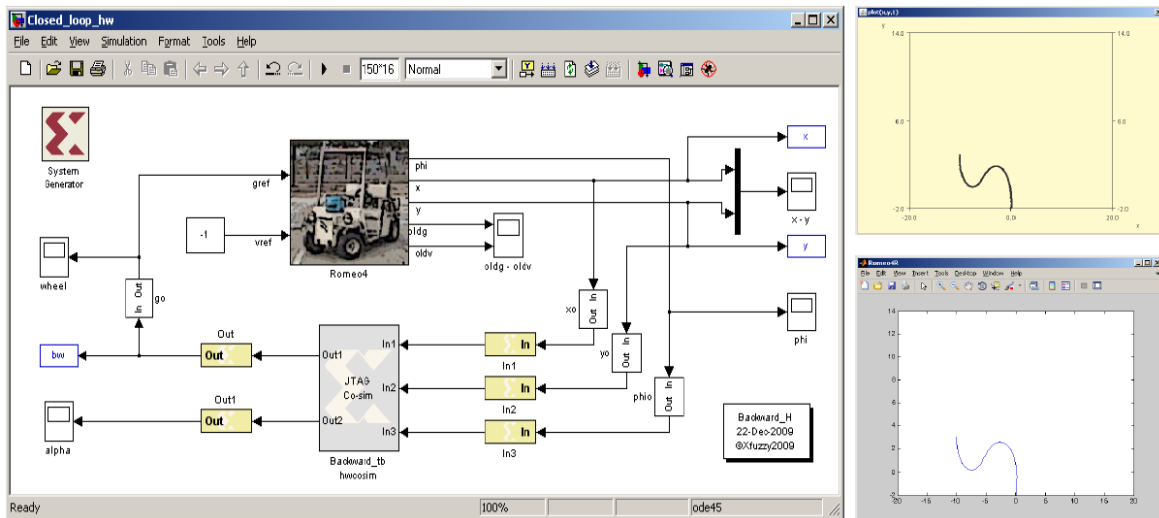


Figure 2: Left: Hardware/software cosimulation using the FPGA implementation of the fuzzy controller in a closed loop with a plant model. Top right: *Xfuzzy* simulation results of parking maneuvers, Bottom right: Simulink simulation.

Design correctness can be verified at this stage by means of the facilities provided by Matlab. The use of a System Generator Block allows hardware synthesis. Also, it is possible to perform functional verification in closed loop through hardware-software cosimulation as shown in Figure 2.

3.3 IP Module Construction and Integration with MicroBlaze

SysGen has options to connect hardware design with MicroBlaze implementation in a smooth way. Basically this consists in defining *input* and *output* registers so they can be addressed by MicroBlaze in various forms, see (Xilinx, 2008a).

The import process in EDK adds interface (glue) logic according to the selected BUS for connection

as well as basic drivers for software communication.

3.4 Control System Implementation

MicroBlaze hardware synthesis connected with the fuzzy controller and with other IP modules is possible thanks to Xilinx XPS tool. According to the design needs and constraints, the designer follows basic steps in order to correctly implement the whole system. Numerous options can be used, including that where the system (MicroBlaze processor and peripherals including the fuzzy controller) can be taken again to Simulink in order to verify correctness of implementation and perform a much more complex simulation.

4 CONTROL SYSTEM DEVELOPMENT OF AN AUTONOMOUS VEHICLE

Parking of autonomous vehicles in a constrained space is a typical control problem in robotics (Li et al., 2003). Starting from any given position and orientation (x, y, Φ) , the autonomous mobile robot must drive forward and backward (as required) at speed v and with a wheel curvature γ in order to always arrive backward at target position $(0, 0, 0)$.

The above methodology has been applied to the realization of a fuzzy control system for autonomous parking of an electric vehicle. The used mobile robot has been an autonomous electric vehicle called Romeo-4R. Romeo-4R is a four-wheeled car with standard Ackerman steering, DC traction and steering electrical motors. A digital signal processor (DSP) TMS-320LF provides support for motor control (encoder inputs and PWM outputs), A/D conversion, and communication links through serial ports, thus easing the low level control of the vehicle. The DSP acquires information from sensors (a gyroscope and encoders) and processes it by using a kinematical model usually employed for car-like robots in order to resolve the actual position (x, y) and orientation (Φ) (Cuesta et al., 2004).

The state of the vehicle is transmitted every 50 ms, thus determining the duration of the control cycle. The fuzzy high-level controller performs the parking control strategy and sends back to the DSP the new required values of speed and wheel curvature, so that the DSP controls the traction and direction motors. This hierarchical control structure allows developing different control strategies in the high-level controller and frees it from the low-level control task of Romeo-4R.

Once known the values of the current state (x, y, Φ, v, γ) of Romeo-4R, the DSP transmits them to the FPGA containing the fuzzy controller through a RS-232 serial interface and using a specific communication protocol (which is also implemented by the program running in the MicroBlaze processor).

In order to give physical support to the development platform, a Xilinx University Program Virtex2-Pro Development System Board has been employed. This board allows cosimulation to be carried out using Matlab.

Figure 2 (right) shows simulation results illustrating the trajectories of parking maneuvers.

5 CONCLUSIONS

A realization strategy for the development of hybrid HW/SW embedded fuzzy controllers on FPGA devices has been described. The design flow combines specific tools for development, simulation, synthesis and implementation using FPGAs. The main contribution of this paper is a methodology for the joint construction of hardware and software components in every stage of design. The proposed methodology is applied to solve a classic robotic problem.

ACKNOWLEDGEMENTS

Project TEC2008-04920 financed by “Ministerio de Ciencia e Innovación” and P08-TIC-03674 by “Junta de Andalucía”. E. del Toro is a MAEC-AECID scholarship PhD. student.

REFERENCES

- Baturone, I., Barriga, A., Sánchez-Solano, S., Jiménez, C.J., and López, D., 2000. *Microelectronic Design of Fuzzy Logic-Based Systems*. CRC Press.
- Cabrera, A., Sánchez-Solano, S., Brox, P., Barriga, A., and Senhadji, R., 2004. *Hardware/software codesign of configurable fuzzy control systems*. Applied Soft Computing, 4, 271-285.
- Cuesta, F., Gómez-Bravo, F., and Ollero, A., 2004. *Parking maneuvers of industrial-like electrical vehicles with and without trailer*, IEEE Trans. on Industrial Electronics, 51, 2, 257-269.
- Li, T. H. S., Shih-Jie, C., and Yi-Xiang, C. 2003. *Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot*. IEEE Trans. Ind. Electron., 50, 867- 880.
- López, D., Jiménez, C.J., Baturone, I., Barriga, A., and Sánchez-Solano, S. 1998. *Xfuzzy: A Design Environment for Fuzzy Systems*. IEEE International Conference on Fuzzy Systems, Anchorage.
- Ross, T. J. 2004. *Fuzzy Logic with Engineering Applications*, Wiley.
- Sánchez-Solano, S., Cabrera, A., Baturone, I., Moreno-Velo, F.J., and Brox, M., 2007. *FPGA Implementation of Embedded Fuzzy Controllers for Robotic Applications*. IEEE Trans. on Industrial Electronics, 54, 1937-1945.
- Xilinx. 2008a. *MicroBlaze Reference Guides*.
- Xilinx. 2008b. *Xilinx System Generator v10.1 for Simulink. User Guide*.
- Zadeh, L. A., 1973. *Outline of a new approach to the analysis of complex systems and decision processes*. IEEE Trans. on Systems, Man, and Cybernetics, 3, 28-44.

SEGMENTATION OF THE SOLID BED IN INFRARED IMAGE SEQUENCES OF ROTARY KILNS

Patrick Waibel, Jörg Matthes and Hubert B. Keller
Institute for Applied Computer Science, Karlsruhe Institute of Technology
Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany
{patrick.waibel, joerg.matthes, hubert.keller}@kit.edu

Keywords: Rotary kiln, Solid bed, Segmentation, Infrared camera.

Abstract: This paper presents two novel methods for segmenting the solid bed in infrared image sequences of metal-recycling rotary kilns. Exploiting the different dynamics and temperatures of gas phase, solid bed and kiln wall, we developed filter chains for an image segmentation of the solid bed. For the image acquisition we employed infrared cameras with a spectral filter. Two image processing algorithms were realized according to the two most common camera positions (frontal and top-left view on the solid bed at the rear-end of the kiln). Results show that both algorithms are capable to segment the solid bed in the image sequences accurately and reliably. The work presented here provides a basis for the extraction of characteristic process state variables, that can help to improve the process control with regard to product quality, energy consumption and emission reduction.

1 INTRODUCTION

Rotary kilns are industrially used for processing materials at high temperatures. Reducing the energy consumption, improving the product quality and lowering pollutant emission are important goals for the operation of rotary kiln plants. A new approach to achieve these goals is an advanced process control that uses additional information from cameras that capture images from the inside of the kiln. An appropriate image processing system is necessary to extract meaningful information of the process state out of the camera images to be used for the process control.

A rotary kiln is a cylindrical vessel that is slightly inclined to the horizontal. While the raw material is mixed by the rotating movement the solids gradually move towards the kiln's lower end. High temperatures are attained by a burner inside the kiln and exothermic reactions of the material. Infrared cameras are capable to capture the spatial arrangement of the solid bed and the temperature distribution inside the kiln all at once. An image processing system can enhance the benefit of the acquired images by identifying specific process parameters, e.g. filling height, repose angle or movement pattern of the solids (Henein et al., 1983). Due to varying process conditions the robustness of the applied image processing algorithms is an important factor. In this paper we address the prob-

lem of segmenting the solid bed in infrared-images of metal-recycling rotary kilns from the two most common camera positions (frontal and displaced to the top-left).

(Zipser et al., 2006) describe a software tool for monitoring and analyzing of video and infrared images of combustion processes. In (He et al., 2009) an intensity-based Fuzzy-C-Means clustering algorithm for segmenting the solid bed in video images of alumina kilns is presented. (Sun et al., 2008) examine the segmentation of the solid bed and burner flame in video images based on texture information and Fuzzy-C-Means clustering. Our segmentation algorithms process infrared images and make use of intensity as well as dynamic properties in a particular filter chain to improve the overall segmentation results. After outlining the image acquisition in section 2 the image segmentation method developed for a frontal view position of the camera is described in section 3. Section 4 depicts the segmentation algorithm for a top-left view of the camera and section 5 concludes this paper.

2 IMAGE ACQUISITION

The analyzed image sequences were captured at a metal-recycling rotary kiln located in Freiberg (Ger-

many). The length of the kiln is 43m with an inner diameter of 3.6m. We used two different infrared-cameras, which were installed at the lower end of the kiln. The first camera (*scenario 1*) with a resolution of 256x128 pixels was located slightly below the rotational axis of the kiln (Figure 1 left). The second camera was installed at the top-left of the rotary axis (*scenario 2*). It has a resolution of 320x240 pixels (Figure 1 right). Both cameras are equipped with a spectral filter at 3.9 μ m where the absorbance of the burning gas atmosphere is at a minimum. The intensity values of the pixels correspond to absolute temperatures (400 to 2000°C) with an accuracy of ± 5 K. A developed image preprocessing system guarantees the validity of the acquired images. In the acquired image sequences the direction of rotation is anti-clockwise. Therefore the solid bed is moved upwards the right side of the kiln with regard to its current repose angle. On legal grounds, parts of the air supply unit had to be blacked out in the presented images.

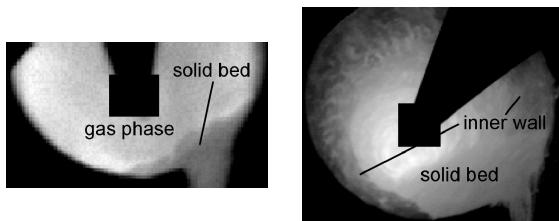


Figure 1: Scenario 1 (left) and Scenario 2 (right).

3 IMAGE SEGMENTATION - SCENARIO 1

3.1 Method

At lower average kiln temperatures the solid bed can easily be segmented via intensity differences, as the solid bed is colder than the gas phase in such situations. However, with increasing temperature the intensity differences of gas phase and solid bed vanish and a solely intensity-based segmentation algorithm fails (Figure 2). A robust filter algorithm has to handle both situation in an adequate way. The algorithm we developed makes use of different intensity properties as well as dynamic properties of the solid bed and the gas phase (Figure 3).

In the *intensity-based part* of the algorithm at first an *automatic thresholding* is performed on a region of interest (ROI) containing the possible locations of the solid bed. This method aims to find regions with homogeneous intensity values via multiple thresholding. Then all minimums in the intensity histogram of



Figure 2: Simple threshold segmentation at low (left) and high (right) mean temperatures.

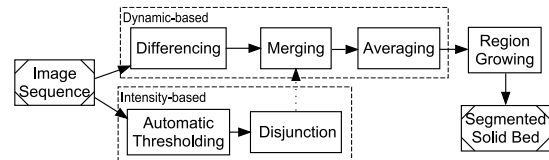


Figure 3: Solid bed segmentation algorithm for a frontal camera view on the kiln.

the ROI of a single image are located. Each minimum is used as a threshold segmentation. The segmented regions are subsequently examined if their mean intensity is below an offset (we used 30K) to the mean intensity of the entire ROI. Regions with mean intensities above this value are not further considered. The region with the highest permitted threshold is subdivided in connected pixel regions, whose areas (number of pixels) are computed. Connected regions with small areas (we used 30 pixels as threshold) are removed as fluctuations in the gas phase can generate these regions. Consequently, the remaining connected regions are defined as solid bed due to their mean intensities and sizes. A grouping of these regions ideally corresponds to the entire solid bed region. However, in many cases only a part of the solid bed is detected or there is no valid segmentation result at all (e.g. if the minimum search in the histogram is not successful). Therefore the dynamic properties of the solid bed and the gas phase are accounted for in the segmentation algorithm. The *dynamic-based part* is adapted from a modified calculation of the total variation (TV) of each pixel. Two temporally successive images are used to compute a difference image (*differencing*). The accumulated absolute differences of an image sequence reflects the total variation for each pixel which is an indicator for the fluctuation behavior of the intensity values. In our algorithm we normalize the total variation with the number of used frames in order to simplify the comparison of different frame limits. Therefore this step can be regarded as an *averaging* of the differences. The intensity values of the gas phase are usually more fluctuating than the solid bed's which leads to higher TVs in the gas phase region. Nevertheless it is possible that moving lumps within the solid bed, which are colder than the surrounding material, facilitate high TV values at their contours even inside the solid bed region. To circumvent this problem and to improve the overall seg-

mentation result the intensity-based part is addressed again. The *merging* step is the key element of the algorithm. The preliminary segmentation result of the intensity-based part is combined with the dynamic-based computations. Cold lumps within the solid bed are always either segmented solely or together with the rest of the bed by the automatic threshold step due to their low temperatures. The automatic thresholding is done with both images which are used for the respective differencing step. The pixels which are defined as solid bed in at least one of the images are approved as solid bed region in the *disjunction* step, i.e. a relocation of a lump can be captured. Now, all pixels included in the identified intensity-based solid bed region are set to 0 in the current difference image. This merging situationally leads to an enhanced distinguishability between the gas phase and solid bed in the averaged difference images. If the mean temperature of the kiln is high, the segmentation algorithm is predominantly based on the dynamics-based part since the intensity-based part rarely detects the solid bed region. When there is a high contrast between solid bed and gas phase in the input images the intensity-based part boosts the results of the dynamics-based part due to the merging step. Additionally, high difference values at the contours of moving lumps are prevented through this combination. For the final segmentation step a *region growing* algorithm is implemented.

3.2 Results and Discussion

In figure 2 segmentation results at the frontal view position with an fixed threshold segmentation are illustrated. The intensity-based threshold segmentation succeeds to detect the solid bed region in image sequences with low mean temperatures of the kiln. As soon as the temperature rises this method fails. Even an adaptive thresholding is not applicable at higher mean temperatures, as there are no intensity differences between gas phase and solid bed. Figure 4 shows segmentation results with the combined segmentation algorithm. They are each based on 200 frames. It can be seen that the modified averaged difference images possess high contrasts between solid bed and gas phase regions also at high mean temperatures of the kiln. This enables an accurate final segmentation with the region growing algorithm.

The developed combined segmentation filter chain proved to be a reliable and precise method to detect the solid bed in scenario 1 infrared images. This allows the extraction of process relevant solid bed features in future works. It is e.g. possible to define a circle segment with two parameters corresponding to

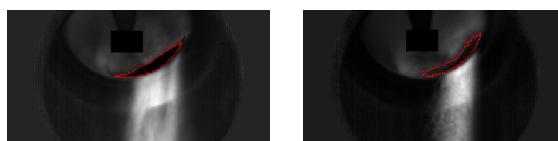


Figure 4: Segmentation with modified averaged difference images at low (left) and high (right) mean temperatures.

filling height and repose angle. In an iterative optimization process both parameters are adapted to find the best-fit of circle segment and segmented solid bed. Besides acquiring these two parameters, the remaining error between circle segment and solid bed region can be used to discriminate the bed movement (e.g. slumping or rolling).

4 Image Segmentation - Scenario 2

Conversely to scenario 1 in this case the solid bed has to be distinguished from the inner kiln wall instead of the gas phase. Intensity values cannot be used for the segmentation since the temperatures of kiln wall and solid bed are similar. Especially at the upper border of the solid bed, where solid parts stick to the wall, an intensity-based segmentation has little prospect of success. The filter chain we developed makes use of the different dynamic properties of the kiln wall and the solid bed. In particular we discriminate the steady rotating movement of the wall from the specific mixing movement of the solid bed.

4.1 Method

The first step of our segmentation process is a *mapping*. Two circles in the acquired images are set, in order to define the geometry of the inner surface of the kiln. Then a geometric mapping of the inner surface to a rectangle is performed (Figure 5 left). With the mapped intensity values a movement analysis is conducted. An optical-flow algorithm (Brox et al., 2004) computes a vector field out of two successive mapped images (Figure 5 right). Each vector specifies the direction and the magnitude of the movement of a pixel between two frames. Due to the mapping step the rotation of the kiln is transformed in a rightward movement. Consequently, discriminating the solid bed from the kiln wall corresponds to detecting the image region that is not constantly moving to the right. Since the material of the solid bed is also transported up along the kiln wall before it collapses respectively slides back in the opposite direction, an discrimination via movement is not possible at all times. Thus we implemented a moving average filter which averages the movements of the last

200 frames. In the so computed image the solid bed can be segmented with a simple threshold operator. Afterwards the segmented region in the mapped rectangular is mapped back to the original image. The single steps of the segmentation filter chain are illustrated in figure 6.

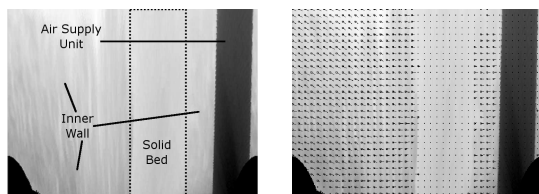


Figure 5: Mapping of kiln's inner surface to a rectangle (left) and vector field of mapped kiln's inner surface (right).

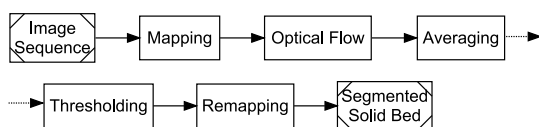


Figure 6: Solid bed segmentation algorithm for a camera view from top-left.

4.2 Results and Discussion

The averaged horizontal velocities of an image sequence are shown in figure 7 left. Brighter gray values correspond to higher velocities. Whereas the dark vertical stripe on the right side is due to the air supply unit, the dark stripe more to the left is caused by the dynamics of the solid bed. After a coarse definition of the ROI the lower part of the mapped solid bed is segmented via thresholding. The upper part is error-prone at some sequences because of the mapping process. Nevertheless the remapping of the segmented region in the original image achieves accurate results (Figure 7 right). In the next step extensions of the segmentation method will be examined. For instance, the best-fitting rectangular of the segmented region in the mapped horizontal velocities will be determined. This could improve the segmentation results in the more distant region of the solid bed. Additionally, the difference between the segmented region and the best-fitting rectangle could be used as an indicator for the particular movement pattern of the solid bed.

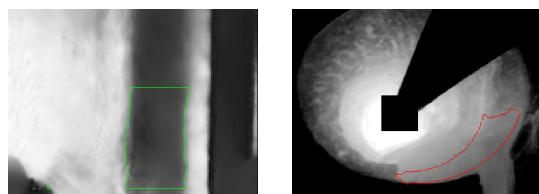


Figure 7: Averaged horizontal velocity values (left) and segmentation result (right).

5 CONCLUSIONS

The extraction of features from infrared image sequences of the inside of rotary kilns provides a high potential to improve the process control. An important requirement is the reliable and accurate segmentation of the specific regions in the acquired images. In this work we presented image segmentation filter chains that were capable to segment the solid bed in infrared images of a rotary kiln from the two most common camera positions. Results show that the combination of intensity-based and dynamic-based features considerably enhance the segmentation results at a frontal view position of the camera. From a top-left view on the solid bed the succession of a mapping process and an optical-flow computation enables satisfying segmentation results. In future works the correlation between extracted solid bed features with process states will be examined, particularly with regard to improvements for the process control.

REFERENCES

Brox, T., Bruhn, A., Papenberg, N., and Weickert, J. (2004). High accuracy optical flow estimation based on a theory for warping. In *ECCV*, pages 25–36.

He, M., Zhang, J., and Liu, X. (2009). Determination of the repose angle of stuff in rotary kiln based on imaging processing. In *ICEMI*, pages 4–97–4–101.

Henein, H., Brimacombe, J., and Watkinson, A. (1983). Experimental study of transverse bed motion in rotary kilns. *Metall. Trans. B*, 14(2):191–205.

Sun, P., Chaia, T., and jie Zhou, X. (2008). Rotary kiln flame image segmentation based on FCM and gabor wavelet based texture coarseness. In *WCICA*, pages 7615–7620.

Zipsper, S., Matthes, J., and Keller, H. (2006). Camera aided control of combustion processes with the software tool inspect. *at - Automatisierungstechnik*, 54:574–81.

VOICE FEEDBACK CONTROL WITH SUBJECT-IN-THE-LOOP

Dan Necsulescu, Lin Zhang, Elisha Pruner

*University of Ottawa, Department of Mechanical Engineering, Ottawa, Ontario, Canada
necsu@uottawa.ca*

Jerzy Sasiadek

*Carleton University, Department of Mechanical and Aerospace Engineering, Ottawa, Ontario, Canada
jsas@connect.carleton.ca*

Keywords: Signal Processing, Voice Feedback, Subject-in-the-Loop, Real-time Requirements, Signal Processing, Voice Training.

Abstract: A subject-in-the-loop feedback control system is composed of a bioengineering system including a subject, whose voice is received by a microphone, a computer that achieves the required signal processing of the sound signal by temporal and/or spectral computations and a speaker or earphones for auditory feedback to the subject of voice training. Frequency domain modifications of the signal are intended for voice training of subjects already familiar with traditional voice training. The objective of this paper is to present alternative methods for the implementation of subject-in-the-loop feedback control systems developed for voice training. The proposed feedback control scheme is an extension of the traditional control systems; feedback sensing and the control law are achieved by the human subject as a self-organizing controller. The experimental set-ups, developed for this purpose, contain programmable digital devices for real time modifications of the frequency content of the voice signal. The paper presents also a preliminary solution that satisfies the requirements for real-time operations, in particular that the subject does not perceive the delay between the sound generation and the auditory reception of the modified sound. The system performs spectral calculations for the analyses of the vocal sound signals. Preliminary experimental results illustrate the operation and the features of the proposed subject-in-the-loop real-time system for voice training.

1 INTRODUCTION

The purpose of this work is to develop a bioengineering experimental set-up with real-time capability for voice signal processing in a closed loop configuration. Acoustic loops refer to systems for signal amplitude increase or decrease of certain sound frequencies using digital manipulation of sound samples. The goal of this research is the design and construction of computer based modules for actors' voice training, as well as, for singers and public speakers who were already subjected to traditional voice training. Such an Audio-Formant Mobility Trainer is an adjunct to voice-training. The reason for previous training requirement results from the fact that the subject will have to produce different voice qualities for which it is necessary to have acquired a certain mobility of the bodily parts that produce speech. The device is intended to facilitate the production of new voice qualities by

increasing the mobility of one's voice formants. The Audio-Formant Mobility Trainer is a module that can perform acoustical experiments with the subject's voice and band-pass filtering for each of the formants for the purpose of auditory-feedback. Any formant can hence be chosen to be manipulated in order to increase the ability of the subject to perceive it in his own voice. There are three types of formant manipulations:

1. Intensity: varying the relative intensity of the formant bandwidth ranging from filtering it out to increasing it above the spectral envelope.

2. Bandwidth: increasing and decreasing its width.

3. Pitch: increasing and decreasing its pitch.

Typically, the learner uses a microphone and headphones while singing or speaking. His voice is analyzed and processed by the computer. This training could be very useful for actors and singers who are called to produce different voice types. It

may be also helpful for learners of new vowels using analysis called LTAS (Long Time Average Spectra), thereby determining the formants that characterize foreign language. Preliminary research work (Nesulescu and Weiss et al., 2006, 2005, 2008) led to the confirmation that complex acoustic phenomena can be simulated for the needs of designing acoustic hardware in the form of a closed loop experimental set-up for acoustics analysis. The presence of the subject in the control loop results in interesting new issues for the feedback control design.

2 AUDITORY FEEDBACK AND VOICE PRODUCTION

Previous research showed that changing voice quality by altering the auditory perception of one's voice is, to a limited degree, possible. If a person's sound production possibilities are enlarged (through voice training), then altered auditory feedback might facilitate the generation of different voice qualities (Neculescu, Weiss, and Pruner, 2008). The set-up consists in a subject hearing his voice through headphones while speaking into a microphone. However, the process allows a series of digital manipulations (temporal and spectral) designed to affect perception while examining the effects on vocal output. Whereas, the intensity feedback manipulations have been studied extensively (Purcell, and Munhall, Vol. 119 2006), (Purcell and Munhall 120, 2006), spectral changes effects on voice quality in auditory feedback and their relationship to voice production are still relatively unknown. Original proponents of the use of servo mechanical theory have claimed a direct effect on the vocal output when modified voice is fed back to the speaker. Essentially, according to this theory, if certain bandwidths of the voice spectra are modified in such a manner as to increase or decrease the energy in those regions, the person emitting those sounds will unconsciously react if the modified voice signal is fed back to his ears. The possibility of affecting voice output by auditory feedback remains a topic of intense interest for those involved in voice, speech and accent training (Neculescu, Weiss and Pruner, 2008). This work has the long-term goal to carry out audio-vocal filtering experiments including subjects with or without vocal training in order to determine whether voice training could allow for vocal adjustments in conditions related to filtered auditory feedback. This paper

describes the construction of computer based module for auditory feedback with no perceived temporal delay.

There are many teaching techniques in voice training, some auditory, some based on movement and some mixed. Independently of the technique, certain pedagogical approaches are often used. One of such techniques is bodily awareness through minimal movements (Purcell and Munhall, 2006). This objective of this approach is an effortless speech-motor learning system. A variable is introduced and the subject perceives it, plays with it, explores it, adjusts to it and integrates it in his own behaviour. This is the purpose of the Audio-Formant Mobility Trainer, an adjunct to voice-training when the learner has had already preliminary training with any traditional technique. The reason for the need for previous training is that the subject will have to produce different voice qualities for which it is necessary to have acquired a certain control of the mobility of the bodily parts that produce speech. The purpose of the device is expected to facilitate the production of new voice qualities by increasing the mobility of one's voice formants.

3 DESCRIPTION OF EXPERIMENT

The first experiment tries to ascertain whether it is possible to teach subjects to vary their fourth formant (F_4) at will. Previous research (Purcell, and Munhall, Vol. 119, 2006) has shown that subjects do it unconsciously when their auditory feedback is manipulated while uttering vowels. It is also known (Purcell and Munhall, 2006), that formant manipulation in pitch and bandwidth changes significantly the perceived voice quality

4 EXPERIMENTAL SET-UP

The main difficulty until recently was to achieve real-time capability in auditory feedback with programmable digital hardware. Some delay in auditory feedback cannot be avoided, but it is desired to reduce it, such that it will not be perceived.

The block diagram of the complete auditory feedback system is shown in Figure 1. Figure 2 shows this system in the traditional control system block diagram form. A human subject carries out in this case the feedback sensing, the comparator and

the controller (regulator). Figure 3 shows the block diagram of the PC based auditory feedback system, while Figure 4 shows the Simulink^R diagram of the real-time system for signal acquisition from AI, filtering, FFT frequency analysis, display and headphone signal generation to AO (Necsulescu, Weiss and Pruner.

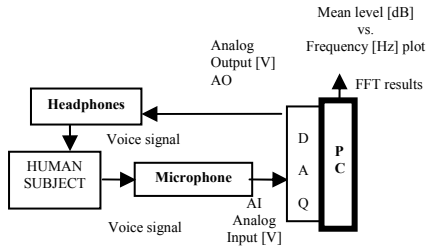


Figure 1: Block diagram of a generic auditory feedback system.

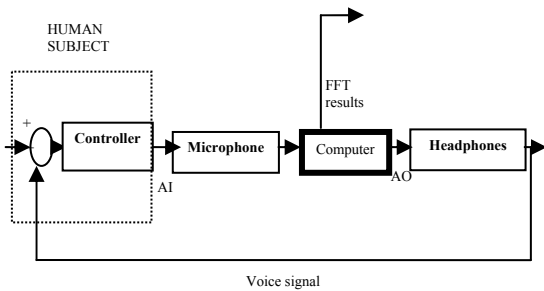


Figure 2: Block diagram of the auditory feedback system in feedback control (regulator) configuration.

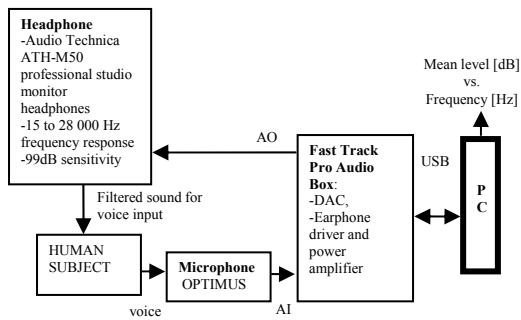


Figure 3: Block diagram of the PC based auditory feedback system.

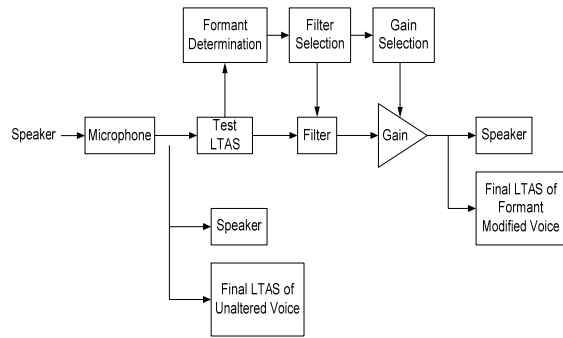


Figure 4: The diagram of the real-time system for signal acquisition from AI, filtering, FFT frequency analysis, display and headphone signal generation to AO.

5 PRELIMINARY EXPERIMENTAL SET-UP TESTING RESULT

The experimental setup was tested for verifying its performance. The current subject, used for experiments, has had extensive voice training. He sang for each audio-vocal filtering condition a 60 seconds French song using a neutral vowel. MATLAB representation of the amplitude versus time and the calculation of the Long Term Average Spectra (LTAS), permits the evaluation of the effects of voice signal processing (Purcell and Munhall, 2006). Figure 5 shows the frequency domain results of the sound signal in case of no headphones. These results are post-processed in frequency domain for the identification of formants

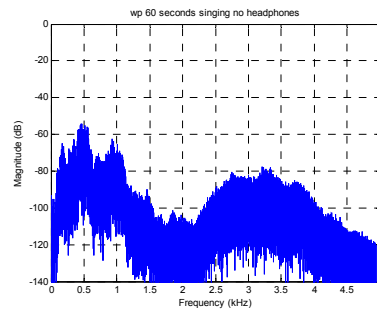


Figure 5: Results for LTAS of the sound signal with no headphones.

Figure 6 shows formant manipulation of the voice with LTAS for 500 Hz bandstop filtering. This figure shows what the subject heard following signal manipulation.

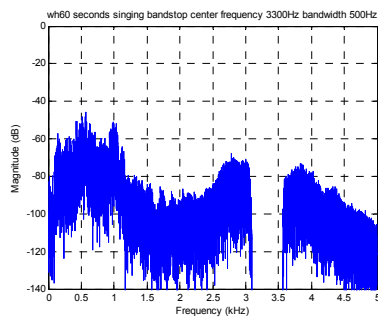


Figure 6: Formant manipulation of the voice with LTAS for 500 Hz bandstop filtering.

Bandstop hearing while singing produces a typical 3 zone spectrum: 1. The highest peaks, from fundamental frequency to 1100 Hz; 2. The second highest peaks, from 2500 to 3500-4000 Hz; 3. A Bowl, from 1100 to 2500 Hz.

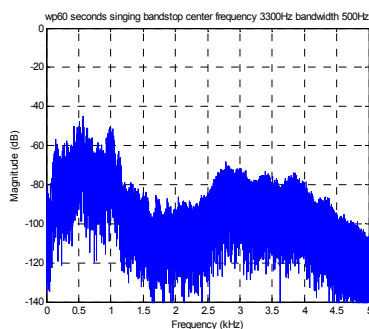


Figure 7: Recording the effects for the 500 Hz bandstop filtering.

The 500 Hz bandstop produces a clear peak between 3500-4000 Hz. This indicates that the bigger the gap the bigger the compensation. After hearing the signal shown in Figure 6, the resulting spectral form from Figure 7 appear similar in this case to the one produced while singing without headphones, shown in Figure 5. Figure 8 shows the results of the intensified amplitude of a 500 Hz about 3300 Hz. After hearing the signal shown in Figure 8, the subject produces the signal with the spectral content shown in Figure 9. The spectral form from Figure 9 differs significantly from the results from Figure 6, for the case of no headphones. The spectrum is flattened when compared to the three-zone spectrum of no headphones condition. This confirms that the real time signal was reduced to the desired frequency domain and the audio test based on the system shown in Figure 1 confirmed subjectively the validity of this result. This preliminary confirmation of the significant effects of signal processing on the subject is an encouraging

result for the prospect of using it in voice training. The subject produced different voice qualities unconsciously when given different auditory feedback conditions. The approach seems useful for training the voice for different voice qualities.

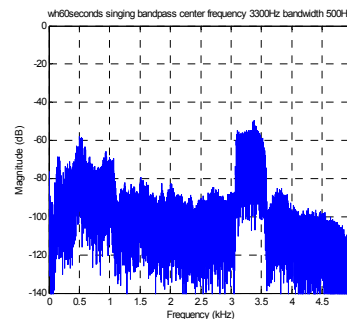


Figure 8: Formant manipulation of the voice with LTAS for 500 Hz frequency band amplification.

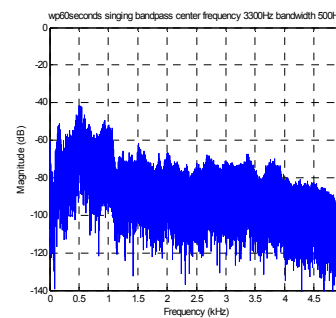


Figure 9: Recording of the effects of the 500 Hz frequency band amplification.

6 CONCLUSIONS

The experimental setup presented in this paper is useful for the voice training experiments in selected subjects. It includes:

- Long term average spectra under several conditions;
- Processing and storing data in the computer;
- Comparison of spectra with filtering and, without filtering;
- Comparison of experimental [trained] and non-trained groups.

The proposed auditory feedback experimental set-up proved to satisfy the requirements of acquiring signals with a microphone and filtering procedure, as well as, displaying and transmitting the modified signals to the earphones in real time. Moreover, auditory signals were FFT processed for the successful identification of each particular subject formants. This experimental set-up was

deemed appropriate for purposes of voice training for selected subjects.

REFERENCES

- Necsulescu, D., Zhang, W., Weiss, W., Sasiadek J., "Room Acoustics Measurement System Design using Simulation", *IMTC 2006 - Instrumentation and Measurement Technology Conference, Sorrento, Italy* 24-27 Apr 2006.
- Necsulescu, D., Advanced Mechatronics, *World Scientific*, 2009.
- Necsulescu, D, Weiss, W., Zhang, W., "Issues Regarding Hardware in the Loop Experimental Set-up for Room Acoustics", Montreal, McGill University, *20th Canadian Congress of Applied Mechanics*, 2005.
- Necsulescu, D., Weiss, W., Pruner, E., "Acousto-Mechatronic System for Voice Training", *Bul. I. P. Iasi, Tom LIV (LVIII), Fasc. X*, 2008, pp. 481 - 487.
- Purcell, D. W. and K. Munhall. "Adaptive control of vowel formant frequency: Evidence from real-time formant manipulation". *Journal of the Acoustical Society of America* 120(2), (2006), pp. 966-977.
- Purcell, D. W. and Munhall, K. "Compensation following real-time manipulation of formants in isolated vowels", *J. Acoust. Soc. Am. Vol. 119*, Apr. 2006, 2288–2297.

ON SOME PECULIARITIES OF CLUSTER ANALYSIS OF PERIODIC SIGNALS

V. Znak

*The Institute of Computational Mathematics and Mathematical Geophysics SB RAS, Pr. Lavrentieva 6, Novosibirsk, Russia
znak@opg.ssc.ru*

Keywords: Cluster analysis, Analysis of periodic signals, Extraction of signals.

Abstract: An important task of geophysical research is in the answer to the question about the quality of signals, i.e., estimating the locus of the signal and the degree of their presence in noises. Such indications determine the degree of trust to consequent estimations (e.g., estimations of wave arrival times). As seismic data are periodic signals in their nature, conventional means for examining such signals are Fourier and spectral analyses. However, this method does not allow us to clear up questions about probability of signals presence and their locus in the recorded data. We consider another approach – the cluster analysis of periodic signals, propose the formal conditions which must be satisfied by a period of signal existence, and give some results of analysis of real data recorded in field conditions.

1 INTRODUCTION

The prompting motive of our research is the needs in noisy geophysical data analysis. The basis of such data is periodic (harmonic or frequency-modulated) signals recorded at discrete instants of time. However, the corresponding signals are widely used, and an appropriate research can be of interest in other fields of activity.

In practice, geophysical data are recorded in field conditions. The point is that in the process of wave propagation and in recording data one or another type of errors takes place. Therefore, analysis of such data demands a special attention. Usually, in this case researchers attract a harmonic (I. I. Gurvich, and G. N. Boganic, 1980) or a spectral analysis (E. A. Davidova, and others, 2002). However, an appropriate approach cannot decide the dilemma “time-frequency” (the spectrum components are listed in a domain, where the time scale is absent).

Currently, methods of wavelet analysis and transformation (A. A. Nikitin, 2006, E. Baziw, 1994) are of interest to researchers. Here, time localization of the signal frequency components can be found. Essentially, such an approach is an analog to convolution or linear concordant filtration, or, in other words, it is a development of the window Fourier analysis.

We will treat periodic signals as time series and consider another approach, based on the cluster analysis. To the best of our knowledge, the notion of a cluster is used by few of authors to analyze periodic signals (Znak V. I. and Grachev O. V., 2009).

2 METHODOLOGY OF CLUSTER ANALYSIS OF PERIODIC SIGNALS

We can treat the time evolvent of a periodic signal on a plane as a specific image, and set a problem of studying some or other its features. However, such an image becomes considerably complicated in the presence of noises.

The problem can be simplified if an image of some integrated estimation of a signal is used as an object of analysis. Here, we offer to employ an estimation of a standard deviation (dispersion) on some running basis. The behaviour of such estimations as time function $\sigma(t)$ will reflect the energy distribution of a signal in the region of their existence. Then, the evolvent of function $\sigma(t)$ on a corresponding 2-D plane can be considered as an image of a cluster formation. Features of such an image are of interest for the purposes of analysis of periodic signals.

As the above estimation we use

$$\sigma(t) = \sigma_k(L) = \sqrt{\sum_{j \in L} (x_j - M(L))^2} / L \quad (1)$$

where L is odd, x is signal values, $j=(L-1)/2, \dots, N-(L-1)/2$ (N is a signal length). We will assume $\sigma(t)$ is integers.

Let $\hat{\sigma}$ be the uppermost dispersion value: $0 \leq \sigma_k \leq \hat{\sigma}$ and t_k is an instant of time. Then, some integer h ($0 \leq h \leq \hat{\sigma}$) will be called a "threshold". Thus, we juxtaposed with our signal estimations a grid $h_l \times \sigma_k$ on a 2-D plane, which will be denoted as Q : $h=0, \dots, \hat{\sigma}$; $k=(L-1)/2, \dots, N-(L-1)/2$. Further, we will suppose that each point of the grid represents an event $q_k(h) \in Q$, where $q \in (0,1)$:

$$q_k(h) \begin{cases} 1, & \text{if } \sigma_k \geq h_l, \\ 0, & \text{if } \sigma_k < h_l. \end{cases} \quad (2)$$

For any threshold h , the respective subset $Q_r(h) \subset Q$ for the adjacent instants will be called a cluster if for all the events q_k of $Q_r(h)$ the corresponding σ_k is greater or equal to the h , i.e.: $\forall q \in Q_r(h): q=1$. The cardinal number of such cluster is

$$\beta_r(h) = \sum_{q \in Q_r(h)} q, r = 1, \dots, m(h) \quad (3)$$

and locus in time is $\Delta t_r(h) = t_{1(r)}(h) \div t_{n(r)}(h)$.

Naturally, both the quantity of such clusters and the cardinal number of each cluster depend on the threshold value.

We can speak about two clusters of the two neighboring thresholds that a cluster $Q_{s(r)}(h+1)$ is a child of $Q_r(h)$ if they are intersecting in time: $\Delta t_r(h) \& \Delta t_{s(r)}(h+1) \neq 0$. We will pool such clusters and call them a cluster family. The cardinal number of this cluster family is

$$b_r(h) = \beta_r(h) + \sum_{s(r)=1}^{n(r)} \beta_{s(r)}(h+1) \quad (4)$$

etc. Let

$$B(h) = \sum_{r=1}^{m(h)} b_r(h) \quad (5)$$

be a common cardinal number of cluster families on the threshold h . Then the relation

$$P_r(h) = b_r(h) / B(h) \quad (6)$$

will be called a representative probability of the family $Q_r(h)$.

Let us consider a series of functions $P_r(h)$, $r=1, \dots, m(h)$, $h=1, \dots, \hat{\sigma}$. We expect that the behavior of such functions reflects the degree of the presence of a signal in noise. At the same time, they are tied to subjects, which have their own locus in time.

The matter of the problem is to investigate the behaviour of these functions for answering the questions about the degree of the presence of a periodic signal in noise, and its locus in noisy data.

3 ON STUDING THE SIGNAL EXISTENCE

Let us consider some cluster formation $\sigma_k(L)$ as an image \aleph under condition of any running basis L (Fig. 1).

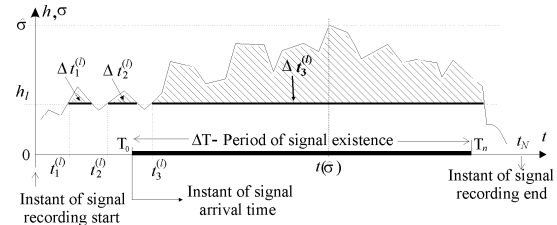


Figure 1: An example of the mapping of the dispersion estimations with a running basis.

Now, turn to the question about picking out the most informative threshold with regard to a representative probabilities. For answer on the question, we will study situations beginning from threshold $h=0$. However, first of all, we will make some remarks based on the nature of a signal in question. We suppose:

- 1) A process of signal recording in time ($t=0$) begins before a signal arrival, i.e., $T_0 > 0$, where T_0 is an instant of a signal arrival time.
- 2) A signal on the dispersion estimator input is $y=s+\xi$, where s is a source signal, and ξ is an additive white noise with zero mean Gaussian distribution. Let t_N be a signal recording period, and ΔT a signal existence period. Then, the following conclusion is a consequence of such supposition: the probability of localizing the uppermost dispersion value $k(\hat{\sigma})$ on ΔT is proportional to the ratios $t_N/\Delta T$ and to the signal-to-noise ratio, i.e., $P(\hat{\sigma}(k) \in \Delta T) \sim t_N/\Delta T$ & $s/\xi = f(t_N/\Delta T, s/\xi)$. (a more exact dependence needs a separate attention).

Now, we will study a cluster families beginning with threshold $h_0=0$. We can say, that the threshold $h_0=0$ is non-informative for us because we have $t_1(0)=0$ for a single clusters family $Q_1(h_0)$ (obviously, $P_1(h_0)=1.0$). We can say the same with regard to the threshold $h_1=1$ if the same conditions $t_1(h_1)=0$ for single $Q_1(h_1)$ ($P_1(h_1)=1$) are fulfilled, and so on.

Let, for the first time in threshold raising, h_l be such a threshold, where $t_1(h_l) > 0$ for $m(h_l) \geq 1$. In this case, we will have the three sets: 1) a set of instants of time of the beginning of cluster families $t_{1(r)}(h_l)$, 2) a set of periods of existence of the appropriate cluster families $\Delta t_{(r)}(h_l)$, and, 3) representative probabilities of the appropriate families $P_r(h_l)$ ($r=1, \dots, m(h_l)$).

Here, the following conditions are fulfilled:

- i) one (or more) of such instant of time at which the condition $t_{1(r)}(h_l) > 0$ is fulfilled;
- ii) a set of probabilities includes such $P_r(h_l)$, that the condition $P_r(h_l) = \max$ is fulfilled;
- iii) a set of periods of existence of cluster families includes such $\Delta t_r(h_l)$ that the condition $t(\hat{\sigma}) \in \Delta t_r(h_l)$ is fulfilled ($1 \leq r \leq m(h_l)$).

We will suppose that a locus of signal existence is reflected by such a period of existence of the cluster family $\Delta t_r(h_l)$, which fulfils (obeys) the following conditions:

$$\left. \begin{aligned} t_{1(r)}(h_l) > 0, \\ P_r(h_l) = \max, \\ t(\hat{\sigma}) \in \Delta t_r(h_l), \end{aligned} \right\} \quad (7)$$

where $\Delta t_r(h_l)$ is $t_{1(r)}(h_l) \div t_{n(r)}(h_l)$.

4 AN EXAMPLE OF STUDING A SIGNAL

The methodology in question was used for analysis of data recorded in the course of monitoring (in 2007) of the Karabetov mud volcano on Mt. of Taman Province (data recorded by Z-component of receivers for profile line T1, results of field experiments are currently accessible on the web_site <http://opg.sccc.ru>). Seismic (or vibro-) records were recorded from 10 T vibratory source with a frequency band of 10–64 Hz, and with a sampling frequency = 0.004 sec. Appropriate data can be found at <http://opg.sccc.ru/db>.

The results obtained are given in the Table 1. The columns of this Table include distances between

a vibratory source and a receiver (S), $h_{\min} \div h_{\max} = \hat{\sigma}$, periods of existence of the appropriate cluster families $t_1 \div t_2$, and appropriate estimations of representative probabilities (P) for running basis

Table 1: Estimations of signal locus in time.

S m.	L	$h_{\min} \div h_{\max}$	$t_1 \div t_2$	P
2363	25	6÷415	3933÷19988	0.94
	75	14÷396	3689÷15669	0.89
	125	16÷390	1520÷14247	0.91
2415	25	9÷305	2041÷19502	0.95
	75	16÷292	2284÷19672	0.96
	125	28÷279	3901÷9316	0.81
2461	25	13÷156	3173÷9917	0.45
	75	24÷133	3190÷10305	0.50
	125	27÷121	3210÷10471	0.52
2557	25	3÷64	3315÷19988	0.90
	75	6÷53	3481÷19963	0.92
	125	6÷46	3455÷19938	0.92
2601	25	5÷165	4045÷19988	0.88
	75	11÷151	4060÷18122	0.85
	125	15÷132	3741÷18114	0.88
2647	25	8÷98	3493÷8855	0.42
	75	12÷88	3593÷14414	0.73
	125	14÷86	2213÷13365	0.77
2698	25	2÷13	8642÷19988	0.73
	75	3÷13	8832÷19963	0.77
	125	3÷13	8807÷19938	0.77
2749	25	6÷87	1410÷12065	0.59
	75	8÷84	1150÷19963	0.98
	125	9÷82	1170÷19938	0.99
2796	25	6÷87	2637÷15919	0.89
	75	10÷78	972÷15800	0.98
	125	10÷71	360÷16047	0.99
2845	25	15÷111	3508÷10413	0.58
	75	21÷105	569÷11594	0.82
	125	22÷100	898÷11615	0.83
2894	25	6÷51	3897÷18815	0.89
	75	9÷46	3889÷15243	0.88
	125	10÷43	3863÷11205	0.59
2999	25	5÷76	9532÷19988	0.51
	75	8÷67	9573÷19963	0.5
	125	8÷56	1824÷19938	0.96
3046	25	6÷60	5646÷19988	0.80
	75	8÷52	2209÷19963	0.97
	125	8÷47	2185÷19938	0.97
3095	25	5÷111	3987÷19988	0.93
	75	7÷90	3988÷19963	0.95
	125	7÷76	3963÷19938	0.95
3141	25	12÷67	3665÷11019	0.65
	75	17÷61	3838÷9707	0.71
	125	18÷54	4054÷9388	0.74
3198	25	12÷74	4945÷19988	0.81
	75	8÷66	6011÷9346	0.43
	125	18÷64	4288÷19938	0.87

$L \in \{25, 75, 125\}$.

Estimations of signal locus in time for $S \sim 0$ m are given in the Table 2 (all the representative probabilities are equal to 1).

The processing and analysis data were obtained by means of interactive computer system of designing and support of one-dimensional weighed order statistics filters (V. I. Znak, 2009).

Table 2: Estimations of signal locus in time for $S \sim 0$ m.

L	$h_{min} \div h_{max}$	$t_1 \div t_2$
25	6 ÷ 415	3870 ÷ 18883
75	14 ÷ 396	3846 ÷ 18908
125	16 ÷ 390	3821 ÷ 18933

By way of example, an image of investigated signal (for $S=2647$ m) and appropriate dispersions is shown in Fig. 2.

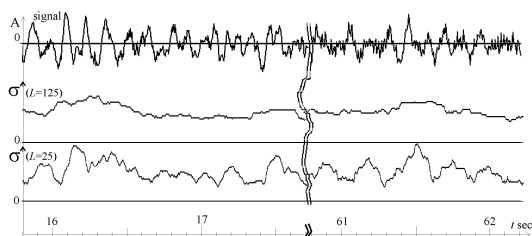


Figure 2: An image of signal and appropriate dispersions for $L=25, L=125$ ($S=2647$ m).

Estimations of the data from the Table 1 are given in Fig. 3.

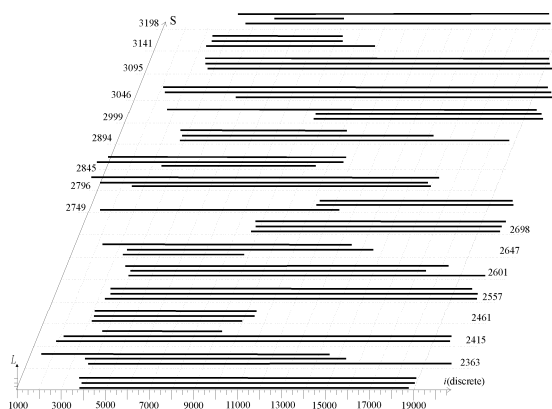


Figure 3: Estimations of the time locus of the signal for running basis $L \in \{25, 75, 125\}$ and for different distances.

5 CONCLUSIONS

We have considered the approach of cluster analysis of periodic signals, proposed the formal conditions

which must be satisfied by a period of signal existence, and given some results of analysis of real data recorded in field conditions. Analysis of the results obtained by studying real signals allows us to say that the approach in question can result in close estimations of a locus in time of a pure signal, and in less close estimations of a locus in time of noisy signals.

Our main objective was restricted by development of the method of formalized analysis of periodic signals for estimation of their period of existence. We have not concerned methods of improving signals as it is a theme of separate investigation. We suppose that more exact decisions can be attained by attracting analysis of the left and the right uniformity of cluster families (Znak V. I., 2009) and frequency processing (Znak V. I., 2005). Cluster families, which reflect a locus of a signal on its boundaries, must have a higher uniformity than for others.

The work is supported by the grant 09-07-00100.

REFERENCES

Gurvich I. I., Boganic G. N., 1980. *Seismic research*. Moscow, "Nedra" (in Russia).

Davidova E. A., Copilevich E. A., Mushin I. A., 2002. Spectral-time method for a mapping of types of geological layers, *Reports of RAS*, 385(5), pp. 682-684, (in Russia).

Nikitin A. A., 2006. New tricks of geophysical data processing and their well-known analogous. *Geophysics*, No 4, pp. 11-15 (in Russia).

E. Baziw, 1994. Implementation of the Principle Phase Decomposition Algorithm," in *Proc. IEEE Transactions on signal processing*, July 2007 **45** (6), 1775-1785.

Znak V. I., and Grachev O. V., 2009. Some Issues in improving quality of noisy periodic signals and estimating their parameters and characteristics numerically by using a cluster approach: problem statement. *Numerical Analysis and Applications*, 2(1), pp. 34-45.

Znak V. I., 2009. Some aspects of estimating the detection rate of a periodic signal in noisy data and the time position of its components. *Pattern Recognition and Image Analysis*, 19(3), pp. 539-545.

Znak V. I., 2005. Co-Phased Median Filters, Some Peculiarities of Sweep Signal Processing. *Mathematical Geology*, 37(2), pp. 207-221.

V. I. Znak, 2009. Some Questions of Computer Support of Designing and Accompanying of One-Dimensional WOS Filters. *Journal of Siberian Federal University, Mathematics & Physics*, 2(1), pp. 78-82 (in Russia).

INTERACTION OF A FLEXIBLE ROBOT WITH ITS ENVIRONMENT

Michael Kastner, Hubert Gattringer, Hartmut Bremer

Institute for Robotics, Johannes Kepler University, Altenberger Strasse 69, 4040 Linz, Austria
{*michael.kastner, hubert.gattringer, hartmut.bremer*}@jku.at

Martin Ramsauer, Paolo Ferrara

FerRobotics Compliant Robot Technology GmbH, Altenberger Strasse 69, 4040 Linz, Austria
{*martin.ramsauer, paolo.ferrara*}@ferrobotics.at

Keywords: Fluidic Muscles, Pneumatic Robot, Contact Detection, Interactive Programming, Human Robot Interaction.

Abstract: Uncomplicated and safe programming interfaces as well as flexible programs themselves become important when robots are used for small lot size tasks or are operated by personnel without special robotics education. This work takes a look at safe and easy interaction of a flexible articulated robot arm – actuated by fluidic muscles – with its environment. A contact detection scheme for stiff collisions at speeds between 50 and 250 mm/s is presented and measurement results are discussed. Moreover, a programming by demonstration concept is described on the basis of a pick and place task. Both strategies (implemented on a seven axis handling robot) rely on physical models to allow an operation without extra sensors.

1 INTRODUCTION

Programming a robot requires a high level of subject-specific knowledge and concentration. Human errors can cause great damage to the robots environment – including people. Studies showed that most robot related accidents occur during programming or fault recovery (Clark and Lehto, 1999). For small lot sizes, frequent reprogramming and adaptation increases the occurrence of these scenarios, thus heightening the overall probability of harm to process equipment and personnel.

Fluidic muscles are interesting actuators for devices interacting with humans, because of their compliance. See (Daerden and Lefeber, 2002) for a technology overview and (Van Damme et al., 2005) for more on soft manipulators. Besides safety considerations covered in (Bicchi and Tonietti, 2004), this compliance can also be used to let the robot be more aware of its surroundings.

For this paper we consider a robot actuated by commercially available muscles (Hesse, 2003) of the McKibben type. We show how the physical model of the robot and its actuators can be used in two scenarios of robot environment interaction. This is done without the need for any sensors in addition to the ones present in the control loops (see Figure 1b).

In the first setting, the robot moves along a trajectory at rather low speed and should be able to detect a (stiff) collision with its surrounding at the tool without causing damage. This sense of touch is meant to free the programmer from the necessity to provide the exact data of the process setup to the robot. Measurement results for different speeds and spatial directions are discussed.

The second part describes the direct interaction of the operator with the robot to allow the programming of a simple pick and place task completely by demonstration.

2 SYSTEM ARCHITECTURE

The robot configuration (1a) comprises seven rotational joints. Five are actuated pneumatically. Of these, three (1, 2 and 4) are driven by the one degree of freedom muscle setup detailed below. The hand joint (6, 7) is operated by a combination of three fluidic muscles and a cardan mechanism. The remaining two joints (3, 5) contain standard brushless DC motors with harmonic drive gearing.

As shown in Figure 1b, a pneumatic muscle drive consists of two muscles (Figure 1c) in an antagonistic setup connected by a sprocket chain. The linear mo-

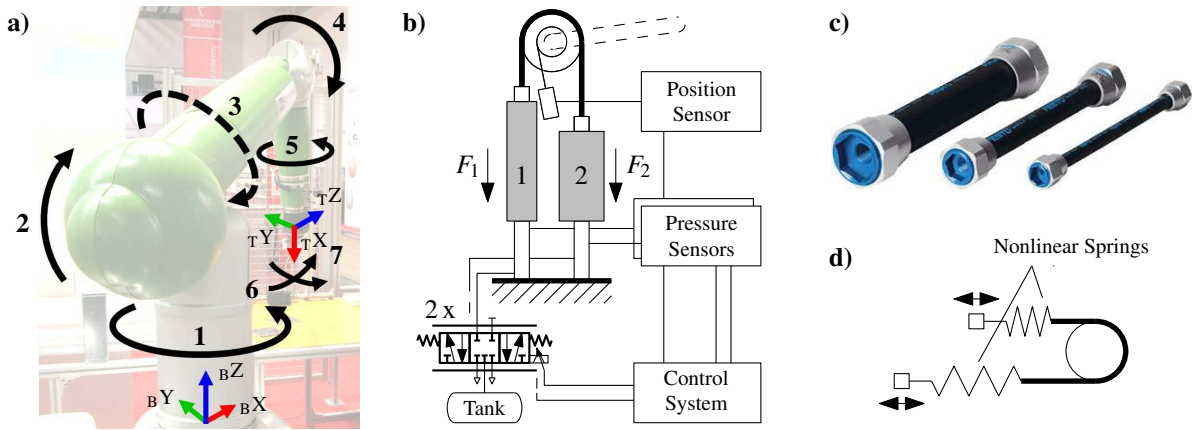


Figure 1: Kinematic setup (joints, base and tool coordinate systems) of the considered robot *Romo* (a), mostly actuated by pneumatic drives like (b). The drives utilize fluidic muscles (c) and their characteristic is similar to the scheme shown in (d).

tion of the actuator is transformed into a rotation by an according sprocket wheel supported by a ball bearing. The absolute rotation angle is measured by a cable extension sensor. Airflow to and from the muscles is regulated by proportional directional valves. The air pressure inside each actuator is measured by a sensor mounted near the inlet. All sensors are connected to 16 bit A/D converters.

The compliance of the system stems from the soft muscles and the compressible work medium air. In this respect the actuation is roughly equivalent to two nonlinear springs with adjustable pre-tensioning (see Figure 1d).

3 MODELING

3.1 Fluidic Muscles

The static relations between force F , contraction h and pressure p of the fluidic muscles (Festo MAS, Figure 1c) are provided in the form of datasheet diagrams by the manufacturer. For this work, we used approximations of the form

$$F = a(h)p + b(h), \quad (1)$$

where $a(h)$ and $b(h)$ are polynomials of order three and six. Their coefficients have been identified in experiments. One model muscle was used for each different diameter.

When two fluidic muscles are combined into an antagonistic setup, the resulting actuator torque is

$$Q_M(\Delta p, q) = r_S (F_2 - F_1), \quad (2)$$

with the sprocket wheel radius r_S and the muscle forces F_1 and F_2 (see Figure 1b). Q_M can be written

as a function of the pressure difference $\Delta p = p_2 - p_1$ and the joint angle q , to which the contractions h_1 and h_2 are geometrically linked.

3.2 Multi Body System

For modeling the mechanical part of the robot, the equation of motion of the multibody system

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Q}_M \quad (3)$$

is used. Here, \mathbf{q} and its time derivatives are the vectors of the joint angles, velocities and accelerations respectively, \mathbf{M} is the mass matrix, \mathbf{g} a term that includes gravitational forces, coriolis forces and so on. A calibration term is also part of \mathbf{g} . The entries of the vector \mathbf{Q}_M are the torques of the joint actuators (mostly muscle pairs like in eq. 2). We calculated and implemented this model by using the projection equation in subsystem formulation (Bremer, 2008)

$$\sum_{k=1}^N \left(\frac{\partial \dot{\mathbf{y}}_k}{\partial \dot{\mathbf{q}}} \right)^T (\mathbf{M}_k(\mathbf{y}_k) \ddot{\mathbf{y}}_k + \mathbf{g}_k(\mathbf{y}_k, \dot{\mathbf{y}}_k) - \mathbf{Q}_k) = \mathbf{0} \quad (4)$$

which allowed us to combine smaller segments k (one drive and the attached arm each) of the robot in a modular way. \mathbf{y} are describing coordinates used for the separate subsystems while the rest of the notation is analogous to the one used in Equation 3. Geometry and inertia data were exported from the CAD construction files, damping terms were identified on the real system.

4 CONTACT DETECTION

A robot manipulator is moving along some path (Figure 2). At an unknown location, it will collide with its

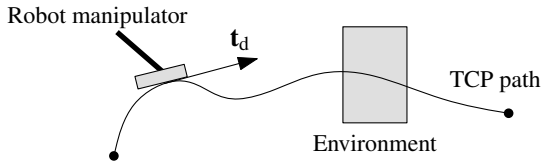


Figure 2: Contact detection scenario.

surrounding. This is deliberate but must be handled in a manner as not to inflict damage. An algorithm should be found to allow the robot an autonomous (i.e. without the use of any extra sensors) detection of such a collision.

After the detection, the robot should abort the maneuver, report the incident to the sequence control and hold its current position. Due to the limited bandwidth of the pneumatic robot system, it is impossible to avoid the peak in the contact force upon collision. This has to be dealt with by choosing an appropriate approaching velocity. After the impact, the robot should not push against the surrounding.

The focus for this task is on two stiff clashing objects. (Haddadin et al., 2008) cover similar topics for an electrically actuated robot with focus on safety.

4.1 Contact Force Estimation

In the contact case, the model multibody dynamics Equation 3 does not hold any longer. There is a remaining term

$$\mathbf{Q}_C = \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{Q}_M(\mathbf{q}, \mathbf{p}) \quad (5)$$

that we interpret as a vector of external contact torques acting on the joints. Deviations from the model are also included there as well as measurement errors. The velocities $\dot{\mathbf{q}}$ and accelerations $\ddot{\mathbf{q}}$ are estimated from the position sensor data.

As already mentioned, we are interested in the tool contact scenario. Therefore, we use the relation for the virtual work

$$\delta \mathbf{x}_{TCP}^T \mathbf{F}_C = \delta \mathbf{q}^T \mathbf{Q}_C \quad (6)$$

and

$$\delta \mathbf{x}_{TCP} = \frac{\partial \mathbf{x}_{TCP}}{\partial \mathbf{q}} \delta \mathbf{q} = \frac{\partial \mathbf{v}_{TCP}}{\partial \dot{\mathbf{q}}} \delta \dot{\mathbf{q}} \quad (7)$$

to get an equivalent tool center force estimation,

$$\mathbf{F}_C = \left(\left(\frac{\partial \mathbf{v}_{TCP}}{\partial \dot{\mathbf{q}}} \right)^T \right)^{-1} \mathbf{Q}_C. \quad (8)$$

The vector $\mathbf{x}_{TCP} = [x \ y \ z]^T$ contains the cartesian coordinates of the tool center point, \mathbf{v}_{TCP} is the according absolute translational speed. From the kinematics of the robot (Figure 1a), one can see that for a

reasonable tool geometry, joints 6 and 7 are insensitive to contact forces because of the small lever arms. More noise than information would be added, which is why we did not include these joints in our algorithm. Joints 3 and 5 are also excluded as they are not actuated pneumatically. The Jacobian using the velocities then reads

$$\frac{\partial \mathbf{v}_{TCP}}{\partial \dot{\mathbf{q}}} = \begin{bmatrix} \frac{\partial v_x}{\partial \dot{q}_1} & \frac{\partial v_x}{\partial \dot{q}_2} & \frac{\partial v_x}{\partial \dot{q}_4} \\ \frac{\partial v_y}{\partial \dot{q}_1} & \frac{\partial v_y}{\partial \dot{q}_2} & \frac{\partial v_y}{\partial \dot{q}_4} \\ \frac{\partial v_z}{\partial \dot{q}_1} & \frac{\partial v_z}{\partial \dot{q}_2} & \frac{\partial v_z}{\partial \dot{q}_4} \end{bmatrix}. \quad (9)$$

4.2 Detection Criterion

We base the decision whether a tool impact occurred or not on the two basic ideas that in the contact case

1. the model Equation 3 changes to Equation 5 and
2. the controller (which was designed for free trajectories) performance declines significantly resulting in a limited tracking accuracy.

The most simple way to utilize the first idea is to compare the projected contact force

$$\mathbf{F}_C = \mathbf{F}_C \cdot \mathbf{t}_d \quad (10)$$

(\mathbf{t}_d is the tangent vector of the desired path, Figure 2) to some threshold L in the form

$$C = \begin{cases} 1 & \text{if } F_C > L \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

where $C = 1$ means that contact is detected. Practically, it was impossible to find an L value resulting in a low amount of false positives and negatives over the desired workspace and speed range. This seems to stem from the fact that the actuator model described in Section 3.1 neglects too many effects (like drift and hysteresis) to provide an accurate force measure.

To just detect the impact, we modified the criterion to use

$$\dot{F}_C(s) = \frac{s}{(s + \lambda)^2} F_C(s), \quad (12)$$

an estimation of the force derivative with some additional low-pass filtering applied.

We also conducted experiments with the second aforementioned idea. Including the controller error on acceleration level in the form

$$D = \dot{F}_C \left(\frac{(\ddot{\mathbf{x}}_d - \ddot{\mathbf{x}})}{\|\ddot{\mathbf{x}}\|} \cdot \mathbf{t}_d \right) \quad (13)$$

showed the best results when again compared to some constant level L . Index d denotes desired values. For very low speeds (below 50 mm/s) the only robust detection found was via the position error ($\mathbf{x}_d - \mathbf{x}$) which led to a rather big delay in the detection. In most cases, this will render the detection useless as such slow movements are typically used for delicate handling scenarios.

4.3 Measurement Results

We evaluated the detection by driving the manipulator against a workbench at various speeds (50, 100, 150 and 250 mm/s), in different directions (X+, Y+, Z- – see Figure 1a – and mixed diagonal) and with 0.5 or 5 kg of payload mass. Each combination was measured ten times.

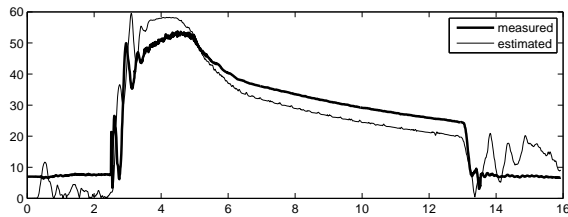


Figure 3: Example of estimated normal contact force in N over time in s – compared to external sensor measurement.

Although the force characteristic (measured with a three axis force sensor) varied over the different scenarios, the detection was successful for nearly all cases without adapting the threshold. Most problematic was the combination of low speed and small payload. Here we found at the same time false negatives (5 for X+) and positives (3 for Y+). Other than that, only one detection of over 300 failed.

The contact force – excluding impact peaks – always stayed below 20 N (and mostly below 10 N for speeds below 150 mm/s).

5 SHOW-DO PROGRAMMING

In guidance mode, a human can grasp the robot and move it freely through the workspace. The robot is still supported by the torques resulting from the model in Equation 3 but any additional effort from the controller is limited to very low values. Due to the soft joints, one can interact with the robot along the complete structure.

The guidance mode can be used to quickly move the robot out of the way, to interactively teach positions or to record complete continuous path segments. In our showcase we used a handle with two buttons, mounted on the lower arm of the robot, to let the user teach complete pick and place applications – similar to a macro recorder known from personal computer software. Tool actions trigger special behaviour – for example the "pick" macro at execution time moves the manipulator in tool direction until it touches the workpiece and does not solely rely on the recorded position information.

The interaction with the robot proved to be intuitive and all kinds of people were quickly able to perform programming tasks.

6 CONCLUSIONS

In this paper we show how to use models of the mechanical system and the actuators of a pneumatically driven robot for interaction with the environment. The illustrated approach worked well when the human operator compensated the remaining model uncertainties. The completely autonomous interaction also showed good results for the simple case of contact detection.

A first effort to employ the presented interaction possibilities for more intuitive programming led to promising feedback from users.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Austrian Center for Competence in Mechatronics (ACCM) for their support.

REFERENCES

- Bicchi, A. and Tonietti, G. (2004). Fast and "soft-arm" tactics [robot arm design]. *Robotics & Automation Magazine, IEEE*, 11(2):22–33.
- Bremer, H. (2008). *Elastic Multibody Dynamics: A Direct Ritz Approach*, chapter 4: Rigid Multibody Systems, pages 59–113. Springer-Verlag GmbH.
- Clark, D. R. and Lehto, M. R. (1999). *Handbook of Industrial Robotics*, chapter 36: Reliability, Maintenance and Safety of Robots, pages 717–754. John Wiley & Sons, 2 edition.
- Daerden, F. and Lefeber, D. (2002). Pneumatic artificial muscles: Actuators for robotics and automation. *European journal of mechanical and environmental engineering*, 47:11–21.
- Haddadin, S., Albu-Schäffer, A., De Luca, A., and Hirzinger, G. (2008). Collision detection and reaction: A contribution to safe physical human-robot interaction. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3356–3363.
- Hesse, S. (2003). *The Fluidic Muscle in Application*. Festo AG & Co.KG.
- Van Damme, M., Daerden, F., and Lefeber, D. (2005). A pneumatic manipulator used in direct contact with an operator. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*.

EXPERIMENTING WITH AUTONOMOUS CALIBRATION OF A CAMERA RIG ON A VISION SENSOR NETWORK

Kyung min Han, Yuanqiang Dong and Guilherme DeSouza
Department of Electrical and Computer Engineering, University of Missouri
349 Engineering Building West, Columbia, U.S.A.
{kx8d, ydfff}@mizzou.edu, desouzag@missouri.edu

Keywords: Vision sensor network, Image clustering, Global coordinate reference.

Abstract: This paper presents a completely autonomous camera calibration framework for a vision sensor network consisting of a large number of arbitrarily arranged cameras. In the proposed framework, a sequence of images for calibration is collected without a tedious human intervention. Next, the system automatically extracts all necessary features from the images and finds the best set of images that minimizes the error in 3D reconstruction considering all cameras in the set.

1 INTRODUCTION

Calibration of multiple camera systems became an important topic along with vision sensor networks (VSN) such as for: virtual and augmented reality; surveillance; battle field reconnaissance; etc. (Remagnino and Jones, 2002; Jaynes, 1999; Koller et al., 1997). In order to calibrate a VSN several critical steps must be taken: 1) acquiring images synchronously; 2) extracting feature points from the images; 3) establishing the correspondence among the feature points in multiple images from multiple cameras; 4) performing individual camera calibrations; and 5) computing a global coordinate reference for all cameras. Currently, a few of these steps still require a number of tedious and manual subtasks such as: selecting good images from which feature points can be extracted for calibration; manually establishing the correspondences between feature points from different cameras; etc. These tasks become quite challenging and time consuming especially when the VSN has a large number of cameras. Moreover, a great human involvement in the calibration process can induce errors that could lead to a poor overall accuracy of the system. Therefore, it is quite desirable that vision sensor networks can be autonomously calibrated.

In (Huang and Boufama, 2002), for example, a semi-automatic calibration system was developed for augmented reality. However, the method presented still requires that the user clicks on four points per image in order to construct homography matrices. Besides the tedious requirement of clicking on a large

number of points, the user is also required to be very careful when performing this task. Otherwise, the accuracy of the calibration will degenerate with every point wrongly selected.

Another common approach is to resort to some special marker, such as a laser pointer (Svoboda et al., 2005) or a LED stick (Baker and Aloimonos, 2000). One of the main drawbacks of these kinds of systems is in the quite large number of images that must be obtained in order to cover a reasonable small space – since only one or two feature points can be obtained from each image.

In (Olsen and Hoover, 2001), a system to calibrate cameras in a hallway was proposed using several square tiles. Similar to the landmarks in (Koller et al., 1997), their method not only requires that several tiles be carefully positioned, but also that the area covered by the tiles spans the field of view of all cameras in the hallway.

A pattern-free approach was proposed in (Chen et al., 2005), where the trajectory of a bouncing ball is used for calibration. Yet their method is tested only using computer simulation and it is unclear whether their algorithm can perform at all in a real situation. Another pattern-free approach is the system described in (Yamazoe et al., 2006). In that case, a geometry constraint is used to extract feature points from a human silhouette. However, their method requires a traditional pre-calibration step in order to estimate the fundamental matrices used for the final calibration.

In this paper, we present a completely autonomous framework that performs optimal multi-camera cali-

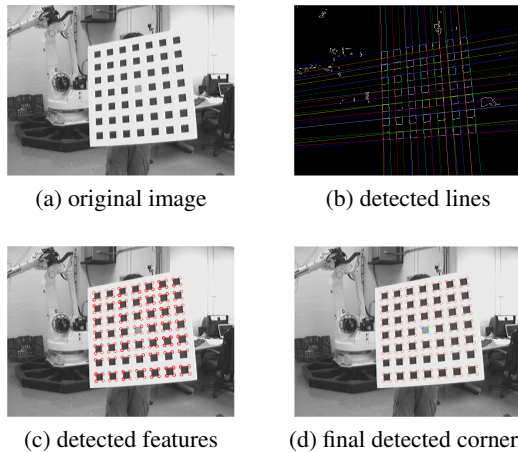


Figure 1: Steps of the detection algorithm in (Han and DeSouza, 2007).

bration in terms of the final error in 3D reconstruction for any given subset of the cameras. In order to achieve that, the system only requires that a sequence of images be captured by the cameras while a human presents a calibration pattern at arbitrary poses in front of the cameras. Then the proposed algorithm automatically: searches for feature points on the pattern that will be used for calibration; selects the best set of images that optimizes the overall accuracy from calibration; and computes the individual camera calibrations as well as the best sequence of transformations from the camera to a global coordinate frame.

2 THE PROPOSED ALGORITHM

As we mentioned earlier, the proposed framework for the calibration of multiple cameras in a vision sensor network consists of several steps. In this paper we will focus only on steps 2, 3 and 5 above.

Figures 1(a)-(d) show the various steps of our **feature detection** algorithm developed in (Han and DeSouza, 2007). As illustrated by these figures, a small number of spurious lines and consequently spurious feature points are initially detected due to noise in the image. However, after a few more steps into the detection algorithm, all spurious feature points are eliminated, as depicted in Figure 1(d). Finally, given the shape of the pattern, the algorithm automatically establishes an order to each corner point. This ordering system is later used for **feature correspondence**.

The existence of noise in the images greatly affects the performance of the feature detection algorithm. Hence, it is necessary to evaluate these images, assign a score to them, and choose only those that can lead to the best calibration. The algorithm initially as-

signs a uniform score of 100 to every image acquired. Then, algorithm starts to deduct a penalty whenever the image fails to satisfy a certain expectation. To **rank the image**, the penalty is formulated as:

$$\begin{aligned} Score_j &= 100 - 100 \times \sum_{i=1}^n p_i + s_i \times |N - n| \\ s_i &= \max(stdv_{u_i}) + \max(stdv_{v_i}) \\ p_i &= \frac{(stdv_{u_i} + stdv_{v_i})}{N} \end{aligned} \quad (1)$$

Where $stdv_{u_i}$ and $stdv_{v_i}$ are the standard deviations in, respectively, the u and v coordinates of detected corners; N is the total number of corners in the pattern and n is the number of corners detected. The rationale behind this scoring scheme is to assign a penalty that is proportional to the uncertainty ($stdv$) in the detected feature point.

Once an image rank is created using the above scores, the algorithm must start selecting images for calibration. We **cluster the images** according to two non-exclusive criteria: orientation (*straight-centered*, *tilted-forward*, *tilted-backward*, *tilted-to-the-left* and *tilted-to-the-right*) of the pattern and its distance (*near*, *medium* and *far*) to the camera. The clustering of the images is performed by a K-means algorithm using the the angles of the edges and the pattern's apparent size. Once the clusters are formed, the algorithm selects for each camera calibration five images according to the rank $Score_j$. That is, one image from each of the five orientations is selected from the *medium* clusters. Next, two more images from the *near* and *far* clusters are selected based solely on their ranks – that is, these images can come from any of the five orientation clusters. Finally, two extra images of the pattern are selected for every pair of cameras that share a view of the pattern at that pose. Once the images are chosen, the calibration is performed using a popular method found in the literature (Zhang, 2000).

The final step of the algorithm is the problem of **finding the best transformation** from the coordinate frame of any camera i to any camera j . However, not all possible paths between cameras assure the same accuracy in 3D reconstruction. Due to the quality of the image used for calibration, some paths may lead to better accuracy than others. We approached this problem using a graph search algorithm which is the same as the problem of finding all-pairs shortest paths. We first compute the shortest-path where weights are scored as described above. Next we apply the Floyd-Warshall Algorithm (Floyd, 1962) to find the best path and therefore the best transformation between camera coordinate systems.

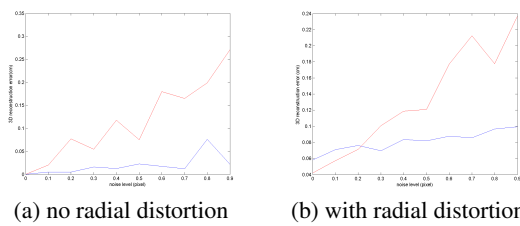


Figure 2: Errors of 3d reconstruction vs. levels of noise. Blue lines denote the errors using the best images and red lines denote the errors using poorly ranked images.

3 EXPERIMENTAL RESULTS

We tested our proposed algorithm for two different situations. The first group of tests was done with synthetic data and we used from 6 to 42 virtual cameras. The second group of tests was done with real data using 6 cameras.

Using the intrinsic parameters from real cameras, we initially created a virtual space with 18 cameras – all pointing to the center of the space. We set the origin of this space at the center, so that the positions of all 18 cameras could be easily determined. These arbitrary intrinsic and extrinsic parameters of the cameras will later be referred to as our ground truth. Next, two thousand positions of the **synthetic** pattern were randomly generated and noise plus camera radial distortion were added at various levels to simulate the effects of real data. One last position of the pattern was created separately from the training set for testing purpose. The above procedure was repeated 10 times and the results were averaged over all trials.

The amount of noise and distortion added varied through the entire experiment. However, even when the amount of noise – the standard deviation of a white (Gaussian) noise – is 2 pixels, the algorithm still performs very well, with less than 1cm of error in 3D reconstruction. Also, in order to contrast the algorithm with a bad scenario in which the images for calibration are not appropriately selected, we compared the performance of the calibration using images that scored poorly. Figures 2(a) and (b) show the error in 3D reconstruction as a function of the noise. The error is calculated as the difference between the estimated (reconstructed) coordinates of the test points and the ground truth. Figure 2(a) shows the simulation performed without adding radial distortion, while for 2(b), a typical radial distortion (from the real lenses) of 0.3 was added.

We also tested our algorithm with **real** data. Using the calibration parameters obtained using the proposed algorithm and the pixel coordinates of a set of predefined points in space as perceived by all 6 cam-

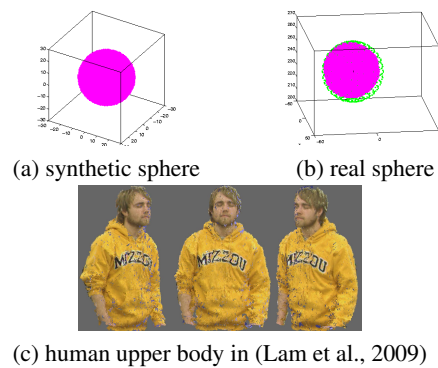


Figure 3: 3D Reconstructions.

eras, we reconstructed the spatial coordinates of these points and compared the calculated values with the real ones.

The calibration error was measured by averaging the results for 20 different snapshots while presenting the reference points to all cameras. The reference points were exactly 50cm apart. Each snapshot was taken by all 6 cameras, so a total of 120 images were used for this test. The accuracy of the final calibration was determined by calculating the distance between two reference points. The accuracy in 3D reconstruction was 50.6264cm – or less than 1.5% of the actual distance. Also, the small standard deviation (0.2498cm) shows that the calibration obtained with our algorithm gives a very consistent 3D reconstruction. It is important to mention that, for the current baseline of the cameras, a deviation of a single pixel in the location of the marks on the ruler already incurs on an error of almost 3mm in the reconstruction.

4 3D OBJECT MODELING

Since the main application for our camera rig is for 3D object modeling, we tested the accuracy of the calibration by reconstructing a sphere based on the idea of visual hull (Laurentini, 1994) and the human body using an algorithm for multi-view 3D modeling presented in (Lam et al., 2009). As before, the tests were conducted for both synthetic data and real data.

In the first case, synthetic data, we created multiples of 6 virtual cameras (6, 12, 18, ...) arbitrarily positioned around the object. By utilizing intrinsic and distortion parameters, we synthesized the images of a sphere (ball) with 20cm of radius. Using the voxel carving approach (Dyer, 2001), we reconstructed the sphere and measured its radius. In simple terms, this approach consists of: 1) defining a set of voxels in the 3D space; 2) marking each voxel according to the occupancy of the object as projected onto each of the

Table 1: Estimated radii vs. the number of virtual cameras.

# of virtual cameras	6	12	18	24	30	36	42
estimated radius (cm)	21.53	21.35	21.00	20.43	20.25	20.15	20.15

$6n$ image planes; and 3) finding the intersection of the occupancy for all $6n$ cameras. Table 1 shows the relationship between the reconstructed radii versus the number of cameras used for reconstruction. As expected, the error decreases as the number of cameras increases. That is because, among other reasons, the occupancy defined by each camera view forms a cone in space and the intersection of any subset of camera views approximates the sphere by the surface of such cones. Every time a new camera is added to the subset, the approximation becomes closer to the actual shape of the sphere. Since this procedure also relies on a sphere-fitting algorithm to circumscribe the occupied voxels, the detected radius tends always to be larger than the actual radius. For the real data, six cameras were used to take images of a ball. For each image, a circular Hough transform was used to detect the boundary and the 2D radius of the ball. As before, we relied on a voxel carving approach to reconstruct the ball. Figures 3(a) and (b) depict the reconstructed sphere for both synthetic data and real data. For the real ball, also with 20cm of radius, a 3D sphere was fitted and the radius was estimated. The performance of the framework using real cameras was 24.3cm. Finally, Figure 3(c) depicts the result from our multi-view algorithm for 3D modeling.

5 CONCLUSIONS

We have presented a novel method for autonomous camera calibration of a multi-camera rig. The experimental results showed that the algorithm is vital in order to obtain good 3D reconstruction. That is, the algorithm's selection of the best images for calibration leads to an improved calibration of as much as ten times of that obtained without using the algorithm. Finally, an application for the camera rig was presented where a sphere was placed in the middle of the rig and a 3D representation of the same sphere was constructed with an error in reconstruction (real camera) approaching the theoretical (synthetic cameras) error.

REFERENCES

Baker, A. and Aloimonos, Y. (2000). Complete calibration of a multi-camera network. In *Proceedings of IEEE International Workshop on Omnidirectional Vision*. IEEE Computer Society.

Chen, K., Hung, Y., and Chen, Y. (2005). Calibrating a camera network using parabolic trajectories of a bouncing ball. In *Proceedings of IEEE International Workshop on VS-PETS*. IEEE Computer Society.

Dyer, C. (2001). Volumetric scene reconstruction from multiple views. In *Foundations of Image Understanding*. Kluwer.

Floyd, R. (1962). Algorithm 97: Shortest path. *Commun. ACM*, 5(6):345.

Han, K. and DeSouza, G. N. (2007). A feature detection algorithm for autonomous camera calibration. In *Proceedings of Fourth International Conference on Informatics in Control, Automation and Robotics*.

Huang, Z. and Boufama, B. (2002). A semi-automatic camera calibration method for augmented reality. In *Proceedings of IEEE International Conference on System, Man and Cybernetics*. IEEE Computer Society.

Jaynes, C. (1999). Multi-view calibration from planar motion for video surveillance. In *Proceedings of IEEE International Workshop on Visual Surveillance*. IEEE Computer Society.

Koller, D., Klinker, G., Rose, E., Breen, D., Whitaker, R., and Tuceryan, M. (1997). Automated camera calibration and 3d egomotion estimation for augmented reality applications. In *Proceedings of the 7th International Conference on Computer Analysis of Images and Patterns*. Springer-Verlag.

Lam, D., Hong, R., and DeSouza, G. (2009). 3d human modeling using virtual multi-view stereopsis from on-the-fly motion estimation. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE Computer Society.

Laurentini, A. (1994). The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Olsen, B. and Hoover, A. (2001). Calibrating a camera network using a domino grid. *Pattern Recognition*, 34.

Remagnino, P. and Jones, G. (2002). Registration of surveillance data for multi-camera. In *IEEE International Conference on Information Fusion*. IEEE Computer Society.

Svoboda, T., Martinec, D., and Pajdla, T. (2005). *A convenient multicamera self-calibration for virtual environments*. MIT Press, Cambridge.

Yamazoe, H., Utsumi, A., and Abe, S. (2006). Multiple camera calibration with bundled optimization using silhouette geometry constraints. In *Proceedings of the 18th International Conference on Pattern Recognition*. IEEE Computer Society.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

STOCHASTIC OPTIMIZATION FOR ENVIRONMENTALLY POWERED WSNs USING MDP MODELS WITH MULTI-EPOCH ACTIONS

Alexandru E. Şuşu

EPFL, Station 14, 1015 Lausanne, Switzerland

alex.susu@gmail.com

Keywords: Wireless Sensor Nodes, Energy Harvesting, Constrained Markov Decision Processes, Multi-epoch Actions.

Abstract: The controller of an environmentally powered wireless sensor node (WSN) seeks to maximize the quality of the data measurements and to communicate frequently with the network, while balancing the uncertain energy intake with the consumption. To devise such system manager we use the Markov Decision Process (MDP) optimization framework. However, our problem has physical characteristics that are not captured in the standard MDP model: namely, the radio interface takes a non-negligible amount of time to synchronize with the network before starting to transmit the acquired data, which translates into MDP actions spanning over multiple epochs. Optimizing without considering this multi-epoch actions requirement results in sub-optimal MDP policies, which, under certain conditions described in the paper, waste on average 50% of the radio activity. Therefore, we incorporate this new constraint in the MDP formulation, and obtain an optimal policy that performs on average 83% better than a standard MDP policy. This solution outperforms also some heuristic policies we use for comparison by 14% and 154%.

1 INTRODUCTION

Advances in microelectronic technology allow us to build low-cost and low-power miniaturized Wireless Sensor Nodes (WSN), which can sense and transmit the information. Such devices seek to attain a good quality of service, while functioning for a long duration. Therefore, a series of energy management techniques are proposed in the literature (Anastasi et al., 2009) to reduce the power consumption of the sensor node, among which the most notable one is duty cycling the activity of the components of the node, such as powering on and off the radio transceiver periodically. Another possibility is to harvest and use the energy from the environment (Paradiso and Starner, 2005). Even if the harvesters can ensure a theoretically unlimited amount of energy over time, the power they provide is unpredictable. We address this issue by using power storage elements, such as rechargeable batteries or supercapacitors, in order for the system to have energy when it is not available from the harvester. However, these buffers are finite, and, therefore, they cannot completely hide the unreliability of the energy source, for example, when the harvester is not generating energy for a long period of time.

Concrete examples of environmentally powered

sensor nodes are found in the literature: solar powered devices (Moser et al., 2008; Jiang et al., 2005; Dubois-Ferrière et al., 2006), thermal powered (Gyselinx et al., 2005) and vibrational ones (Roundy et al., 2005).

To motivate the novelty of the paper, we focus on a particular characteristic of these energy harvesters. While the solar radiation and, together with it, the energy output of a photovoltaic panel, have normally a slow variation over the range of hours, an eolian harvester can experience fast variations, in the order of seconds (Twidell and Weir, 1986). We call the former type a slow-dynamics harvester, and the latter a high-dynamics one.

In this paper, we consider a wireless sensor node powered by such a high-dynamics harvester, which uses eolian energy. The node runs a reactive application, which senses and transmits wirelessly the data to a basestation at a specified rate. This rate is normally in the same order of magnitude with the frequency of variation of the harvested energy. Managing such a device is a novel contribution and, due to the fast temporal properties of the harvester, it raises new constraints that we need to take into consideration.

Our goal is to devise a sensor node controller that maximizes the number of measurements of the physical property in the time unit (the sensing duty cycle)

and transmit the data to the basestation at the specified rate. Similar optimization formulations can be found in (Kansal et al., 2004), which adapts the system duty cycle in order to match the energy profile intake, and in (Niyato et al., 2007), which computes policies that optimize certain Quality of Service (QoS) metrics. Differently from them, we perform multi-criteria optimization, having two types of commands to manipulate the sensing and the radio transceiver duty cycle.

Since, in principle, we also want to have a reduced production cost, in this paper we consider a sensor node system with: i) an energy harvester device that generates on average the required energy level, since this impacts the production cost; ii) a small sized energy storage element - the size of the buffer is determined by the longest "blackout" period. We argue that the problem we present is useful when the system has little energy in the storage element, when prediction of the energy intake can bring great benefit.

The contributions of the paper are: i) we build a representative Markov chain model for the eolian harvester powering the node; ii) we reduce the sensor node control problem to an average reward Constrained Markov Decision Process (MDP) formulation, one of the most complex planning problems; iii) then, we introduce in the optimization problem the proposed multi-epoch action constraints, relevant for our setting; iv) we solve the problem rigorously using various tools we developed on top of existing software, and compare the benefits of our method to some simple heuristic policies we introduce.

2 DESCRIPTION OF THE SYSTEM MODEL AND OF THE OPTIMIZATION PROBLEM

As advocated in (Şuşu et al., 2008; Poggi et al., 2000), an accurate model of the energetic sources is essential for evaluating the system's average productivity or the availability, and, in principle, to ensure the system's management. Simulation provides results only for the period over which environmental energy data is available. Since the results are different if we use other time series with the same statistical properties, we are interested to know the range of these results. Therefore, finding a representative model able to capture the uncertainty of the energy source requires attentive thinking. (Poggi et al., 2000) proposes a first-order stationary Discrete Time Markov Chain (DTMC) model for each month of the year, due to the big monthly variations, built from traces taken over a period of 20 years. In a similar direction

goes (Nfaoui et al., 2003) for wind speed measurements.

Similarly, in this paper we use an offline built first-order DTMC model for an eolian energy harvester using wind speed traces collected by the SensorScope project (Barrenetxea et al., 2006) from EPFL. Since we do not perform experiments with a real device, the model for the energy harvester is using simplifying assumptions such as the energy produced by an eolian harvester is directly proportional to the wind speed. Therefore, in this paper we do not put accent on an end-to-end treatment of this problem from theory to full implementation, but focus mostly on the modeling and optimization part.

The method described in this paper is general in the sense it can be applied in settings using other forms of environmental energy, which can be represented by Markovian models.

The system has a dedicated controller that observes the parameters of the node and controls it in order to optimize its functionality. A command specifies the sensing duty cycle (e.g., with values 0%, 50%, 100%) and the power state of the radio transceiver. Our modeling problem uses a discrete time setting: the controller is invoked at each $time_step = 1\ second$ period (or epoch).

We assume, after studying the real and simulated values of our sensor node platform, that the energy consumed by the operation of the radio transceiver in an epoch is 40 mJ. The energy consumed in an epoch by the sensing equipment and the microcontroller is a multiple of 10 mJ, proportional to the duty cycle. We also assume the sensor node has a small energy storage element of 1,000 mJ. For our model, we assume that a unit of energy represents 10 mJ.

2.1 The MDP Model

As already discussed, we formulate this control problem on a discrete time MDP model, \mathcal{M} , which is defined by: i) a finite set of reachable states (under any policy), \mathcal{S} ; ii) a finite set of actions, \mathcal{A} ; iii) an initial probability distribution over \mathcal{S} , p^1 ; iv) a transition probability matrix for each action, represented by the function $p : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ denoting the probability to transition at a destination state from a source state if using a specific action; v) two reward functions $r_R, r_{SDC} : \mathcal{S} \rightarrow \mathbb{R}$, which we define below.

For our setting, \mathcal{M} is obtained through the parallel composition of the system modules: the energy harvester, the energy buffer, the node (with the sensing and radio components) and the controller, as depicted in Figure 1. The MDP actions are the already discussed commands that control the sensing duty-

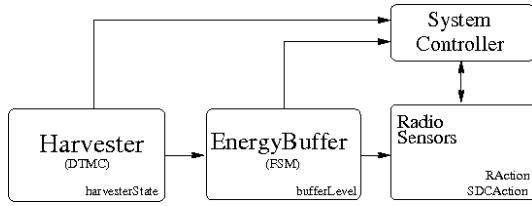


Figure 1: Behavioral black-box model of the system.

cycle and the radio. The duration of an epoch takes $time_step$ units.

MDP States and Actions: Two of the state variables of the system are $bufferLevel$, which represents the energy stored in the energy buffer at each epoch and $harvesterState$, which describes the energy output of the harvester in a period. The action variables, which are also state variables for the MDP model, are $SDCAction$, which specifies the sensing duty cycle of the node, and $RAction$, which represents the power command to the radio transceiver.

An MDP state $s \in \mathcal{S}$ is defined by the tuple of values of these four variables. Also, an MDP action $a \in \mathcal{A}$ is completely defined by the values of $SDCAction$ and $RAction$ in the following epoch.

MDP Rewards: To express the quality of the sampled data by the node, we add to each MDP state s the reward $r_{SDC}(s)$, which is a function of the sensing duty cycle. We use in this paper a concave function: 0.0 for 0% duty cycle (for $SDCAction = 0$), 8.0 for 50% duty cycle ($SDCAction = 1$) and 10.0 for 100% duty cycle ($SDCAction = 2$).

As already discussed, we can put a soft real time constraint on the functionality of the node: we want to have a certain average period, which we call $T_{latency}$, of transmitting via radio the acquired data to the basestation. We do this in order that the basestation continuously benefits of sensed data that is about $T_{latency}$ seconds old.

To account for the frequency of sending via radio the data we put on each state s the reward $r_R(s)$ with value 1.0 if s employs the radio transceiver ($RAction = 1$) and 0.0 if not ($RAction = 0$).

We also do not want to allow the possibility of issuing an action that results in running out of energy without terminating the initiated action. Therefore, we put big negative rewards on such states, such that the solution MDP policy avoids them. This makes sense since the solution either maximizes the objective function value, or keeps the value of the constraint above a specified threshold and using any number of big negative reward states in the solution is not compatible with these goals. For the simulations we perform in Section 3.1, if we run out of power in an

epoch (i.e, $bufferLevel' < 0$) we consider the sensing duty cycle reward and the radio reward to be zero.

Semantics: The precise semantics of the MDP model is the following: i) we start with $RAction = 0$ and $SDCAction = 2$ (the radio transceiver is turned off and the node senses with maximum duty cycle), and with $bufferLevel$ set at maximum; ii) the energy intake happens at the beginning of the epoch, instantaneously, and the consumption happens immediately afterwards during the same epoch, by executing the actions for $RAction$ and $SDCAction$, associated to the state; iii) the controller observes the values of the system variables at the beginning of the epoch and computes the values of $RAction'$ and $SDCAction'$ for the next epoch.

2.2 The Optimization Problem

As we have previously stated, we seek to maximize the sensing MDP reward, while meeting an average radio transmission rate of $1/T_{latency}$.

The solution to our problem is a Markov randomized stationary policy $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})$, which prescribes for each state the optimal probability distribution over the actions the controller has to use in order to attain the problem objectives.

The mathematical formulation of our problem is given in (1). $T_{latency}$ is a constant representing the number of epochs the radio needs to transmit, on average, during a $T_{latency}$ period, for the time being, equal to 1.

$$\begin{aligned} \max_{\pi} \lim_{N \rightarrow \infty} \frac{1}{N} E^{\pi} \left\{ \sum_{i=0}^{N-1} r_{SDC}(i) \right\} \\ \text{s.t. } \lim_{N \rightarrow \infty} \frac{1}{N} E^{\pi} \left\{ \sum_{i=0}^{N-1} r_R(i) \right\} \geq \frac{T_{latency}}{T_{latency}} \end{aligned} \quad (1)$$

We can prove our MDP is unichain. In such a case, (Puterman, 1994) arguments we can reduce this optimization problem to the Linear Program (LP) in (2). The variables of the LP, $x_{s,a}$, are called limiting average state-action frequencies and represent the probability the system occupies state s and chooses action a under the solution policy corresponding to x .

$$\begin{aligned} \max \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} x_{s,a} r_{SDC}(s) \\ \text{s.t. } \sum_{a \in \mathcal{A}} x_{s,a} - \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} p(s'|s,a) x_{s',a} = 0, \forall s \in \mathcal{S}, \\ \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} x_{s,a} = 1, \\ \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} x_{s,a} r_R(s,a) \geq \frac{T_{latency}}{T_{latency}}, \\ x_{s,a} \geq 0, \forall s \in \mathcal{S}, a \in \mathcal{A} \end{aligned} \quad (2)$$

```

MDP
const int PSEMAX = 100; //the capacity of the energy storage element
                        //(1 unit = 10 mJ)
module Harvester
  harvesterState: [0..9] init initHarvesterState;

  [tick] harvesterState = 0 -> 0.927 : (harvesterState' = 0)
        + 0.064 : (harvesterState' = 1) + ...;
  //...
  //the rest of the description of this module is omitted
endmodule

formula update_bufferLevel = bufferLevel + harvesterState - SDCAction
- 4*RAction;

module EnergyBuffer
  bufferLevel:[0..PSEMAX] init PSEMAX;

  [tick] update_bufferLevel > 0 & update_bufferLevel <= PSEMAX ->
    (bufferLevel' = update_bufferLevel);
  [tick] update_bufferLevel > PSEMAX -> (bufferLevel' = PSEMAX);
  [tick] update_bufferLevel <= 0 -> (bufferLevel' = 0);
endmodule

//A generic system controller: see both RadioController and
SDCController
module RadioController
  RAction:[0..1];

  [tick] true -> (RAction' = 0);
  [tick] true -> (RAction' = 1);
endmodule

module SDCController
  SDCAction:[0..2];

  [tick] true -> (SDCAction' = 0);
  [tick] true -> (SDCAction' = 1);
  [tick] true -> (SDCAction' = 2);
endmodule

```

Figure 2: The PRISM MDP model of the system.

3 PRACTICAL DETAILS OF THE PROBLEM

To concretely solve the problem, we start by specifying the model in PRISM, a probabilistic analysis tool (Kwiatkowska et al., 2004) we use extensively in our project. A simple model example with a freely specified system controller, the unknown of our problem, is given in Figure 2.

We develop some small programs that take the MDP representation exported from PRISM and output the corresponding LP that solves the MDP optimization problem. To make the problem more precise and robust, the coefficients of the LP have 16 decimal digits precision. We solve the LP with CPLEX (IBM ILOG, 2008), a fast commercial (I)LP solver, which is able to handle LPs up to tenths of thousands of variables.

The capacity of the energy storage element, $PSEMAX$, is 100 and $T_{lateness}$ is 10. We also assume there is only one initial state, the one with a full energy storage element, with the maximum level of harvested energy, with $RAction = 0$ and $SDCAction = 2$.

Once the MDP policy is obtained, to assess that it conforms to the specification - mainly that the ex-

pected radio reward is met - but also to compute various other properties such as the value of the expected sensing reward, we compose the solution policy with the PRISM specification of the MDP and perform exhaustive simulation (i.e., probabilistic model checking) by using the cumulative reward properties of PRISM to compute the expected total rewards for a specified number of steps, and then compute the average reward per epoch.

Since we want to show the benefits of implementing a stochastic policy controller, we introduce two more runtime policies, which are deterministic and compare the three of them in the following sections. The three different controllers are:

- **MDP:** this is the stochastic policy, which uses the energy harvester model to accurately predict the future energy income. Relying on the current value of $bufferLevel$ and on the prediction, the policy chooses the optimal sensing duty cycle, $SDCAction'$, and the power state for the radio, $RAction'$.
- **Conservative:** the policy varies the sensing duty cycle proportionally with the $bufferLevel$. Also, the policy controls the radio the following way: in the case $bufferLevel$ is too small, the controller assumes the worst-case situation under which it does not receive anymore energy from the environment, and turns off the radio; otherwise, the radio is turned on, periodically.
- **Greedy:** the policy uses the maximum sensing duty cycle, no matter what the value of $bufferLevel$ is. The node tries to turn the radio on and to send data to the basestation periodically, at every $T_{lateness}$ seconds, disregarding the fact it might run out of power.

The Conservative and Greedy policies do not make use of the probabilistic harvester model. The Greedy controller does not care about the state of the energy harvester and observes, at most, only the charge of the energy buffer. In all the experiments we perform in the following sections, we count for the radio and sensing rewards of the policies only the epochs in which the system does not run out of energy.

3.1 Considering The Multi-epoch Radio Transceiver Actions

So far, we did not take into consideration that turning on the radio transceiver of the sensor node followed by transmitting the acquired data are costly operations, which can take longer than one epoch

```

module RadioController
  RAction:[0..1];
  counter:[0..T_LATENESS - 1];

  [tick] counter = 0 -> (RAction' = 0) & (counter' = 1);
  [tick] counter > 0 & counter <= T_LATENCY - 1 & RAction = 0 ->
    (RAction' = 0)
    & (counter' = counter + 1);
  [tick] counter = 0 -> (RAction' = 1) & (counter' = 1);
  [tick] counter > 0 & counter <= T_LATENCY - 1 & RAction = 1 ->
    (RAction' = 1) & (counter' = counter + 1);
  [tick] counter >= T_LATENCY & counter < T_LATENESS - 1 ->
    (RAction' = 0) & (counter' = counter + 1);
  [tick] counter = T_LATENESS - 1 -> (RAction' = 0) & (counter' = 0);
endmodule
    
```

Figure 3: The PRISM radio controller specification with $T_{latency}$ -epochs long actions.

because the node has to synchronize with the network. For the SensorScope platform, which uses the BMAC protocol, the time the radio needs to synchronize to the network and send the data is $T_{latency_exact} = 2.25 \text{ seconds}$ (Şuşu et al., 2008). For the sake of the energy efficiency, since the radio is the most power consuming component of the sensor node, we do not want to turn it off while synchronizing or transmitting. Therefore, under these conditions, the radio power off command should be issued at least $T_{latency} = \lceil \frac{T_{latency_exact}}{time_step} \rceil = 3$ epochs later from the moment we turned on the radio.

In order to handle the optimization with multi-epoch actions, we add a new state variable, *counter*, to the PRISM model, which keeps track of how many epochs have passed from the beginning of the current $RAction = 1$ issue. We then allow to change to $RAction = 0$ only if $counter \geq T_{latency}$.

Adding this variable gives us the possibility to turn on the radio only at the beginning of the $T_{lateness}$ period for exactly $T_{latency}$ periods. This implies that the counter variable has to take values between 0 and $T_{lateness} - 1$. The PRISM specification of the radio controller with the variable *counter* is given in Figure 3. We compose this component with the others of the model in Figure 2. The radio constraint in our formulations (1) and (2), specifying to have at least $T_{latency}/T_{lateness}$ average radio reward per epoch, is in accordance with the behavior described above.

4 OPTIMAL SOLUTION AND COMPARISON

We call π_3 the MDP optimal policy that generates only $T_{latency}$ -epochs long radio actions ($T_{latency} = 3$) and π_1 the one on which we do not impose this constraint.

We present in Table 1 the simulation results obtained with PRISM for π_3 , together with the ones for the Conservative and the Greedy policies, for which

Table 1: The expected total radio (R) and sensing (SDC) rewards of the solution policies for constrained optimization, for various horizon length (hl) values.

hl	MDP policy π_3		Conservative policy		Greedy policy	
	R	SDC	R	SDC	R	SDC
10^4	3,099	45,587	1,945	61,768	1,031	52,727
10^5	30,822	453,302	19,395	615,398	10,192	524,071
10^6	308,055	4,530,441	193,893	6,151,698	101,809	5,237,512

we present only the radio rewards that are $T_{latency}$ -epochs (or more) long. As we can see, the π_3 policy has average rewards per epoch of 0.30 for the radio and 4.53 for sensing. π_1 attains on average per epoch rewards of 0.30 for the radio (out of which only 49.57% satisfies the $T_{latency}$ constraint) and 4.96 for sensing. To compare π_1 and π_3 we use the product between the expected $T_{latency}$ -feasible radio reward and the expected sensing reward. This product is 0.744 for the former and 1.359 for the latter, which means π_3 performs with 83% better than π_1 . Using the same metric, π_3 is better with 14% than the Conservative policy and with 154% than the Greedy one. The Greedy policy is the only one that runs out of energy, about 1.73% of the time.

The MDP has 39,390 reachable states. The associated LP uses six times more variables and takes more than six days to be solved exactly on a standard computer platform. Therefore, we use competitive approximation methods, the details of which we omit in this paper, to find the solution, which reduce the search time to a couple of hours.

The optimal policy π_3 prescribes for each reachable state of the system the best sensing duty cycle and radio management that maximizes the expected sensing quality, generates only $T_{latency}$ -epochs long radio transmissions and does not run out of energy, for the given harvester DTMC model.

5 CONCLUSIONS

In this paper we have modeled and improved the functionality of a wireless sensor node with Markov Decision Processes (MDPs). Because of the long time the radio transceiver takes to synchronize with the network, we have introduced MDP actions that take longer than one epoch to complete. Optimizing without taking into consideration these multi-epoch actions results in suboptimal MDP policies. We proposed a method to find an optimal solution and compared the result to various heuristic policies.

Our problem with multi-epoch actions has some similarities with the Semi-Markov Decision Process

(SMDP) (Puterman, 1994; Hu and Yue, 2007) models with variable sojourn times. Note that our setting is even more different, since we have action types with different completion times.

REFERENCES

- Anastasi, G., Conti, M., Di Francesco, M., and Passarella, A. (2009). Energy conservation in wireless sensor networks: A survey. *Ad Hoc Netw.*, 7(3):537–568.
- Barrenetxea, G., Dubois-Ferriere, H., Meier, R., and Selker, J. (2006). A weather station for SensorScope. In *Demo Session, In Information Processing in Sensor Networks (IPSN 2006)*.
- Dubois-Ferrière, H., Meier, R., Fabre, L., and Metrailler, P. (2006). TinyNode: A Comprehensive Platform for Wireless Sensor Network Applications. In *Information Processing in Sensor Networks (IPSN 2006)*.
- Gyselinx, B., Hoof, C. V., Ryckaert, J., Yazicioglu, R. F., Fiorini, P., and Leonov, V. (18-21 Sept. 2005). Human++: Autonomous wireless sensors for body area networks. In *Custom Integrated Circuits Conference, 2005. Proceedings of the IEEE 2005*, vol., no.pp. 13-19.
- Hu, Q. and Yue, W. (2007). *Markov Decision Processes with Their Applications*. Advances in Mechanics and Mathematics, v. 14. Springer, Dordrecht.
- IBM ILOG (2008). ILOG CPLEX User’s Manual.
- Jiang, X., Polastre, J., and Culler, D. E. (2005). Perpetual environmentally powered sensor networks. In *IPSN*, pages 463–468.
- Kansal, A., Potter, D., and Srivastava, M. B. (2004). Performance aware tasking for environmentally powered sensor networks. *SIGMETRICS Perform. Eval. Rev.*, 32(1):223–234.
- Kwiatkowska, M., Norman, G., and Parker, D. (2004). Prism 2.0: A tool for probabilistic model checking. *QEST*, 00:322–323.
- Moser, C., Thiele, L., Brunelli, D., and Benini, L. (2008). Approximate control design for solar driven sensor nodes. In *HSCC '08: Proceedings of the 11th international workshop on Hybrid Systems*, pages 634–637, Berlin, Heidelberg. Springer-Verlag.
- Nfaoui, H., Essiarab, H., and Sayigh, A. (2003). A stochastic Markov chain model for simulating wind speed time series at Tangiers, Morocco. *Renewable Energy* 29 1407-1418.
- Niyato, D., Hossain, E., and Fallahi, A. (2007). Sleep and wakeup strategies in solar-powered wireless sensor/mesh networks: Performance analysis and optimization. *IEEE Transactions on Mobile Computing*, 6(2):221–236.
- Paradiso, J. A. and Starner, T. (2005). Energy scavenging for mobile and wireless electronics. *Pervasive Computing, IEEE*, 4(1):18–27.
- Poggi, P., Notton, G., Muselli, M., and Louche, A. (2000). Stochastic study of hourly total solar radiation in Corsica using a Markov model. *International Journal of Climatology, Volume 20, Issue 14, Pages 1843 - 1860*.
- Puterman, M. L. (1994). *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley-Interscience.
- Roundy, S., Leland, E. S., Baker, J., Carleton, E., Reilly, E., Lai, E., Otis, B., Rabaey, J. M., Wright, P. K., and Sundararajan, V. (2005). Improving power output for vibration-based energy scavengers. *Pervasive Computing, IEEE*, 4(1):28–36.
- Şuşu, A. E., Acquaviva, A., Atienza, D., and Micheli, G. D. (2008). Stochastic modeling and analysis for environmentally powered wireless sensor nodes. *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops, 2008. WiOPT 2008. 6th International Symposium on*, pages 125–134.
- Twidell, J. W. and Weir, A. D. (1986). Renewable energy resources.

A FRAMEWORK BASED ON A HIGH CONCEPTION LEVEL TO GENERATE CONFIGURATIONS IN PRODUCTION SYSTEMS

Marwa Kanso, Pascal Berruet and Jean-Luc Philippe
Lab-STICC, UBS Saint Maude, Lorient, France
{marwa.kanso, pascal.berruet, jean-luc.philippe}@univ-ubs.fr

Keywords: Reconfigurable manufacturing systems (RMSs), Model Engineering, Scheduling, Configuration, Recovery & reconfiguration.

Abstract: In this paper, a framework is presented to generate configurations in reconfigurable manufacturing systems (RMS). This framework is based on a high conception level of reconfigurable manufacturing systems, and uses a multicriteria decision algorithm to select operations to carry out the request. Provisional product scheduling is then used to define the “simple configuration”. This paper focuses on the generation process of the “near-most appropriate” configuration to carry out a request. To do so, the same multicriteria decision algorithm is used to select “reserved operations” to improve the generated configuration. A case of study illustrates our approach and demonstrates how we can use this framework to generate the “near-optimal” configuration and improve reconfigurability.

1 INTRODUCTION

Manufacturing flexibility is the key for markets facing increasing client demands, frequent volume changes and product requirement. In such a system, choices at the system organization level can be delayed until exploitation. Indeed, this organization must be taken into account at the conception level. To help a designer choose (resources, operations, ...), a set of analysis and evaluations must be applied to the system. The system description must specify all possible information for this analysis. New scheduling techniques which help to answer more complex demands are needed. On the other hand, the main goal in today's markets is the reaction of manufacturing systems when unexpected events occur. Recently, new ideas related to design, description, sequencing products, planning, loading and scheduling policies have been introduced. Architecture design for manufacturing systems, equipped with the correct level of flexibility to face the specific production problem, are introduced in (Nucci and Grieco, 2008) and in (Terkaj et al., 2009). (Kurnaz et al., 2005) considers that the order in which products are produced can have a considerable impact on primary performance metrics. Sequencing decisions can impact customer responsiveness. In (Dpto et al., 2002), loading and scheduling processes evolves jointly in Flexible Manufacturing Cell (FMC). (Lamotte, 2006) proposes DeSyRe,

a language to describe reconfigurable manufacturing systems. In this paper, we propose improving the DeSyRe language to take into account and improve the reconfiguration concept at the description level. We also consider how disruptions (machine breakdowns, customer order changes, etc.) impact the generated configuration in a reconfigurable manufacturing system producing multiple products. The rest of this paper is organized as follows. Section 2 gives an extended presentation of the high conception level of RMS's. Section 3 presents the configuration generation process. A use case illustrates the use of our framework in section 4, before moving on to the conclusion in section 5.

2 A HIGH CONCEPTION LEVEL FOR RMS

Manufacturing systems can generally be described from multiple levels of granularity. Fractal manufacturing systems (Ryu et al., 2003) takes into account this granularity by using a multi-level representation. To represent and manipulate reconfigurable manufacturing system concepts, we use a description language called “DeSyRe” basically developed and introduced by (Lamotte, 2006). Metamodels show relations between different aspects of a RMS. Models

combine a horizontal decomposition between architecture and configuration and a vertical one between logical and physical descriptions. We apply this high level conception to add new dynamic aspects and improve the RMS description. In the next subsections, we present the extended description language version “DeSyRe_E”.

2.1 The represented Aspects

In “DeSyRe_E”, the reconfigurable system is broken down according to the decomposition illustrated in Figure 1. A horizontal axis separates the architecture of the system from its configuration. The architecture consists of all system elements (functions, or resources) and their potential connections. It is presented in section 2.2. These components are parameterized and inter-connected through the configuration presented in section 2.3. The vertical axis separates the logical architecture from the physical one. The logical part consists in the functions and their associations to form logical operating sequences. The physical part consists in the resources and the transport between these resources. The physical part provides the structure on which the logical part is executed. Operations are in the center of the model description. Each operation implements a function on a resource creating a link between the logical and the physical architecture. It also links the configuration to the architecture since operations are defined in the architecture and used in the configuration.

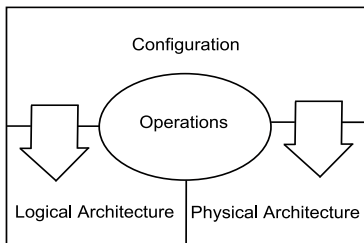


Figure 1: The new decomposition of RMSs.

2.2 A Meta-model to represent Architectures

The architecture of a reconfigurable manufacturing system should represent all system potentialities (see figure 2). It is separated into two parts: the logical architecture, which is constituted by functions and function sequences applied to products that can be performed on the system, and the physical architecture which describes the physical elements of the system (resources, connections, ports, etc...). To

achieve a complete architecture representation, operating modes of the whole system should be provided. The role of such representation is to help carry out decisions about resource modes and configuration changing. A lot of information is needed to manage resource modes, see (Hamani et al., 2009), and perform performance and cost analysis. This information is introduced using a simple diagram graph for each resource to describe their modes and transitions between these modes. For the sake of simplicity, here, we propose to take into account the following modes: the mode “stop” when the power is off, the mode “in production” when the resource is in use, the mode “in preparation” when the resource is not in use, nor ready to execute an operation, the mode “break down” when a failure occurs on the resource and the mode “idle” when the power is on and the resource is ready to use.

Logical architecture is mapped onto physical architecture through potential operations, which link a function with a resource.

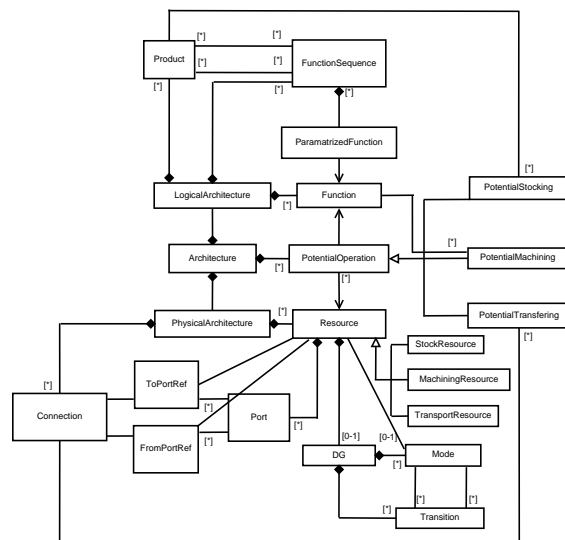


Figure 2: Extended architecture metamodel.

2.3 A Meta-model to represent Configurations

The concept of configurations lay on the separation of the component content from its use in the architecture. A configuration is constituted by function instances to realize a machining operation, connection instances to realize a transfer operation, operations which refer to an operation defined in the architecture model and operation sequences which realize a function sequence. The metamodel representing the configuration is given in figure 3.

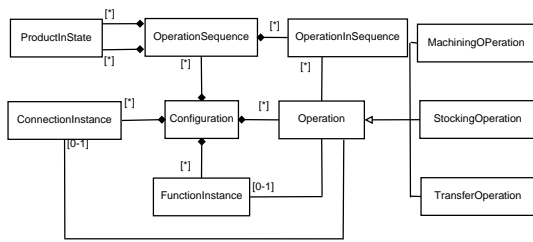


Figure 3: Extended configuration metamodel.

Several types of configurations are defined depending on the use of architecture components. A minimal configuration contains the needed operations to carry out a mission. A simple configuration is defined as the set of “used” operations which lead to the completion of a mission. A flexible configuration is defined by a set of “used” and “reserved” operations to simplify the reconfigurability aspect. Finally, a maximal configuration is defined as the set of potential operations which lead to carrying out a mission. In the next section we present the Configuration Generation module.

3 CONFIGURATION GENERATION

As can be seen in the figure 4, three technologies are used: Model driven engineering (MDE), object oriented programming and multiCriteria Decision Making (MCDM). Modeling concepts are introduced to describe architectures and configurations. The Model Driven engineering approach (MDE) is then used, a transformation module is used to generate the maximal configuration from the architecture description model. This transformation is expressed in a transformation language such as ATL (Bézivin et al., 2005).

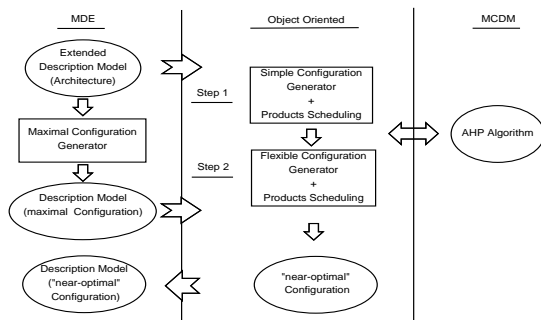


Figure 4: A framework to generate configurations.

Generated code is based on the architecture and maximal configuration description models. Object

oriented development (using Java) is then used to carry out the “near most-appropriate” configuration. At runtime, both steps (1 and 2) use the MultiCriteria Decision Making approach (MCDM) to solve indeterminism due to the high flexibility of our system and to carry out the request in the configuration. For more details about the MCDM approach see (Kanso et al., 2009). A simple configuration is generated in step 1. The obtained configuration contains only the “used” operations to carry out the mission. It means that, in case of failure there is no guarantee to replace the failed operation. Our framework tries to avoid the reconfiguration process as much as possible, by improving the generated configuration with a compatible degree of flexibility. We deal with two kinds of unexpected events: changes at the request level (case of customer demand changes) and changes at the system level itself (case of failures and system errors). When unexpected events occur, errors, failures or customer demand changes provide the need of additional operations to replace the failed ones. Operations in need are evidently defined in the architecture. To answer this need, a flexible configuration is generated in step 2. This last configuration consists of the simple configuration improved by adding “reserved” operations using the MCDM approach. The obtained configuration description model is then generated using MDE.

4 SAMPLE EXPERIMENTATION

We present, in this section, a simple example to demonstrate how our framework can be used to generate the “near optimal” configuration. The purpose of this experimentation is not to generate optimal suggestions, but rather to provide a simple demonstration to help users taking their decisions during production. To do so, we consider a simple reconfigurable manufacturing system with two products. The “logical architecture” contains two functions (F_1 and F_2) and three function sequences (S_1 , S_2 and S_3). S_1 consists in implementing F_1 . S_2 consists in implementing F_2 . S_3 consists in implementing F_1 then F_2 .

The “Physical Architecture” of the proposed system is defined to set up our experiments (see figure 5). It consists of conveyors ($CV_1 - CV_6$), buffers (IN and OUT), machining areas ($M_1 - M_4$) and a robot (Rbt_1). To increase the flexibility of the system, this architecture consists of the two-directional conveyors (CV_2 and CV_5). In the description model of the architecture, we refine, for each resource, its capacity, its active mode, the list of the concerned resource modes and the transitions between these modes.

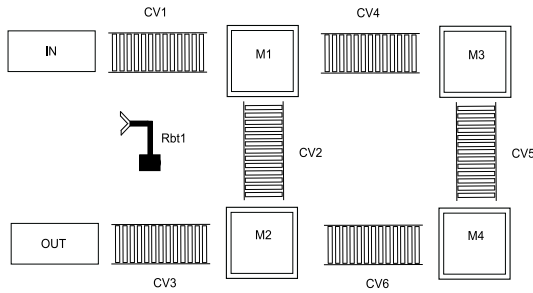


Figure 5: Physical Architecture of the proposed system.

Eight machining operations have been defined to assign the defined functions to the machining areas: $O_{M1,F1}$, $O_{M1,F2}$, $O_{M2,F1}$ and $O_{M2,F2}$, etc...

Fourteen transfer operations have also been defined to realize a transfer using a transfer resource: $O_{ICv1,INM1}$, $O_{IRbt,INM1}$, $O_{IRbt,INM2}$, $O_{ICv2,M1M2}$, $O_{ICv2,M2M1}$, etc...

4.1 Maximal Configuration Generation

In this subsection, we present the transformation module used to generate the “maximal configuration” using the architecture description model. Based on the specified mission (product types and quantities), some operations cannot be used in the system. The concerned resources are idle during production. To optimise the configuration cost, we must generate the “maximal configuration”. At this stage, we take into consideration the system part which contains the possible operations to carry out the request. If all potential operations can be used to realize our mission, the maximal configuration represents the use of the whole system. For reasons of simplicity, in our example, the maximal configuration represents the use of the whole system to carry out two different product types (P_4 and P_5). Each operation in the architecture model is projected to generate an operation in the configuration model. The link to a function instance or a connection instance in the configuration model is based on the type of the operation in the architecture. A function sequence can be realized by different operation sequences. For example, S_1 is the function sequence which consists of applying the function F_1 on a product. It can be realized on the machining resource M_1 , M_2 , M_3 or on M_4 . These four solutions are defined in the configuration model as a set of 88 operation sequences, since transfer may be realized by different transfer sequences too. Indeed, we generate 674 possible operation sequences based on the function sequences defined in the architecture model.

It should be noticed that by using all the operations and operation sequences of the system in the configuration at runtime, the reconfiguration time, in

case of failure can be shorten (it depends on calculation and decision making time), and the cost of the configuration is surely much higher since a lot of resources are idle. Therefore, we propose to generate the “near-optimal” configuration which answer a better cost-performance trade-off. In the next subsection, we refine the “Simple Configuration generation” process based on the architecture and the maximal configuration description models.

4.2 Simple Configuration Generation

The specified mission consists of producing three lots of P_4 and two lots of P_5 . Decisions, of which transporter (or machining resource) is used, are taken by applying the ‘AHP’ algorithm. For more details about the proposed algorithm please check (Kanso et al., 2009). The “Simple Configuration Generation” process consists of selecting, for each product instance, an operation sequence taken into account the system state (resource availability, waiting time, preparation time, mode changes, transfer between two different machining resources). We use the necessary information in the architecture to decide which is the most appropriate operation to realize next. This process has been implemented using Java in eclipse environment. Once the application is launched, the selection process starts. The architecture of the proposed system is updated depending on each resource capacity and modes in the system while executing an operation on a product. The “simple configuration” is obtained when each product instance specifies its operation sequence and its provisional scheduling. We define at this level the “used operation” as an operation used by a product to realize a function in a configuration. For the proposed example we get the configuration composed of the followed operation sequences:

1. to realize P_4 , S_1 must be applied. Therefore, the operation sequence chosen is as follows:
 - $OS_{1,S_1}: \{O_{ICv1,INM1}, O_{M1,F1}, O_{IRbt,M1OUT}\}$
2. to realize P_5 , S_3 must be applied. Therefore, the operation sequence choosed is as follows:
 - $OS_{12,S_3}: \{O_{ICv1,INM1}, O_{M1,F1}, O_{M1,F2}, O_{IRbt,M1OUT}\}$

4.3 Flexible Configuration Generation

In case of production problems and more precisely in case of failure when unexpected events occur (demand changes, resource breakdowns, unavailable tools, etc...). Previously generated simple configurations may no longer work. Operations used in the “simple configuration” may be unique and no another

operation, defined at the configuration level, can realize the same function. The obvious solution is to reconfigure the system. Although, we propose to improve the generated configuration and optimise the request completion time. Therefore, we need to extend the “simple configuration” and add “reserved operations”. At this level, we define a “reserved operation” as an operation chosen to replace a failed operation in a configuration. This will help us carrying out the request in the system without going through the reconfiguration process. To do so, “reserved operations” are selected using the ‘AHP’ algorithm. When the “reserved operations” are specified, we generate the corresponding operation sequences to fill out the generated configuration. In other words, we generate reserved operation sequences to extend the obtained “simple configuration” and get the “flexible configuration” to improve system responsiveness. Projecting this extension on our example, we get the “flexible configuration” by duplicating operations since each function (or transfer) can be realized using four resources ($M_1 - M_4$) for a function, and the robot or a conveyor for a transfer. We note that in our case the obtained “flexible configuration” corresponds to “162” operation sequence of the “maximal configuration” using two machining resources (M_1 and M_2) and “used operations” are realized in priority.

5 CONCLUSIONS

Reconfigurable Manufacturing Systems are faced with frequent disruptions that impact the production quality. A low-cost solution is needed to recover the system. In this paper, the use of a high conception level for reconfigurable manufacturing systems is presented to include different features and describe both architecture and configuration. The suggested framework produces “near-optimal” configuration by generating the corresponding “flexible configuration” based on the generated “simple configuration” and the architecture. A simple example illustrating the way we can use the proposed framework is presented in section 4. First, the production environment has been presented. Secondly, the “maximal configuration” generation is presented by refining the transformation module. Third, the “simple configuration” generation process is described before moving on to presenting the “flexible configuration” generation process. Configuration choices usually depend on preferences, on choices obtained at the scheduling level, and on constraints defined by clients at the request level. “Minimal” and “Simple” configurations are not always the best solutions. Configurations with

a higher degree of flexibility may respond to requests with better measures, and may answer a better cost-performance trade-off. Although the existing framework provides opportunity for many different types of analyses, additional extensions will be beneficial as well. These include inventory management during production and transfer sequence management; new configuration extension rules; and more specific criteria to improve operation choice. More generally, we would like to permit a broader class of configurations, such as serial parallel lines with crossover, so that the framework can be applied in an even greater number of circumstances.

ACKNOWLEDGEMENTS

The authors are pleased to acknowledge the support by the associate professor at Polytech’Marseille Fouzia Ounnar.

REFERENCES

- Bézivin, J., Jouault, F., Rosenthal, P., and Valduriez, P. (2005). The AMMA platform support for modeling in the large and modeling in the small.
- Dpto, A. G., Gmez, A., Fuente, D. D. L., Parreo, J., and Puente, J. (2002). Scheduling in flexible manufacturing systems. *Applied Artificial Intelligence*, 15:949–963.
- Hamani, N., Dangoumau, N., and Craye, E. (2009). Reactive mode handling of flexible manufacturing systems. *Mathematics and Computers in Simulation*, 79(5):1421–1439.
- Kanso, M., Berruet, P., and Philippe, J. (2009). Multi-criteria decision making approach for reconfigurable manufacturing systems. In *proceeding ISBN of the 28th EAM Conference on Human Decision-Making and manual Control, SEPT. 3-4, 2009*, pages 37–44.
- Kurnaz, S., Cohn, A., and Koren, Y. (2005). A framework for evaluating production policies to improve customer responsiveness.
- Lamotte, F. D. (2006). *Proposition d’une approche haut niveau pour la conception, l’analyse et l’implantation des systemes reconfigurables 2006*. PhD thesis. Université de Bretagne Sud, Lorient, France.
- Nucci, F. and Grieco, A. (2008). The operational strategies in focused flexible manufacturing systems. *ISC’08 International Supercomputer Conference. Heidelberg, Germany, June 18 20*.
- Ryu, K., Son, Y., and Jung, M. (2003). Modeling and specifications of dynamic agents in fractal manufacturing systems. *Comput. Ind.*, 52(2):161–182.
- Terkaj, W., Tolio, T., and Valente, A. (2009). Designing manufacturing flexibility in dynamic production contexts. In *Design of Flexible Production Systems*, pages 1–18.

AN ADAPTABLE ARCHITECTURE FOR INTELLIGENT CONVEYORS

Dennis Ommen, Jens Kamenik, Carsten Beth

OFFIS Institute for Information Technology, Escherweg 2, Oldenburg, Germany
{dennis.ommen, jens.kamenik, carsten.beth}@offis.de

Jan C. Busch, Alexander Kulas, Eckhard Cramer, Axel Hahn

Department of Computing Science, Carl von Ossietzky University, Ammerländer Heerstrasse 114-118, Oldenburg, Germany
{jan.c.busch, alexander.kulas, eckhard.cramer}@uni-oldenburg.de, hahn@wi-ol.de

Keywords: Intelligent Transportation System, Autonomous Agents, Automation Architectures, Networked Robotics.

Abstract: To optimize the transportation processes inside transfer stations the degree of automation has to be increased without the loss of flexibility. Therefore this paper proposes a detailed architecture for an intelligent material flow system based on the technologies of Multi-Agent-Systems (MAS) and wireless sensor networks. Furthermore, a novel framework is proposed that eases the integration process between the MAS and the physical level of heterogeneous conveyors.

1 INTRODUCTION

Flexibility will be one of the most important drivers for technological improvements in material flow systems in the future. A new flexible material flow system has to be reconfigurable by design and in the end there are no additional costs allowed (Windth, 2007). To circumvent this problem new systems should be designed as autonomous decentralized cooperating objects, e.g. goods and the transportation system autonomously make decisions (Scholz-Reiter et al., 2007b). The autonomous decisions can be made in two ways (1) the good driven way (Scholz-Reiter et al., 2007a) (Scholz-Reiter et al., 2006): An embedded device attached to the package escorts the goods to its destination. During the transportation process the embedded device cooperates with the environment to achieve its goal. (2) The transportation system driven way: The environment around the goods makes the decisions. With the arrival and identification of a good at the entry transfer point the intelligent environment autonomously creates a specific transport order. From now on this virtual order escorts each intelligent transportation device that handles these goods.

We assume that for material flows system that are specialized on movement of goods, it will not be feasible to attach an intelligent device to each good, due to cost reasons. Therefore we follow the approach of the intelligent environment with its intelligent transporta-

tion devices. Thereby the intelligent conveyor is able to plan and drive the appropriate route through the environment to reach the sink transfer point. These intelligent transportation devices can have very different abilities and automation degrees and are therefore suited for different transportation task and situations. For example, automated guided vehicles (AGV) are used for basic load while traditional fork lift trucks (FLT) are used in peak load situation. The challenge that will be addressed here is the coordination of heterogeneous conveyors inside a transfer station which is based on Multi Agent Systems (MAS) and Wireless Sensor Networks (WSN).

Paper Organization. The remainder of the paper is organized as follows: Section 2 describes the transfer station scenario with the heterogeneous transportation devices within. Section 3 introduces the architecture of the proposed MAS. Section 4 describes a framework that eases the integration process between MAS and the field level of the different conveyors. Section 5 describes the intelligent environment that is based on WSN technology. Section 6 ends the paper with concluding remarks.

2 SCENARIO

Traditionally, heterogeneous conveyor types are used inside a transfer station. The transportation devices

are operated and/or coordinated by human beings, to transport goods from the entry transfer point to the sink transfer point. Each kind of transportation device has different abilities to cope with the current load situation of the transfer station, depending on its degree of automation and flexibility. Each of these properties make an optimal coordination of the heterogeneous devices (regarding transportation cost, throughput or optimal balance load of the conveyors) very difficult. In the scenario depicted in figure 1, AGV and manually operated FLT work together to cope with the current load situation.

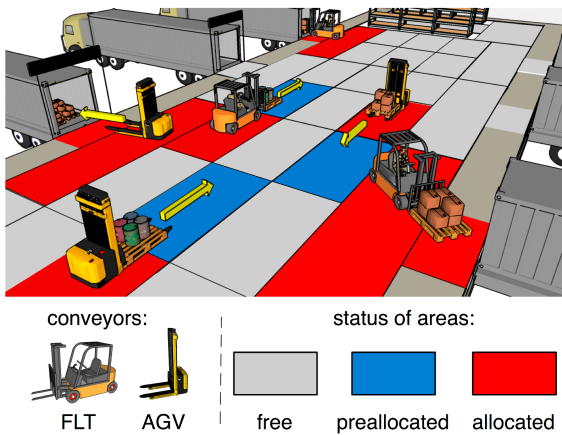


Figure 1: Material flow with an intelligent environment.

To ease the coordination process for an optimal throughput, all conveyors are intelligent transportation systems. This means each conveyor is able to decide itself if he is willing to take a transport order or not. But this freedom comes with duty. Each intelligent transportation system has to offer its position to an intelligent environment. The intelligent environment is virtually separated into areas that can either be free or set to be (pre)allocated by the intelligent conveyors. The different states of areas are needed to avoid collisions between the heterogeneous intelligent conveyors. Before an intelligent conveyor is entering a new area, it will allocate this area (depicted red) or will search for another route to reach his sink transfer point. During the planning process of routes, the conveyor will eventually be able to (pre)allocate the areas it will need during its transport process (depicted blue). The intelligent environment is physically represented by a WSN. This WSN is able to locate the intelligent conveyors and also could act as a database for the mentioned area status information.

3 CONTROL ARCHITECTURE

The scenario described in the previous section would traditionally require a central control, incorporating the state of every conveyor and the position of every good into work directives given to every conveyor. The design and implementation of a central control system is a non-trivial and lengthy task, and “a key cost driver” (Schmidt and Schulze, 2009) in the calculation of transport costs.

De-centralized systems promise to alleviate this costs by using self-organization of (semi-) autonomous agents, allowing for rapid adaption of the transport routes to new requirements. This paper proposes a decentral organization for material flows, using a system of multiple autonomous agents working together (MAS). This distributed approach is also the subject of other research projects, for example (Ten Hompel et al., 2008) and (Leit 2009).

3.1 Agents

Our architecture consists of two types of agents – transfer points and conveyors – interacting with each other and an intelligent environment. While conveyor agents expose a uniform set of properties and abilities to the MAS, the underlying physical objects they represent can differ widely. In this regard, an agent is an abstraction, providing a common way of interacting with physical objects that perform the same function (e.g. conveying goods), yet function in different ways.

Transfer Point. A transfer point represents a place where goods are transferred from one part of the system into another. The physical objects these agents represent could be terminals stationed at transfer stations, used to enter transport jobs into the MAS. The transfer point agents then communicate the job to all conveyor agents.

A transport job consists of the start and end transfer points, as well as a time constraint for finishing the transport and relevant information about the good to be conveyed, like weight and dimensions.

The conveyor agents, upon receiving a job, can bid on the job in the manner of an auction, giving a cost estimate for the transport. The transfer point agent selects the conveyor with the least cost estimate, and assigns the job to it.

Conveyor. A conveyor agent exposes the conveyors functionality to the MAS, allowing the transport of goods to be planned and executed. They receive job announcements from the transfer point agents, and bid on them with a cost estimate for the transport. The

estimate is calculated by finding a route through the work area, querying the environment (see figure 2) for the states of the areas along the planned route (see section 2 and figure 1), and calculating the costs of a transport along that route. The winner of the auction then generates a job from the auction, and proceeds to pick up and deliver the goods.

To do this, a series of operations needs to be generated which the physical agent has to execute. This is done by breaking a transport job down into abstract actions, which are then turned into a series of operations using a device dependent transformation. An action is an abstract description of one step of the transport process, such as ‘pick up good X’ or ‘transport good to point A’, while an operation is an atomic function that a physical conveyor can perform, such as ‘drive forward’, ‘revolve belt’ or ‘locate good X within environment’. Operations are sent via a middleware, which allows for device independent communication. An overview of this process can be seen in 2.

However, in case of a human operating the conveyor, actions are not broken down further but sent directly, since it can be assumed that a human will understand the intention and act correctly, making fine grained instructions pointless.

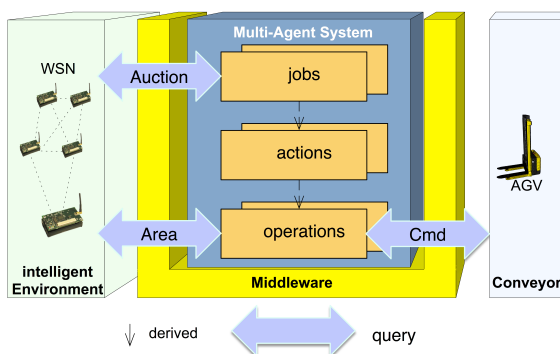


Figure 2: Control architecture of the conveyor.

3.2 Agent Requirements

To let an agent fulfill its tasks, it needs an intelligent environment and a physical conveyor. An additional middleware enables the agent to interact with the physical objects, which can differ widely, and the intelligent environment (see figure 2). For a detailed description of the middleware, see section 4. The intelligent environment consists of a WSN (see section 5), where sensor nodes are placed throughout the working area. It will be queried for information by the conveyor agents, for example for their own position in the environment, whether or not a certain area

is currently occupied by other conveyors, or how the occupation of an area has been in the past.

In a pro-active scenario, the conveyors may additionally exchange information about their predicted usage of the environment, further improving the quality of the planned routes (preallocation, see section 2). However, since interactions with human beings can not be fully predicted, information generated this way is only of limited accuracy, possibly resulting in a probabilistic planning strategy. It is currently not determined how or where data about future usage of areas can be saved and kept available to all agents. Further research into this problem field is required.

4 MIDDLEWARE

The aforementioned MAS is one of the key component to operate the described material flow scenario with its heterogenous conveyors. But the MAS needs always direct contact to the conveyor or its operator to put the planned route into practice. Nevertheless the communication between the MAS and the conveyor is from the MAS point of view very abstract. Therefore is a link between the MAS and the Conveyor needed. This chapter presents a novel Integration Process for Conveyor with a corresponding configurable Middleware (MW). Thereby, the MW acts as mediator between the MAS and the Conveyers Sensor and Actors. Therefore we briefly discuss the requirements of the such a MW and than propose the MW architecture itself.

4.1 Middleware Requirements

Such a middleware should ease the integration process between the MAS and the conveyor. Therefore the MW should address the following aspects:

- adaptable to the unique conveyor specific query set, that the MAS is presupposing
- adaptable to the heterogenous sensors and actors the conveyor possesses. This means:
 - eased integration of different communication protocols, like CAN or ProfiBus, into the MW
 - on-the-fly transformation of the exchanged binary messages between the MAS and conveyors. This is necessary due to the different message content that the MAS and the Conveyors imply.

4.2 Architecture

To comply with the above stated requirements our approach is based on SOA Technology. Although we don't assume that XML based Automation-Protocols, e.g. OPC UA, will achieve the major acceptance in the domain due to real time constraint and overhead, we propose to adapt the SOA idea of an Enterprise Service Bus (ESB) to our needs. Historically ESB comes from the Enterprise Architecture Integration where systems and protocols are traditionally very heterogenous. Hence an ESB allows to integrate different protocols extensions (called adapters) to the middleware. To mediate now between the different semantic meaning of the messages a message transformation is needed, which is traditionally XSLT based. To overcome the realtime hurdles due to the XSLT interpretation of the ESB our Middleware acts as extendable Framework (see figure 3).

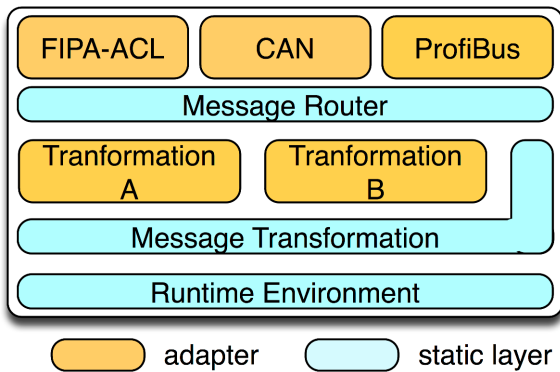


Figure 3: Middleware Architecture.

Each middleware layer (depicted in blue) will provide basic function that run the specific adapters (depicted in red). Such an Architecture will ease the portation step to other programming languages and other operating system so that this MW could even be ported to an Embedded System.

5 WIRELESS SENSOR NETWORK

In the cognitive environment of the proposed transfer station scenario (Ommen et al., 2009) (Beth et al., 2009) the routes are distributed and intelligent entities. They need to be networked to communicate with the conveyors to enable the distributed localization method. The routes have to provide the ability to localize the conveyor, to answer the occupation state of the route and to give an average usage feedback of

the route (re-active approach). Furthermore the sensor network should acts as a communication relay for the conveyors and other sensor nodes.

5.1 Limited Resources

Resources are a critical point in designing a sensor network. For the deployment of the sensor network in the logistic in-door scenario the following points have to be considered.

1. One point to consider is the critical bandwidth and range of the sensor network.
2. For the deployment of a large number of nodes in industrial environments – with high flexibility and mobility requirements – the installation of a static electric power supply for all nodes is too expensive and not feasible. Therefore there are always nodes that can not be supplied with static electric power and the limited energy of the sensor nodes has to be considered.
3. A third point is the limited storage space of a sensor node, e.g. a typical sensor node like the MicaZ¹ has only 4KB RAM and 512 KB of flash memory on board.
4. A last point is the inaccurate RSSI-based localization of standard sensor nodes.

5.2 Approaches

In the proposed scenario the sensor network uses multi-hop communication to circumvent the limited propagation of radio waves in logistic in-door facilities and to extend the range of the sensor nodes and the conveyors.

Localization Procedure. The conveyors need to know their own position. The distributed fixed sensors nodes have an area to observe. This area is surrounded by at least three - in our model - four sensor nodes. A special node of the group is the home node, it is a sensor node responsible for fault tolerantly storing the data of one or more routes in the area. For the localization a trilateration method based on IEEE 802.15.4b compatible (Nanotron Technologies, 2007) ranging data will be used. In (Röhrig, 2009) this localization method, combined with appropriate Kalman filtering, provided an accuracy of 0,5m. The conveyor node has to initiate the localization procedure with the nodes of the localization group. A localization group is formed by issuing a broadcast call

¹www.xbow.com

for all neighborhood nodes able to localize the conveyor node. The localization procedure comprises the gathering of the distances from the mobile conveyor node to all nodes in the localization group and the necessary algorithms² to calculate the position. Afterwards, the conveyor node sends the position back to the home node. The home node of the route has to store the arrival time of the query, the ID of the conveyor and the estimated occupation time of the route. The estimated occupation time is a worst case time consisting of the average passing time for the route and an additional time for loading or unloading goods on the route.

Distributed Queries. Occupation and Average Queries from the conveyor nodes can be executed from every point in the scenario area and are answered by the home nodes of the localization groups. On *Occupation Queries* the conveyor asks for the occupation state of a route with a certain ID and at the time of the query. To answer this query, a message with the route ID and the occupation query type has to be sent to the home node of the route. If the route is occupied, the home node answers the query by sending back the occupation time to the inquirer ID. On *Average Queries* The conveyor asks for the average occupation state of a route in the past. This allows the conveyor to derive a usage estimation and to choose an alternative route in case of a possible jam situation. To answer this query, a message with the route ID and the estimation query type has to be sent to the home node of the route. The home node has to gather the past occupation times of the route belonging to the queried ID and averages them. Afterwards, it sends the averaged time back to the inquirer.

The data scheme for the localization of the nodes needs the attribute of the identity of the conveyors, the entry time of the conveyors and the (estimated) occupation time of the route:

RouteOccupation : { [ID: char, entrytime: float, occtime: float] }

The data scheme for the averaged values of the nodes needs the attribute of the identity of the route and the average occupation time $t_{occtime}$ in time period $t_{average}$:

RouteAvg: { [RouteID: char, occtime: float] }

Storage Strategy and Energy Consumption. (Diao et al., 2007) stated that the storage of sensor data is energetically (Pottie and Kaiser, 2000) more efficient instead to transmit the data via the rf-channel. As a side-effect the bandwidth usage of

the sensor network is reduced too. Furthermore, a consideration of the storage place of the sensor data necessary. The variables that have to be considered are the query rate, the event rate and the storage place. There are three approaches to choose the storage place in a distributed sensor network. The central data storage to an external server. This is optimal for applications where the query rate to the server is larger than the event rate (Tilak et al., 2006). For applications with a higher event rate than query rate the local storage is more energy efficient. For large scale sensor networks, e.g. greater than 10000 sensor nodes, with a slightly higher event rate than query rate - the data centric storage (DCS) (Ratnasamy et al., 2002) is suitable. Here the place for data storage is chosen based on the attributes of the sensor data. For the scenario in this paper the *local storage* is suitable. The data is stored at the same place where the query takes place and therefore no additionally communication is necessary. The limited storage space on the sensor nodes has to be considered too. Assumed that the typical workload for an 100m x 100m sized cross-docking area is 750000 goods a year, the workload per hour would be 129^3 goods. If the localization group is squared and has a side length x_{loc} of 25m, e.g. the typical working range for a nanotron localization system, the memory used per hour M_{hour} would be 9288 Bytes (8 localizations x 9 bytes per localization x 129 goods). A simple approach is to store every localization event occurred in one single home node responsible for the whole scenario. The (MicaZ) RAM would last for 1/2 hour and using the (MicaZ) flash storage would last for 56 hours. The intelligent approach is to store the localization events only for the occupation time of the conveyors and afterwards the time is used to build the average occupation time. If only the average value per hour is stored - the RAM is sufficient for 4 days (4kb / 5 bytes/ 8 localizations) and the flash storage would last for 546 days.

5.3 Wireless Sensor Network - Conclusion

The main problems for a wireless sensor network in a decentral logistic scenario are reliable localization and the limited resources of the nodes. It was discussed that the localization with IEEE 802.15.4b based ranging is suitable for the scenario. Furthermore, with an appropriate storage strategy the limited storage space and energy resources are sufficient for the scenario.

²Solving a linear equation

³365 days a year and 16 hours a day

6 CONCLUSIONS AND FUTURE WORK

In this paper several approaches for optimization of the transportation processes inside a transfer stations have been showed. The proposed architecture for intelligent material flow systems includes Multi-Agent-Systems and wireless sensor networks that are cooperating in a pro-active or re-active manner. Further research has to be done to decide the best cooperating manner. The main problems for a wireless sensor network in a intelligent material flow system have been figured out and it was discussed that with an appropriate storage strategy the limited storage space and energy resources are sufficient for an intelligent material flow systems. Furthermore, a novel framework was proposed that eases the integration process between the control level and the physical level of heterogeneous conveyors. In the future the architecture and the framework have to be tested for real logistic setups and the *lessons learned* will be demonstrated in successive publications.

ACKNOWLEDGEMENTS

This contribution was supported by the German federal state of Lower Saxony with funds of the European Regional Development Fund (ERDF) within the scope of the research project "Cognitive Logistic Networks" (CogniLog).

REFERENCES

- Beth, C., Kamenik, J., Ommen, D., and Hahn, A. (2009). Design aspects of cognitive logistic systems. In *Int. Conf. on Dynamics and Logistics (LDIC)*, Bremen.
- Diao, Y., Ganesan, D., Mathur, G., and Shenoy, P. J. (2007). Rethinking data management for storage-centric sensor networks. In *CIDR*, pages 22–31.
- Leitao, P. (2009). Agent-based distributed manufacturing control: A state-of-the-art survey. *Eng. Appl. Artif. Intell.*, 22(7):979–991.
- Nanotron Technologies, G. (2007). Real time location systems, version 1.02, document-id: Na-06-0148-0391-1.02.
- Ommen, D., Beth, C., Kamenik, J., and Hahn, A. (2009). A system-architecture for robotic movements of goods - approaches towards a cognitive material flow system. In Filipe, J., Andrade-Cetto, J., and Ferrier, J.-L., editors, *ICINCO-RA*, pages 342–347. INSTICC Press.
- Pottie, G. J. and Kaiser, W. J. (2000). Wireless integrated network sensors. *Commun. ACM*, 43(5):51–58.
- Ratnasamy, S., Karp, B., Yin, L., Yu, F., Estrin, D., Govindan, R., and Shenker, S. (2002). Ght: A geographic hash table for data-centric storage in sensor networks. In *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*. ACM.
- Röhrig, C. (2009). Lokalisierungsverfahren für drahtlose Sensornetzwerke. In Sieck, J. and Herzog M.H., editors, *Wireless Communication and Information – Radio Engineering and Multimedia Applications*, pages 81–97. Verlag Werner Hlsbusch, Boizenburg, Deutschland.
- Schmidt, T. and Schulze, F. (2009). Future approaches to meet complexity requirements in material handling systems. In *FME Transactions*, volume 37, pages 159–166.
- Scholz-Reiter, B., Jagalski, T., and Bendul, J. (2007a). Selbststeuerung logistischer prozesse in produktionsnetzen. *Industrie Management*, 23:7–10.
- Scholz-Reiter, B., Jagalski, T., and de Beer, C. (2007b). Selbststeuerung logistischer prozesse in produktionsnetzen. *Industrie Management*, pages 19–22.
- Scholz-Reiter, B., Rekersbrink, H., and Freitag, M. (2006). Kooperierende routingprotokolle zur selbststeuerung von transportprozessen. *Industrie Management*, 22:19–22.
- Ten Hompel, M., Follert, G., and Roidl, M. (2008). Künstliche intelligenz im internet der dinge: Die zukunft der materialflusststeuerung mit autonomen agenten.
- Tilak, S., Abu-Ghazaleh, N. B., and Heinzelman, W. (2006). *Storage Management in Wireless Sensor Networks*, pages 257–281. John Wiley & Sons, Inc.
- Windth, K. (2007). Selbststeuerung intelligenter objekte in der logistik. *Selbstorganisation - Ein Denksystem für Natur und Gesellschaft*, pages 19–22.

ENGINEERING TIME IN AN ONTOLOGY FOR POWER SYSTEMS THROUGH THE ASSEMBLING OF MODULAR ONTOLOGIES

Jorge Santos, Luís Braga

Departamento de Engenharia Informática, Instituto Superior de Engenharia, Porto, Portugal
{ajs, 1050515}@isep.ipp.pt

Anthony G. Cohn

School of Computing, Leeds University, Leeds, U.K.
a.g.cohn@leeds.ac.uk

Keywords: Power Systems, Ontologies, Knowledge Engineering, Temporal/Spatial Reasoning and Representation.

Abstract: In this paper we investigate how timeless ontologies such as DFault, an ontology for fault diagnosis in power transmission networks can be re-engineered to include temporal entities. We propose a methodology, FONTE (Factorising ONTology Engineering complexity), that allows this complex process to be factored by dividing the problem into parts: modelling the domain concepts ontology (atemporal and aspatial), modelling or acquiring the temporal and /or spatial ontology, and finally producing the target ontology by assembling these modular ontologies via a semi-automated process.

1 INTRODUCTION

In recent years many technologies have been developed allowing the capture and transmission (*e.g.*, smart sensors networks and wireless communications) of information about the state of a wide range of devices connected to networks scattered over large geographical areas (*e.g.*, power transmission grids). The emergence of technology standards lowered the prices for these devices/technologies and eased its integration with existing systems. The result is that the data owners are overloaded with large quantities of heterogeneous information. One major challenge therefore consists of developing applications able to provide end-users with a system overview in a human readable, relevant and consistent manner, so as to interpret and manage the system being monitored. There are at least two issues obstructing this goal: the information heterogeneity and the requirement for appropriate models for spatial and temporal knowledge (STK). The modelling of STK in intelligent systems is a complex process. This work proposes a semi-automatic method, FONTE (Factorising ONTology Engineering complexity), based on the assembling of multiple ontologies in order to obtain the target ontology. This method tackles both the above problematic issues: information heterogeneity and modelling STK. The first is addressed through the use of ontologies as building blocks, given that ontologies are

shared and common agreed models about a specific domain. The second issue is addressed through the use of a set of rules that drives the process of assembling orthogonal categories, like space and time, in a semi-automatic way, releasing the knowledge engineer and the experts from the rendering of intricate concepts related to complex theories of time and space.

2 FONTE METHODOLOGY

In order to illustrate the assembly process two ontologies will be used as building blocks for the target ontology, a temporal ontology and the timeless domain ontology DFault (www.dei.isep.ipp.pt/~jsantos/F2C/DFault) for fault diagnosis in power transmission networks.

Modular Ontologies. The assembly process can be used either for the development of ontologies with time from scratch, or for re-engineering existing atemporal ontologies in order to include time. For our case study we have used the time-less DFault ontology that captures the main concepts related to the characterisation of the Portuguese National Electricity Transmission Grid (RNT) (www.ren.pt), and concepts related to fault diagnosis in power transmission networks (previously used to develop the SPARSE sys-

tem (Vale et al., 2002)).

The DFault ontology (OWL version) includes 92 classes, 30 properties and 30 restrictions. The hierarchy is divided in four sub hierarchies, with root concepts: Event, Entity, Local and Object. The Event class includes the different type of events that may occur in the electric network, such as breaker tripping or power lines going out of service. The Entity class includes the organisations (e.g., suppliers, clients) and persons involved in the management/exploration of the network. The Object class is divided into physical objects, which include the electric network devices and facilities, and abstract objects, which include the faults diagnosis and messages acquired from the SCADA (Supervisory Control And Data Acquisition) system. The Local class represents a hierarchy of physical locations, which are used to spatially characterise the network devices and facilities.

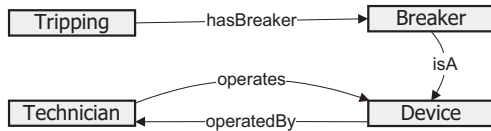


Figure 1: Excerpt of the DFault Ontology.

Fig. 1 presents an excerpt of the DFault ontology that was used in order to elucidate the FONTE method and the Protg plug-in that supports it. A circuit breaker is an electrical device designed to protect a circuit from damage caused by an electrical event such as overload or short circuit. Its basic function is to detect a fault condition and, by interrupting continuity (tripping), to immediately discontinue electrical flow. A circuit breaker can be reset (either manually or automatically) to resume normal operation.

The temporal ontology used in our case study (see Fig. 2 for the UML-like depiction of an excerpt) embodies many concepts such as Instant or Period found in ‘standard’ ontologies such as OWL-Time (www.w3.org/TR/owltime/) or SUMO (www.ontologyportal.org/) and assumes a standard interpretation, mapping time points and intervals to real numbers and intervals on the real line respectively. A temporal representation requires the characterisation of time itself and temporal incidence, which are represented in our temporal ontology by TemporalEntity and Eventuality, respectively. A further notion, TimedThing, which is used during the assembly process, bridges between temporal concepts and domain concepts .

Temporal Entities. In the temporal ontology we used for the case study there are two subclasses of TemporalEntity: Instant and Period. The relations *before*, *after* and *equality* can hold between Instants,

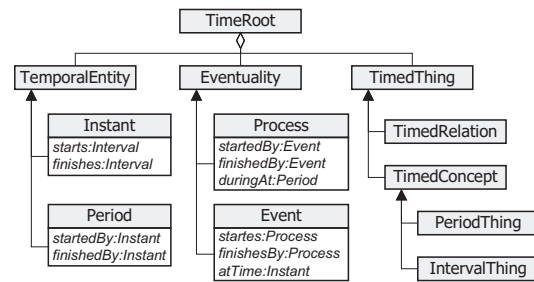


Figure 2: Excerpt of the Temporal Ontology.

respectively represented by the symbols: $<$, $>$, $=$, allowing an algebra based on points to be defined (Vilain et al., 1989). It is assumed that the *before* and *after* are strict linear, i.e. irreflexive, asymmetric, transitive and linear. The thirteen binary relations proposed in the Allen’s interval algebra (Allen, 1983) can be defined easily from $<$, $>$, and $=$. The *starts* and *finishes* are relations from TemporalEntity to Instant. Also, there are no null duration periods and each period is unique.

Processes and Events. There are two subclasses of Eventuality, Process and Event, in order to be possible to express continuous and instantaneous eventualities, respectively. Event has a relation *atTime* to Instant while Process has a relation *duringAt* to Period. The relations *starts* and *finishes* can be used to state what can start or finish a process.

The Assembly Method. FONTE (Santos and Staab, 2003) allows the targeted complex ontology to be built by factorising concepts into their temporal, spatial and domain (atemporal and aspatial) aspects, and then assembling the temporally/spatially situated entity from these primitive concepts. This is more similar to a Cartesian Product than a union of ontologies. Each of these component ontologies can be built/acquired independently, allowing a factorisation of complexity. The ontologies assembly is performed through an iterative and interactive process that combines two types of inputs: *i*) human assembly actions between the component ontologies; and *ii*) automatic assembly proposals obtained from semantic and structural analysis of the ontologies. This process is propelled by a set of rules and a set of constraints. The set of rules drives a semi-automatic process proposing assembly actions; the set of constraints allows to assess which of the generated proposals are valid.

The proposed methodology divides the task of building a temporalised ontology for power network control and monitoring into the task of constructing simpler sub-ontologies which can be built independently, factorising the problem complexity. The domain concepts ontology DFault (DO) and the temporal ontology (TO) (both described in the previous section) will be used to illustrate the assembly process.

The methodology hence proposes to obtain the target ontology (power network control and monitoring with a temporal) through a semi-automatic process of assembling. The concept `DO.Device.Breaker` will be linked with `TO.TemporalEntity`, while the property `DO.Device.Breaker.triggering` will be linked with `TO.Process`. In this way the instances of `DO.Device.Breaker` concept will have a time span of use. Instances of `DO.Device.Breaker.triggering` process have a time span in which they occur and additionally the user will be asked to assemble the events that define the beginning and ending of this process (e.g., `DO.Device.Breaker.opening` and `DO.Device.Breaker.closing`). After this step, related concepts, properties and axioms are proposed for assembling, through a cascade process: e.g. sibling concepts of `DO.Device.Breaker` such as `DO.Device.Line` or `DO.Device.Transformer` will be proposed for the assembly process; properties and axioms related to `DO.Device.Breaker.triggering` like `...triggering.monophasic` or `...triggering.triphasic` would be proposed for assembly depending on their characteristic event sequence.

Protg Plug-in. In order to support the iterative and interactive process used in FONTE, a Protg plug-in was developed. An assembly task consists of the definition of the actions to be performed in the target ontology after the performance of an assembly action (e.g., creation of a relation `isA` between a domain and a temporal concept). Due to the characteristics of the platform (Protg), two types of tasks were defined:

Internal Tasks, which allow basic operations to manipulate the ontologies to be performed (e.g., create, delete and modify classes or properties), and provide access to the API functionalities of the Protg platform in a transparent mode;

External Tasks. (also called assembly rules), which are procedures written in a pseudo-code language that includes common program language instructions (e.g., if, then, else) and special keywords (e.g., do, propose, check) whose semantics has been previously provided. In order to facilitate the edition/creation of these tasks, a specific tool supported in a graphic interface was developed (see below).

The FONTE plug-in architecture (see Fig. 3) has different abstraction levels which present several advantages for the knowledge engineer: *i*) the knowledge engineer does not need to know the specifics of the Protg API to manipulate the ontologies. In addition, the Internal Tasks provide an abstraction level between External Tasks and Protg API assuring independency between the External Tasks and the Protg API version; *ii*) the External Tasks may be created/edited during the execution time and do not re-

quire the alteration of the application and consequent compilation; *iii*) Different rules set (stored in distinct files) allow different temporal/spatial theories in the assembly process to be used in a flexible way.

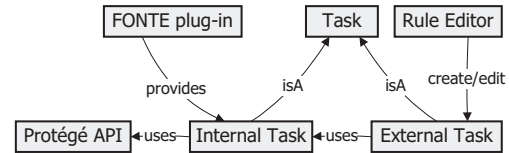


Figure 3: FONTE plug-in architecture.

The plug-in provides a set of functionalities, such as: linking concepts of the domain and temporal/spatial ontology; accepting, rejecting or even delaying the execution of a task; and visualising statistics of the assembly process. As depicted in Fig. 4, the plug-in has two panels for the manipulation of ontologies (on the left-hand side) and a list of proposals (on the right-hand side). The panel further to the left contains the domain ontology (DFault, which is timeless and spaceless); from this panel it is possible to access the classes and properties hierarchies. The other panel contains the temporal/spatial ontologies to be used as construction blocks for the production of the target ontology. The list of proposals contains the records of the task instances generated by the system. Details of this list are presented below.

To promote the assembly process the knowledge engineer needs to select the ontologies that will participate in the assembly process as well as the files containing the assembly rules for each ontology; these can be selected using the setup window (triggered by the setup button shown in Fig. 4).

All the tasks that are successfully performed (either triggered manually by user-driven action or automatically by the structural analysis module) are added to a list containing the instance tasks historic. Associated with each task instance proposal there is: a trigger list (the elements that triggered the proposal); the task weight (an indication of the importance of each proposal, which influences the likelihood of proposal acceptance during the assembly process); and a question in natural language (a phrase that summarises the proposal objective, instantiated with the elements contained in the instance task).

As the assembly process progresses, more proposals are generated. If different concepts happen to propose the same task instance, all the elements that have triggered that proposal are included in the trigger list and the proposal weight is increased.

All the proposed task instances are stored in the list of proposals, which can be sorted by different criteria (e.g., id, trigger or weight). The user can then accept, reject, or even delay for later analysis,

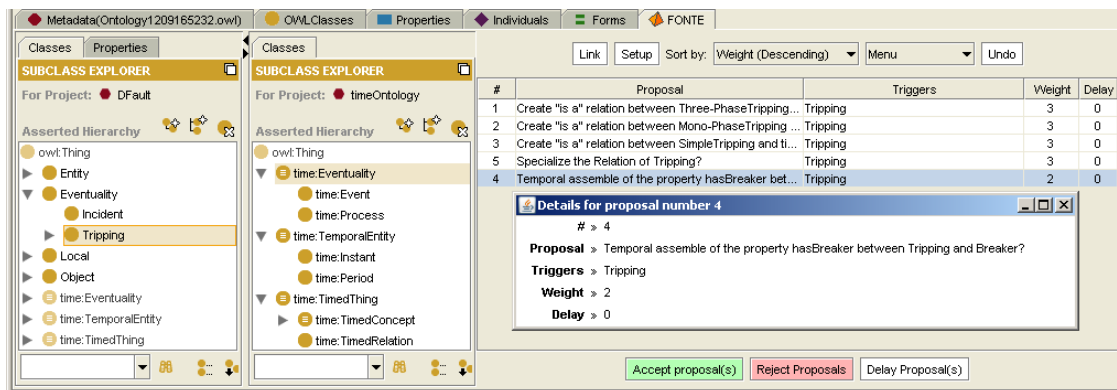


Figure 4: FONTE plug-in for Protg.

each of the proposals. In order to avoid overloading the knowledge engineer with useless proposals, rejected proposals are never automatically proposed again(though they may be manually retrieved).

In addition to the functionalities described above, the plug-in also provides statistics about the assembly process (results of the tool performance, including the initial and current status of the domain ontology, the number of tasks that has been initiated by the user and how many proposals have been accepted or rejected). A facility for saving a sequence of performed tasks as a *script file* is also provided to allow a knowledge engineer to easily totally or partially repeat a certain set of tasks.

An application was developed to facilitate the creation of external files of rules. This supports the knowledge engineer through a simple and interactive graphic interface. The file management system provides a graphic visualisation of the rules included in each file and offers various functionalities, such as: sorting the list through different criteria; modifying the order in which the rules are interpreted during the assembly process; visualising the rules in XML or pseudo-code; removing, editing or creating new assembly rules. The knowledge engineer is alerted about potential consistency errors (*e.g.*, non declared variables) or warnings (*e.g.*, to declaring a variable that is not used).

3 CONCLUSIONS AND FUTURE WORK

In this paper we discussed the engineering of time in DFault, an ontology for fault diagnosis in power transmission networks. We proposed FONTE, a method that supports the engineering of complex ontologies including temporal and/or spatial knowledge that allows process complexity to be factored by dividing

the problem in parts: modelling the domain concepts ontology (atemporal and aspatial), modelling or acquiring the temporal and/or spatial ontology, and finally producing the target ontology by assembling these modular ontologies through a semi-automated rule based approach.

A Protg plug-in developed to support method FONTE, which allows FONTE to be used in an integrated form in the development of ontologies, was also described. The FONTE methodology works independently of the temporal/spatial theory since it allows different sets of assembly rules to be used for each specific theory. A tool to support the creation/editing of these rule sets was also summarised.

Future work includes: *i*) evaluating the generic characteristics of the proposed method with different spatial/temporal ontologies (including 4D ontologies); *ii*) developing a functionality to predict the impact of the acceptance of a particular proposal; *iii*) improving the generation of automatic proposals during the assembly process; (ii) and (iii) may be achieved through the use of semantic analysis, previously successfully used in diverse processes of ontology engineering (*e.g.*, merging, mapping and alignment).

REFERENCES

Allen, J. (1983). Maintaining knowledge about temporal intervals. *Communication ACM*, 26(11):832–843.

Santos, J. and Staab, S. (2003). FONTE - Factorizing Ontology Engineering complexity. In *The 2nd Int. Conf. on Knowledge Capture (K-Cap'03)*, pages 146–153.

Vale, Z., Ramos, C., Faria, L., Malheiro, N., Marques, A., and Rosado, C. (2002). Real-time inference for knowledge-based applications in power system control centers. *Journal on Systems Analysis Modelling Simulation (SAMS)*, Taylor&Francis, 42:961–973.

Vilain, M., Kautz, H., and Beek, P. (1989). Constraint propagation algorithms: a revised report. *Readings in Qualitative Reasoning about Physical Systems*.

ASSEMBLY SYSTEMS FOR LOW PRODUCT DEMAND

Estimation of Final Results

Waldemar Grzechca

*Institute of Automatic Control, The Silesian University of Technology, ul.Akademicka 16, 44-100 Gliwice, Poland
waldemar.grzechca@polsl.pl*

Keywords: Assembly line balancing, Single line structure, Assembly round table, Quality of results.

Abstract: The paper considers assembly systems for low product demand. In the last five decades a large variety of assembly line structures and solutions procedures have been proposed to balance assembly line. Author of this paper compares single assembly line and assembly rotating round table. Estimation of final results of balance of both structures is discussed. It is shown that implementation of different structures are appropriate for low product demand. Numerical example of design assembly single line and assembly rotating round table helps to understand mentioned structures.

1 INTRODUCTION

Since always people created new items for their own needs and if these appeared to be helpful they tried both to improve them and manufacture them faster. In order to balance supply and demand the development of technology was a must. Definition of production can be therefore understood as transforming raw materials into a complete valuable product. This transformation combines various tasks of human work, automation and technology. It consists of steps after which the temporary product is closer to the final state. All these processes combined together define the assembly line which formal definition states: Industrial arrangement of machines, equipment, and workers for continuous flow of workpieces in mass-production operations. An assembly line is designed by determining the sequences of operations for manufacture of each component as well as the final product. Each movement of material is made as simple and short as possible, with no cross flow or backtracking. Work assignments, numbers of machines, and production rates are programmed so that all operations performed along the line are compatible. Automated assembly lines consist entirely of machines run by other machines and are used in such continuous-process industries as petroleum refining and chemical manufacture and in many modern automobile-engine plants. Although it does not seem difficult by the definition it is a complex field of research. One of the reasons may be the fact that the

first automated production line was implemented in 20th century, actually in the year 1913 in Ford Motor Company, USA. In assembly systems the most often used is the flow line – a particular example of such a structure is the assembly line. Balancing of such a line consists of assigning various tasks to work stations (Salveson, 1955). The objective of balancing leads to defining the cycle time with constant number of work stations or inversely calculating the number of stations with given cycle time. In order to start balancing we need to have a finite set of work stations, tasks with corresponding times and relationships between them i.e. in a form of a precedence diagram. Balancing of an assembly line is the answer to the question - how to allocate resources on a flow line in order to finalize the end product most effectively. Effectively in this case means assigning tasks equally between stations to minimize idle times and equalize work load. A balanced line needs to fulfill (Sury, 1971), (Scholl, 1998), (Beker and Scholl, 2005):

- precedence diagram restrictions
- positive number of stations (at least one)
- cycle time c greater or equal maximum station time.

2 ASSEMBLY LINE STRUCTURE

There exist also a classification regarding plant layout which is used to describe the arrangement of

physical facilities in a production plant (Scholl, 1998). Five types of layout can be distinguished:

- serial lines,
- U-shaped lines,
- parallel lines,
- parallel stations,
- two-sided lines.

2.1 Serial (Single) Lines

This is a very basic layout of a flow line production systems. It is determined by the flow of materials. It is mostly used for small size products. These lines have several disadvantages:

- monotone work,
- sensibility due to failures,
- inflexibility due to changing demand rates.

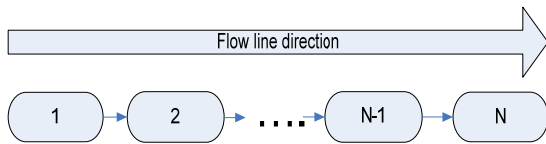


Figure 1: Serial line.

2.2 U-shaped Lines

In order to deal with the problems of a serial line it was redesigned to a form of U-shape (U-line). In such a line operators can work at more than one station simultaneously. For example first operator may both load and unload product units. As they are included in more tasks during production process they are gaining very important experience and enlarge horizons. It is very helpful in case of just-in-time production systems as it improves flexibility which is crucial in dynamically changing demand rates. What more, stations are closer together what results in better communication between operators and in case of emergency they are able to help each other effectively.

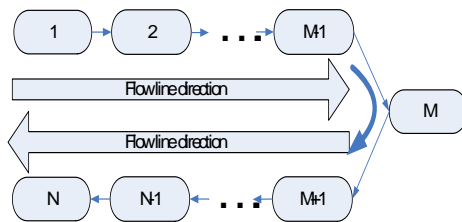


Figure 2: U-line structure.

2.3 Parallel Lines

In order to deal with problems described in case of a serial line it might be a good idea to create several lines doing the same or similar tasks.

Figure 2. U-line structure.

The advantages of such a solution (Sauer, 1997):

- increased flexibility for mixed-model systems,
- flexibility due to changing demand rates,
- lowered risk of machine breakdown stopping the whole production,
- cycle time can be more flexibly chosen which leads to more feasible solutions.

The optimal number of lines is however a subject of discussion for every single case separately.

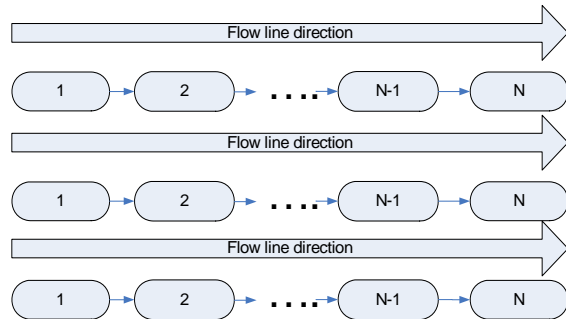


Figure 3: Parallel lines.

2.4 Parallel Stations

As an extension of serial lines bottlenecks are replaced with parallel stations. Tasks performed on parallel stations are the same and throughput is this way increased (Askin and Zhou, 1997).

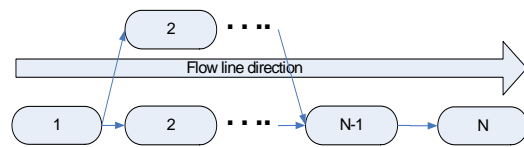


Figure 4: Parallel stations.

2.5 Two-sided Lines

This kind of flow lines is mainly used in case of heavy workpieces when it is more convenient to operate on both sides of a workpiece rather than rotating it. Instead of single working-place, there are pairs of two directly facing stations such as 1 and 2. As an example car line can be considered, and mounting some parts like: side – doors (left, right side), muffler (i.e. right side) or lights with no

preference to the side. Such a solution makes the line much more flexible as the workpiece can be accessed either from left or right (Bartholdi, 1993). In comparison to serial lines:

- it can shorten the line length,
- reduce unnecessary work reaching to the other side of the workpiece.

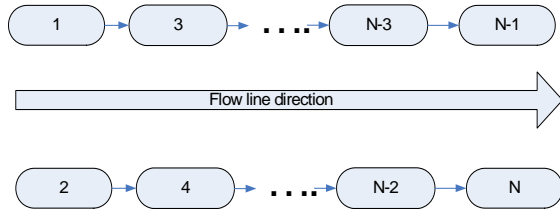


Figure 5: Two-sided line.

3 LOW MIX PRODUCT DEMAND

The volume of production is not a widely discussed topic over the literature. There are numerous articles about mixed-model assembly systems however they do not investigate the problem of low product demand. A formulation of a problem given in (Bukchin et. al, 2002) should give an idea about it. J. Bukchin indicates that it's long gone, when everybody was buying a black painted Ford T as long as it was cheap. Back then, high productivity was achieved by introducing a perfectly single model with no additional features.

Nowadays, the life cycle of a product is relatively short and the demand for varied product is high. Consequently, a set of similar products needs to be assembled in relatively low volume. The goal to such an approach is flexible responding to shorter product life cycles, low to medium production volumes, changing demand patterns and a higher variety of product models and options.

The conditions for such an installation are:

- assembly-to-order production,
- low product demand (low volume production),
- number of tasks greater than number of stations,
- lack of mechanical conveyance,
- Highly skilled workers.

It might be extended with conditions given by (Heike et. al, 2001):

- flexible fixtures,
- flexible tooling,
- delivery of material.

Such conditions give a good base for an assembly system robust to demand changes. Having a good balancing algorithm is a goal in this case.

When the demand for a set of similar products is insufficiently high in order to install a complete assembly line a solution given in (Battini et. al, 2007) might be used. Most of the authors use combined precedence diagrams in order to reduce multiple models into a single model. As the plant layout, they majority uses a straight line in some cases allowing parallel workstations for omitting the bottleneck effects. What more, some allow duplicating stations in series. Authors investigation U-shaped lines indicate their benefits over traditional serial lines. Some of them are:

- improvement in labour productivity,
- job enlargement for human operators,
- great interaction between operators,
- reduction in number of required workstations,
- lead time contraction,
- increase of flexibility.

They suggest (Aase et. al, 2004) this kind of lines in case of number of tasks less than 30 and 10 stations. Fixed position layout should be taken into account dealing with heavy workpieces as it is more convenient to switch operators places rather than i.e. rotating the part (Heike et. al, 2001). Generally, when set-up times required between different versions are significantly high a job shop layout suits the best (McMullen, 2007).

4 ASSEMBLY ROTATING ROUND TABLE

The model and the procedure discussed in this section bases on (Battini et. al, 2007). D. Battini introduces a mixed-model assembly system consisting of a rotating assembly table with a fixed number of stations. It is a semi-automated system therefore some stations are occupied by human operators, some by machines and other are free. Human operators are indicated by "O" while automated ones as "A". The resource assignment is assumed to have no limitations, every operator or machine can be placed at any station of the table. The product assembled with such a system is assumed to be homogenous with some additional features that enable creation of joint precedence diagram with known tasks' durations. The rotating table is a multi-turn one, as a matter of fact a batch of one single product is completed in n number of turns, with $n \geq 2$. The table is an example of unpaced

synchronous line controlled assembly system. It means that all the tasks performed by operators need to be completed before the shift of the table. It is assumed that it has a pneumatic motion and all operators need to press a button as an information that they finished their task. If all the tasks are finished the table switches their position with switch time $t_s \geq 2s$ (move time between 2 stations). Every switch of the table – one station at each table switch.

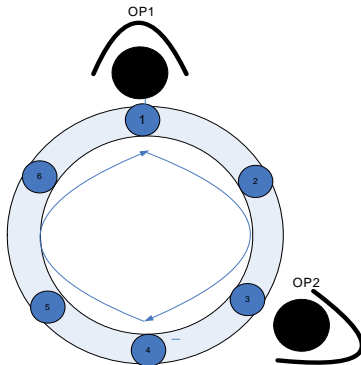


Figure 6: Example of rotating assembly round table (two human operators and six stations).

The assumption of rotating round table are:

1. The assembly rotating round table is multi-turn type.
2. Precedence diagrams of all model types can be accumulated into a single combined precedence diagram.
3. The line production policy is “assembly-to-order”.
4. Workpieces are fixed on the table and there is only one workpiece at the station of the table at a time.
5. Each station has only either one operator or one actuator.
6. Idle operators cannot be used to help the operators of other stations
7. The table switches only when all the opened stations have finished their job.
8. The first task of the cycle is the load of all the workpieces of the same batch on a table and is always assigned to first operator.
9. The last task of the cycle is the download of the assembled units and can be assigned to any operator.

The objectives for this assembly system are:

1. Optimize the load balancing of each station activated in the rotating table
2. Optimize the resource positioning in order to minimize the entire make span of the

assembly batch, and consequently, the average cycle time.

The goal of this paper is to compare serial assembly system and rotating round table.

5 ESTIMATION OF FINAL RESULTS OF BALANCING PROBLEM

Some measures of solution quality have appeared in line balancing problem. Below are presented three of them (Scholl, 1998).

Line efficiency (LE) shows the percentage utilization of the line. It is expressed as ratio of total station time to the cycle time multiplied by the number of workstations:

$$LE = \frac{\sum_{i=1}^K ST_i}{c \cdot K} \cdot 100\% \quad (1)$$

where:

- K - total number of workstations,
- c - cycle time.

Smoothness index (SI) describes relative smoothness for a given assembly line balance. Perfect balance is indicated by smoothness index 0. This index is calculated in the following manner:

$$SI = \sqrt{\sum_{i=1}^K (ST_{max} - ST_i)^2} \quad (2)$$

where:

- ST_{max} = maximum station time (in most cases cycle time),
- ST_i = station time of station i .

Time of the line (LT) describes the period of time which is need for the product to be completed on an assembly line:

$$LT = c \cdot (K - 1) + T_K \quad (3)$$

where:

- c - cycle time,
- K -total number of workstations.

The average cycle time for rotating round table is calculated due to the formula:

$$C = \frac{\sum_{l=1}^Z \sum_{kk \in AS_z} \{\max[t(S_k)]_Z + t_s\} \cdot X_k}{K} \quad (4)$$

where:

- C – average cycle time,
- $t(S_k)$ – station load,

AS_z – set of stations activated in turn z ,
 $Z = 1, \dots, Z$ are table runs,
 K – total number of stations,
 t_s – switch time of the table,
 X_k – distance in switches between the major load station and each activated in turn z .

6 NUMERICAL EXAMPLES

In this chapter an illustrative example of serial assembly line and assembly rotating round table is shown. An 8 tasks example of final product is considered. In both cases for founding end solution of balance a heuristic procedure (Update Immediately First Fit – Number of Followers) was implemented.

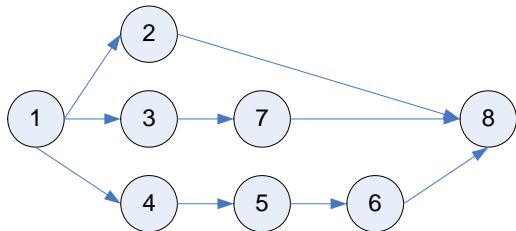


Figure 7: Precedence graph of numerical example.

Table 1: Operation time of numerical example.

Task i	Time t_i	Task i	Time t_i
1	18	5	7
2	13	6	14
3	6	7	11
4	9	8	2

6.1 Serial Assembly Line

We consider serial assembly line with two workers it means with workstation. It is a problem known as Simple Assembly Line Balancing Problem Type 2 when the number of stations is given and value of cycle time is calculated.

$$c = \left\lceil \frac{\sum_{i=1}^N t_i}{K} \right\rceil \tag{5}$$

where:

c – cycle time of serial assembly line,
 t_i – operation time of task i .

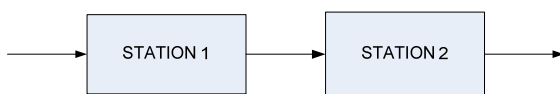


Figure 8: Serial two stations line.

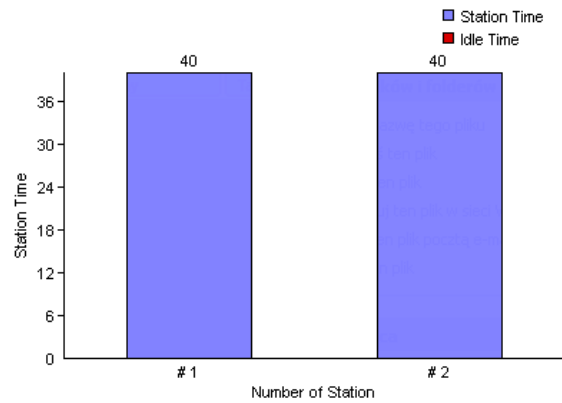


Figure 9: Balance of serial line for calculated example. The calculated cycle time is 40 (the total operation time is 80) so we got final solution of balanced line: Station 1 {1, 4, 3, 7} and Station 2 {5, 2, 6, 8}. The solution is optimal (mostly we obtain using heuristic method only feasible solution) and calculated measures are: SI = 0, LE = 100% and LT = 80).

6.2 Assembly Rotating Round Table

We consider now assembly rotating table with 2 human operators and six workstations. We obtain final results for 6 cases it means we calculate average cycle time for six different location of human workers. Starting from position 1 and 2 we relocate second operator to location 3, 4, 5 and 6. Operator 1 is always assigned to station 1. Relocation of Operator 2 causes that the distance between both workers changes.

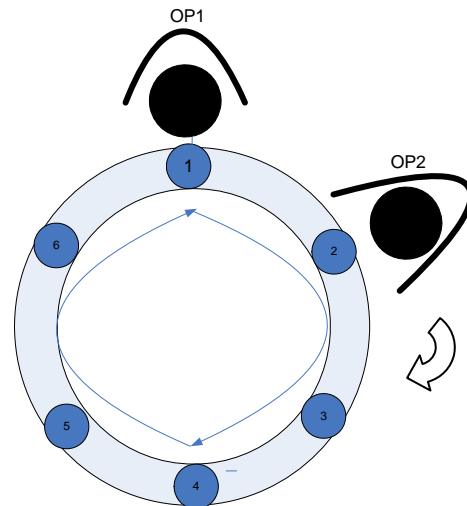


Figure 10: Location of human workers at assembly rotating round table (1st case) and direction of movement.

Using heuristic described in (Battini et. al, 2007) we obtained results which are presented in Table 2:

Table 2: Operation time of numerical example.

	OP 1	OP2	Cycle	Turns
1	Station 1	Station 2	53	2
2	Station 1	Station 3	61	3
3	Station 2	Station 3	53	2
4	Station 1	Station 4	56	3
5	Station 2	Station 4	61	3
6	Station 3	Station 4	53	2
7	Station 1	Station 5	58	3
8	Station 2	Station 5	56	3
9	Station 3	Station 5	61	3
10	Station 4	Station 5	53	2
11	Station 1	Station 6	70	2
12	Station 2	Station 6	58	3
13	Station 3	Station 6	56	3
14	Station 4	Station 6	61	3
15	Station 5	Station 6	53	2

The best average cycle time for assembly rotating round table is 53 and it occurs always when Operator 1 and Operator 2 are located next to other. In this case we need to execute only two turns. The final solution is: Operator 1 executes tasks 1 and 6 and Operator 2 executes tasks 2, 3, 4, 5, 7 and 8. Additionally we can calculate the time when final products is ready to unload from assembly system. In our case the ready product leaves the system in 216 units of time. We should remember that assembly rotating system is mostly effective in case when product demand is equal to the total number of stations.

7 CONCLUSIONS

In the paper two assembly systems were considered. First assembly lines were presented. Next assembly rotating round table was shown. The problems seems interesting for low product demand. Known procedures of solving balance of line structures allow to get very easy optimal or near optimal solution for two stations line. Investigated assembly rotating round table allows to quick changes of assembling different product. Heuristic procedure improves the result of average cycle time from 70 to 53. This kind of assembly table takes benefits from layout described in section 4 dealing with their disadvantages such as monotony, boredom, operators overload and communication. Different measures of final result (smoothness index, line efficiency, line time or average cycle time) simplify the choice of the most appropriate solution. We should underline that assembly rotating round table system don't need additional sequencing procedure. Mixed product assembly deals with many precedence relations but we choose only this one

with maximal number of tasks and connection. Therefore we calculated the balance of whole model with maximal task time operations. It allows to choice appropriate cycle time of turn. In serial lines we need to sequence the mix product model and sometimes to stop the line (different model cycle time) or to add additional parallel station.

This research was supported in part by grant of Ministry of Science and Higher Education BK 209/Rau1/2009 t.5

REFERENCES

- Aase, G. R., Olson, J. R., Schniederjans, M. J., 2004, U-shaped assembly line layouts and their impact on labor productivity: an experimental study, *European Journal of Operational Research*, 156, 698-711
- Askin, R. G., Zhou, M., 1997, A parallel station heuristic for the mixed-model production line balancing problem, *International Journal of Production Research*, 35(11), 3096-3106
- Beker, C., Scholl, A., 2005, A survey on problems and methods in generalized assembly line balancing, *European Journal of Operational Research*, 168, 694-715
- Bartholdi J. J., 1993. Balancing two-sided assembly lines: a case study, *International Journal of Production Research*, 31(10), 2447-2461
- Battini, D., Facio, M., Ferrari, E., Persona, A., Sgarbossa, F., 2007, Design Configuration for a Mixed Model Assembly System in Case of Low Product Demand, *The International Journal of Advanced Manufacturing Technology*, 34(1), 188-200.
- Bukchin, J, Dar-El, M, Rubinovitz, J., 2002, Mixed model assembly line design in a make-to-order environment, *Computers & Industrial Engineering*, 41, 405-421
- Heike, G., Ramulu, M., Sorenson, E., Shanahan, P., Moinzadeh., K, 2001, Mixed model assembly alternatives for low-volume manufacturing: the case of the aerospace industry, *International Journal of Production Economics*, 72, 103-120
- McMullen, P. R., 1997, A heuristic for solving mixed-model line balancing problem with stochastic task durations and parallel stations, *International Journal of Production Economics*, 51(1), 77-190
- Salveson, M. E., 1955. The assembly line balancing problem, *Journal of Industrial Engineering*, 62-69
- Scholl, A., 1998. Balancing and sequencing of assembly line, *Physica- Verlag*
- Sauer, G. A., 1998, Designing parallel assembly lines. *Computer Industrial Engineering*, 35(3-4), 467-470
- Sury, R. J., 1971. Aspects of assembly line balancing, *International Journal of Production Research*, 9, 8-14

CONTROL STRATEGY OF CONSTANT MILLING FORCE SYSTEM AND METAL REMOVAL RATE MAXIMIZATION

Franc Cus, Joze Balic and Uros Zuperl

*Faculty of Mechanical Engineering, University of Maribor, Smetanova 17, 2000 Maribor, Slovenia
{franc.cus, joze.balic, uros.zuperl}@uni-mb.si*

Keywords: Adaptive Force Control, Neural Control Scheme, Machining, End-milling.

Abstract: An adaptive control system in conjunction with off-line optimization is built which controlling the cutting force and maintaining constant roughness of the surface being milled by digital adaptation of cutting parameters. In this way it compensates all disturbances during the cutting process: tool wear, non-homogeneity of the workpiece material, vibrations, chatter etc. The basic adaptive control design is based on the control scheme (UNKS) consisting of two neural identifiers of the process dynamics and primary neural regulator.

1 INTRODUCTION

The use of computer numerical control (CNC) machining centers has expanded rapidly through the years. A great advantage of the CNC machining center is that it reduces the skill requirements of machine operators. However, a common drawback of CNC end milling is that its operating parameter such as spindle speed or feedrate is prescribed conservatively either by a part programmer or by a relatively static database in order to preserve the tool. As a result, many CNC systems run under inefficient operating conditions. For this reason, CNC machine tool control systems, which provide on-line adjustment of the operating parameters, are being studied with interest (Balic, 2000). These systems can be classified into three types: a geometric adaptive compensation (GAC) system; an adaptive control optimization (ACO) system; and an adaptive control constraints (ACC) system.

There is no controller that can respond quickly enough to sudden changes in the cut geometry to eliminate large spikes in cutting forces. Therefore, we implement on-line adaptive control in conjunction with off-line optimization. The optimization is performed with algorithm developed by researchers (Zuperl, 2004) and (Cus, 2003). In our AC system, the feedrate is adjusted on-line in order to maintain a constant cutting force in spite of variations in cutting conditions.

2 NEURAL FORCE CONTROL STRATEGY

The overall force control strategy consists of optimizing the feedrates off-line, and then applying on-line adaptive control during the machining process. The basic idea of this design is to merge the off-line cutting condition optimization algorithm and adaptive force control (Figure 1). Based on this new combined control system, very complicated processes can be controlled more easily and accurately compared to standard approaches. The objective of the developed combined control system is keeping the metal removal rate (MRR) as high as possible and maintaining cutting force as close as possible to a given reference value. Combined control system is automatically adjusted to instant cutting conditions by adaptation of feedrate.. Sequence of steps for on-line optimization of the milling process is presented below:

1. The recommended cutting conditions are determined by ANfis software (Mursec, 2000) for selecting the recommended cutting conditions.
2. Optimization of recommended cutting conditions by PSO optimization.
3. The pre-programmed feedrates determined by off-line optimization algorithm are sent to CNC controller of the milling machine .
4. The measured cutting forces are sent to neural control scheme.

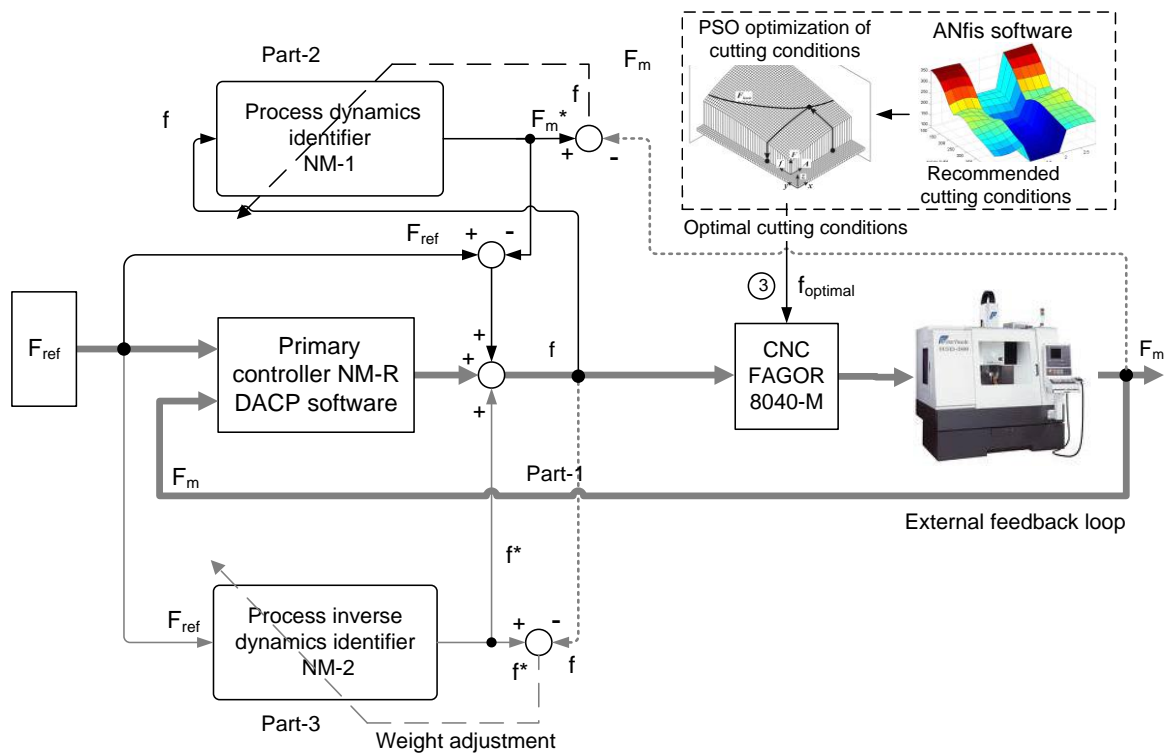


Figure 1: Adaptive force control combined with off-line optimization.

5. Neural control scheme adjusts the optimal feedrates and sends it back to the machine.
6. Steps 1 to 3 are repeated until termination of machining.

The adaptive controller adjusts the feedrate by assigning a feedrate override percentage to the CNC controller on a 4-axis Heller, based on a measured peak force. The actual feedrate is the product of the feedrate override percentage (DNCFRO) and the programmed feedrate. The fundamental control principle is based on the neural control scheme (UNKS) consisting of three parts (Figure 1). The first part is the loop known as external feedback (conventional control loop). The feedback control is based on the error between the measured (F_m) and desired (F_{ref}) cutting force. The primary feedback controller is a neural network (NM-R). The second part (NM-1) acts as the process dynamics (cutting dynamics) identifier. The third part of the system is neural network 2 (NM-2). The NM-2 learns the process inverse dynamics. The UNKS operates according to the following procedure. The sensory feedback is effective mainly in the learning stage. This loop provides a conventional feedback signal to control the process. During the learning stage, NM-2 learns the inverse dynamics. As learning proceeds, the internal feedback gradually takes over the role of

the external feedback and primary controller. Then, as learning proceeds further, the inverse dynamics part will replace the external feedback control. The final result is that the plant is controlled mainly by NM-1 and NM-2 since the process output error is nearly zero.

3 EXPERIMENTAL SET-UP

The data acquisition equipment consists of dynamometer, fixture and software module. The cutting forces were measured with a piezoelectric dynamometer (Kistler 9255) mounted between the workpiece and the machining table. The interface hardware module consists of a connecting plan block, analogue signal conditioning modules and a 16 channel A/D interface board (PC-MIO-16E-4). In the A/D board, the analogue signal will be transformed into a digital signal so that the LabVIEW software is able to read and receive the data. The ball-end milling cutter with interchangeable cutting inserts of type R216-16B20-040 with two cutting edges, of 16 mm diameter and 10° helix angle was selected for machining. The cutting insert material is P10-20 coated with TiC/TiN, designated GC 1025.


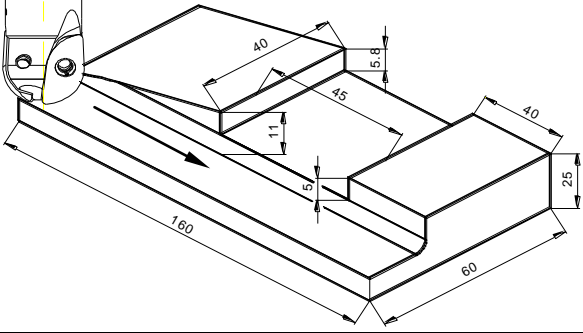
<p>Experiment: Prismatic Workpiece</p> 	
<p>Test_A Constant feedrate</p>	<p>Cutting conditions: Feedrate: 0.08mm/tooth, Cutting speed: $v=80\text{m/min}$, Pre-programmed axial depth of cut $A_D=2\text{ mm}$, Radial depth of cut $R_D=4\text{mm}$, $F_{ref}=270\text{N}$ (Kopac, 2002), Result: Figure: 3a.</p>
<p>Test_B Proposed adaptive control system</p>	<p>Starting feedrate: 0.08mm/tooth, Allowable adjusting rate: 00.8 - 0.20 mm/teeth, Cutting speed: $v=80\text{m/min}$; Result: Figure: 3b.</p>

Figure 2: Plan of experiment; Cutting conditions for prismatic workpiece.

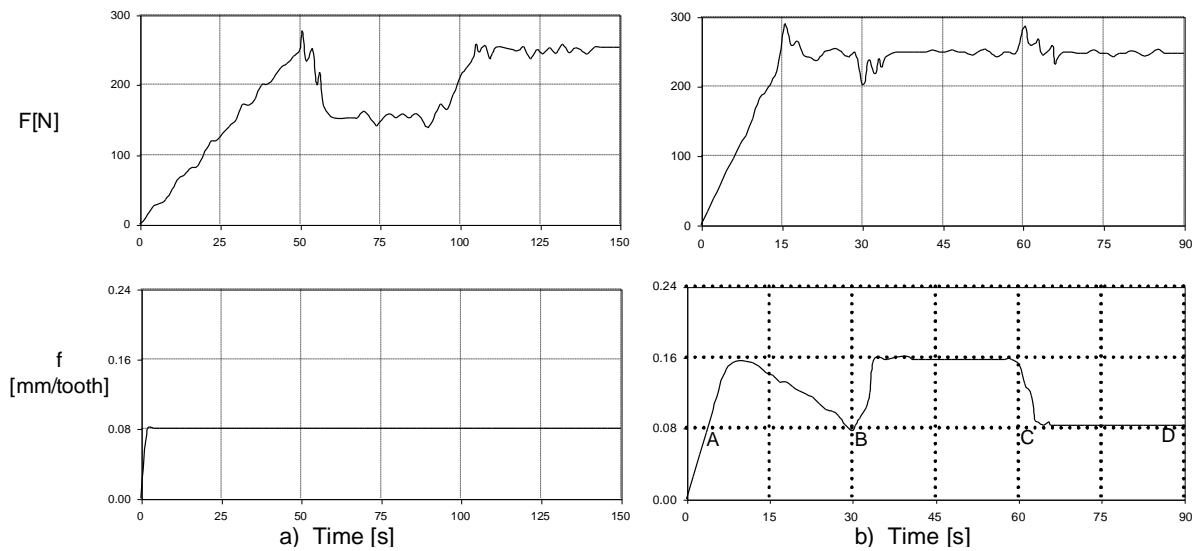


Figure 3: Response of MRR, resulting cutting force and feedrate. a) Conventional milling-Test_A. b) Milling with proposed adaptive control system-Test_B.

4 PROGRAM DACP

The program for digital adaptation of cutting parameters (DACP) is developed by software packet LabVIEW 7. During developing of DACP program the following requirements are taken into consideration:

- It must established communication between dynamometer and data acquisition card,
- Enable the selecting of measuring channels and calibration of measuring system,
- Establish communication with CNC controls,
- Actuate visual and sound signals in case of cutting tool overloading.

Control panel of DACP consists of three main parts. Upper part of panel is a monitoring part. Monitoring part has switches which enable user to define scanning parameters, measuring ranges, and accuracy of measuring.

Middle part is a control part. It consists of four buttons for controlling the milling process. Reference cutting force and desired surface roughness is set into the system by two graphical slides. All important information about communication state is displayed at the bottom of control panel. A communication module was developed to communicate with CNC via an RS-232 serial line.

5 EXPERIMENTAL TESTING OF CONTROL SYSTEM

The stability and robustness of the proposed control strategy is verified by experiments on a CNC milling machine for Ck 45 and 16MnCrSi5 XM steel prismatic workpieces with variation of axial cutting depth. Details of the experimental conditions and the dimensions of the workpiece are shown in Figure 2. Feedrates for each cut are first optimized off-line, and then machining runs are made with controller action. The first test is conventional cutting with the constant feedrate (Test_A). In the second test, the proposed control system was applied to demonstrate its performance (Test_B). The parameters for adaptive control are the same as for the experiments in the conventional milling (Zuperl, 2003).

Figure 3 is the response of the cutting force and the feedrate when the cutting depth is changed. It shows the experimental result where the feedrate is adjusted on-line to maintain the maximal cutting force at the desired value.

6 RESULTS AND DISCUSSION

As compared to most of the existing end milling control systems (Chen, 2006), the proposed adaptive system has the following advantages: 1. the computational complexity of UNKS does not increase much with the complexity of process; 2. the learning ability of UNKS is more powerful than that of conventional adaptive controller; 3. UNKS has a generalisation capability; 4. insensitive to changes in workpiece geometry, cutter geometry, and workpiece material; 5. cost-efficient and easy to implement; and 6. mathematically modeling-free.

Comparing the Figure 3a to Figure 3b, the cutting force for the neural control milling system is maintained at about 250N, and the feedrate of the adaptive milling system is close to that of the conventional milling from point C to point D. From point A to point C the feedrate of the adaptive milling system is higher than for the classical CNC system, so the milling efficiency of the adaptive milling is improved.

The time analysis for conventional and adaptive control system has been carried out. By adaptive control system of time saving of 40% with one cut was reached. The complete machining requires 15 cuts; thus machining of a simple workpiece is shortened for 155 seconds.

The system remains stable in all experiments, with little degradation in performance. The results

reached are in accordance with the objectives of researches, according to which the controlled cutting force must not deviate from the desired value for more than 10% (Zuperl, 2005).

7 CONCLUSIONS

In this paper, an intelligent control algorithm that controls feedrate is proposed to regulate the cutting force. On the basis of the cutting process modeling, off-line optimization and neural control scheme (UNKS) the combined system for off-line optimization and adaptive adjustment of cutting parameters is built. This is an adaptive control system controlling the cutting force and maintaining constant roughness of the surface being milled by digital adaptation of cutting parameters.

In order to check the applicability of the adaptive control algorithm, cutting experiments were carried out under various cutting conditions, different tool diameters and different work materials. Experiments have confirmed efficiency of the adaptive control system, which is reflected in improved surface quality and decreased tool wear.

REFERENCES

- Balic, J., 2000. A new NC machine tool controller for step-by-step milling. *Int. j. adv. manuf. technol.* 8, pp. 399-403.
- Chen, C., Zhibin, M., (2006). An intelligent approach to non-constant feed rate determination for high-performance 2D CNC milling. *International Journal of Manufacturing Technology and Management.* pp. 219-236.
- Cus, F., Balic, J., 2003. Optimization of cutting process by GA approach. *Robot. comput. integr. manuf.*, 19, pp. 113-121.
- Kopac, J., 2002. Režalne sile in njihov vpliv na gospodarnost obdelave = Cutting forces and their influence on the economics of machining. *Stroj. vestn.* 3, pp. 121-132.
- Mursec, B., Cus, F., Balic, J., 2000. Organization of tool supply and determination of cutting conditions. *J. mater. process. Technol.*, 100, pp. 241-249.
- Zuperl, U., Cus, F., 2004. A determination of the characteristic technological and economic parameters during metal cutting. *Stroj. vestn.* pp. 252-266.
- Zuperl, U., Cus, F., 2003. Optimization of cutting conditions during cutting by using neural networks. *Robot. comput. integr. Manuf.*, 19, pp. 189-199.
- Zuperl, U., Cus F., 2005. Tool cutting force modeling in ball-end milling using multilevel perceptron. *J. mater. process. technol.*

THE VALUE OF INFORMATION SHARING IN A SERIAL SUPPLY CHAIN WITH CENTRALISED AND DECENTRALISED DECISION

Mansour Rached, Zied Bahroun, Belhassen Zouari
LIP2, FST, 2092, Manar 2 Tunis, Tunisia
mansour.rached@insa-lyon.fr, {zied.bahroun, belhassen.zouari}@fst.rnu.tn

Armand Baboli, Jean-Pierre Campagne
Université de Lyon, LIESP, INSA-Lyon, F-69621, Villeurbanne, France
{arman.baboli, jean-pierre.campagne}@insa-lyon.fr

Keywords: Information sharing, Supply chain, Downstream, Upstream information, Centralised, Decentralised decision.

Abstract: In this paper, we present a simulation based model in order to study the effects of information sharing in a serial supply chain. This chain is multi-product and multi-echelon. Our approach focuses on the study of two shared information simultaneously. The first one is the replenishment leadtime coming from the upstream and the second one is the customers' demand coming from the downstream of the supply chain. Thus, we present four scenarios of information sharing. The demand is supposed normally distributed and the leadtime is random. We develop a cost model consisting of holding, ordering, penalty and transportation costs. The difference of the optimal costs between each studied scenario represents the performance indicator of the information sharing. Two different decisions are considered in our work: centralised and decentralised. The developed model for each studied situation is solved by ILOG CPLEX integrated in a JAVA program. To conclude, the results of our numerical experimentations are analysed.

1 INTRODUCTION

For many enterprises, the supply chain management has become an important element of strategic advantage to gain a competitive edge over their competitors. In the severe competition times, the enterprises want to obtain competition advantage, they must carry on the cooperation with the other echelons in the same supply chain, and try to establish win-win cooperation relationship. The information sharing is the foundation of the cooperation between different links of supply chain, and also is one main origin of the supply chain advantages.

In this context, researches are numerous. The existing works deal with the study of different types of information sharing in different circumstances. Namely, the studied decision and the source of the share, etc. The centralised decision represents the most treated case (Chu and Lee, 2006), (Rached *et al.*, 2009). The decentralised decision is treated in

(Birendra *et al.*, 2007), (Laux *et al.*, 2004), (Li *et al.*, 2006). However, (Zhao and Qui, 2007) considered simultaneously centralised and decentralised decision. Concerning the provenance of the information to share, on the one hand, many papers have treated the case of downstream information sharing (Agrawal *et al.*, 2008), (Hsiao and Shieh, 2006), (Li and Zhang, 2008). On the other hand, (Chen and Yu, 2005), (Jia *et al.*, 2007) and (Mehrabi *et al.*, 2007) presented shared information which is coming from upstream. However, few papers considered the case of shared information coming simultaneously from the upstream and the downstream (Birendra *et al.*, 2007), (Rached *et al.*, 2009). The solver CPLEX is used in many recent works including a study of optimisation and the information sharing in supply chains. In (Li and Zhang, 2008) authors present an objective function which includes a cost formulation solved using GAMS-CPLEX. In (Lehoux *et al.*, 2007), authors presented seven models aiming to study the value of collaboration between a supplier and his customer.

To compare and analyse their developed models, they use the solver CPLEX.

According to the above review, we propose to study the effects of two information shared simultaneously upon the supply chain performance with several scenarios and circumstances. The remainder of this paper is organised as follows. Section 2 reviews the structure of studied supply chain and introduces the different studied scenario. Section 3 presents the simulation and analysis results. Finally, conclusions are made in Section 4.

2 STRUCTURE OF STUDIED SUPPLY CHAIN AND THE RESOLUTION METHOD

The supply chain structure in this paper is a four-echelon model Figure 1, which includes several customers, one retailer, one warehouse and one supplier. There are various items of information which can be shared such as inventory level, production capacity, etc. The shared information refers to costumers' demand coming from the downstream of the supply chain and replenishment leadtime coming from the upstream. We assume that the costumers' demand is normally distributed, the leadtime between the supplier and the warehouse is random and the leadtime between the warehouse and the retailer is fixed for all periods.

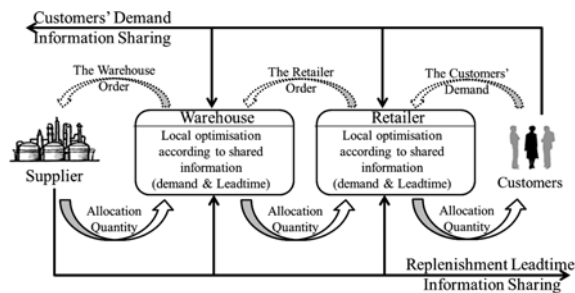


Figure 1: Studied supply chain with information and materiel flows.

We study the sharing of two information simultaneously, which are the demand coming from the downstream of supply chain and the leadtime coming from the upstream.

To investigate the effect of information sharing upon supply chain performance, for each studied decision, four scenarios are designed with respect to the two types of information mentioned above.

In the case of decentralised decision, the retailer performs a local optimisation of its costs before

placing their orders. According to this order quantity, the warehouse proceeds to a second local cost optimisation to place an order to the supplier.

According to the centralised decision presented in Figure 2, the four studied scenarios are as follows:

- Case of Replenishment Leadtime and Demand Information Sharing: At the beginning of each period, the decision maker (warehouse) is informed about the exact replenishment leadtime and the customers' demand of the current period. So, he calculates the order quantity witch minimise the total system cost according to the replenishment leadtime and the exact demand of all periods.
- Case of Replenishment Leadtime Information Sharing Only: The order quantity is calculated by the decision maker according to the exact replenishment leadtime and independently of the costumers' demand.
- Case of Demand Information Sharing Only: In this case, the warehouse calculates the order quantity according to the exact costumers' demand and independently of replenishment leadtime.
- Case of No Information Sharing: The decision maker cannot choose the optimal order quantity according to the replenishment leadtime and/or the costumers' demand of each period. So, he uses fixed values of both information for all periods to calculate the optimal order quantity.

3 EXPERIMENTAL RESULTS

In our simulation and in order to study the sensibility and the robustness of our formulation, we use different combinations of replenishment leadtime and costumers' demand in each scenario as follows:

- Small leadtime $L_t = [1 \text{ to } 4]$, noted $L1^{st}$;
- Large leadtime $L_t = [1 \text{ to } 12]$, noted $L2^{nd}$;
- High leadtime $L_t = [8 \text{ to } 12]$, noted $L3^{rd}$;
- First mean value of demand (high demand) $\mu = 380$, noted $D1^{st}$;
- Small demand $\mu = 100$, noted $D2^{nd}$;
- Medium demand $\mu = 230$, noted $D3^{rd}$.

We use in each simulation the same standard deviation $\sigma = 80$.

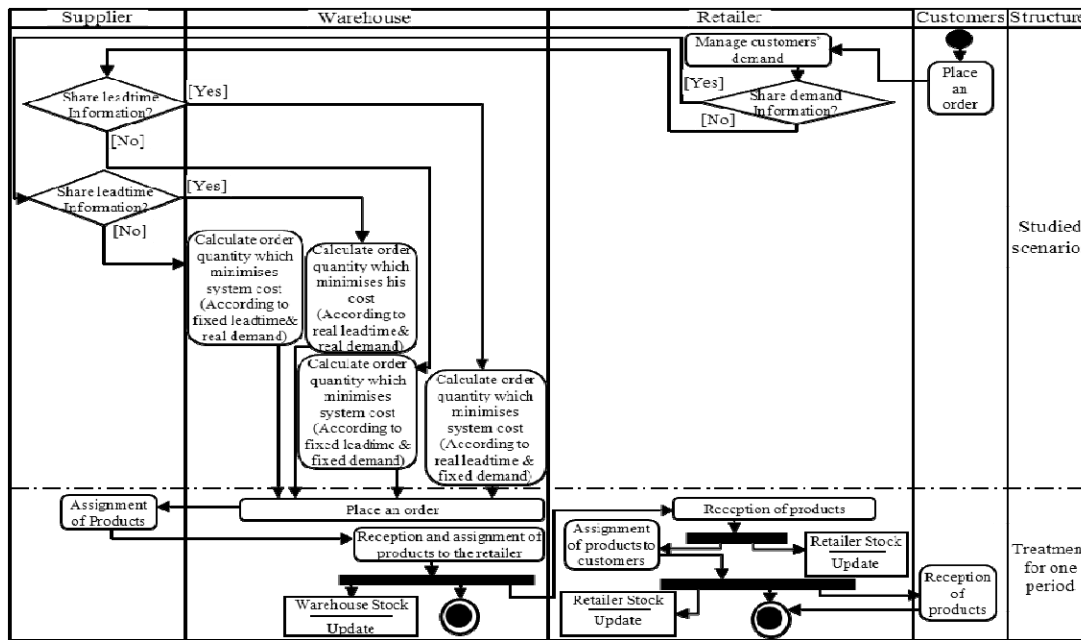


Figure 2: Activity diagram integrating all studied scenarios in a centralised decision.

In our work we deal with the problem of optimisation based on a centralised (Table 1) and decentralised decision (Table 2) using CPLEX.

- IS: Is the case of Information Sharing;
- NIS: Is the case of No Information Sharing;
- DemI: Demand Information;
- LtI: Leadtime Information.

Table 1: Total system cost in centralised decision.

			DemI		
			TSC (10 ³)		
			IS	NIS	
LtI	L1 st	D1 st	IS	5,1347	8,4099
			NIS	6,1216	11,1132
		D2 nd	IS	3,9821	5,2154
			NIS	4,7612	7,1256
		D3 rd	IS	4,4782	6,8154
			NIS	5,9373	10,0347
	L2 nd	D1 st	IS	5,1277	8,2367
			NIS	7,4385	10,9521
		D2 nd	IS	3,9482	5,9610
			NIS	4,5659	6,9581
		D3 rd	IS	6,0274	7,7147
			NIS	6,8416	8,2618
L3 rd	D1 st	IS	7,1639	9,2184	
		NIS	8,1374	10,3514	
	D2 nd	IS	4,1132	4,8952	
		NIS	4,5173	5,5901	
	D3 rd	IS	3,2467	5,1458	
		NIS	3,8962	7,7259	

Table 2: Total system cost in decentralised decision.

			DemI		
			TSC (10 ³)		
			IS	NIS	
LtI	L1 st	D1 st	IS	6,2704	10,5806
			NIS	9,2921	12,8980
		D2 nd	IS	5,2502	6,7198
			NIS	6,4988	7,5044
		D3 rd	IS	5,9443	7,8265
			NIS	6,3447	10,5558
	L2 nd	D1 st	IS	6,9277	9,5627
			NIS	8,8154	12,9171
		D2 nd	IS	5,2502	7,2198
			NIS	6,4059	8,2461
		D3 rd	IS	6,2276	8,0747
			NIS	6,7546	8,7996
L3 rd	D1 st	IS	7,8686	9,8018	
		NIS	8,4053	11,0514	
	D2 nd	IS	5,3212	6,7635	
		NIS	5,8177	6,9859	
	D3 rd	IS	5,2606	7,5315	
		NIS	5,9153	9,9776	

In Table 1 and Table 2, we present nine values of simulations. Illustrated by the different combinations of (L1st, L2nd and L3rd) and (D1st, D2nd and D3rd). For each case we present TSC (10³) for four scenarios of customers' demand and replenishment leadtime information sharing.

Based on the simulations in Table 1, we study the centralised decision. Regarding the total system cost, the results show an average percentage of 21.12% of reduction of logistic cost in the case of

leadtime information sharing compared to the scenario of no information sharing. Whereas the demand information sharing compared to the scenario of no information sharing presents an average of 33.15% reduction of logistic cost. When the two information are simultaneously shared, we obtain 44.66% of of logistic cost reduction compared to the case of no information sharing. For the decentralised decision illustrated by Table 2, Compared to the case of no information sharing, we can deduce 16.70% and 27.75% of reduction of logistic cost concern, respectively, the case of leadtime information sharing and, the case of demand information sharing. When the two studied information are simultaneously shared, we obtain 38.92% of reduction of logistic cost compared to the case of no information sharing. In the case of no information sharing, the centralised decision presents 12.16% of reduction of logistic cost compared to the case of decentralised decision. Thus, the centralised decision compared to the case of decentralised decision presents a percentage at 16.83%, 18.72% and 20.43% of reduction of logistic cost concern, respectively, the case of leadtime information sharing, the case of demand information sharing and the case of two information shared simultaneously. In the studied circumstances, we can conclude that the information sharing and the centralised decision present an advantage in terms of reduction of logistic cost compared to the case of decentralised decision.

4 CONCLUSIONS

In this paper, we are interested to the evaluating of information sharing in supply chain. We treated in particular the case of multi-product multi-echelon supply chain. We studied the value of information, specially two information shared simultaneously. The first information comes from the downstream (demand) and the second come from the upstream (leadtime) of supply chain. Moreover, the effect of two kinds of decision has been also studied (centralised and decentralised decision). We used the traditional replenishment policies as a logistic reference cost. The numerical experimentation shows that the information sharing allows reducing the total logistic costs (transportation and storage costs). Moreover, the centralised decision is more beneficial in terms of reduction of logistic cost compared to the decentralised decision.

REFERENCES

- Agrawal, S., Sengupta, R.N., Shanker, K., 2008. Impact of information sharing and lead time on bullwhip effect and on-hand inventory. *European Journal of Operational Research*. 192(2), 576-593.
- Birendra, K.M., Srinivasan, R., Xiaohang, Y., 2007. Information sharing in supply chains: Incentives for information distortion. *IIE Transactions*. 39(9), 863-877.
- Chen, F., Yu, B., 2005. Quantifying the Value of Leadtime Information in a Single-Location Inventory System. *Manufacturing & Service Operations Management*. 7(2), 144-151.
- Chu, W.H.J., Lee, C.C., 2006. Strategic information sharing in a supply chain. *European Journal of Operational Research*. 174(3), 1567-1579.
- Hsiao, J.M., Shieh, C.J., 2006. Evaluating the value of information sharing in a supply chain. *International Journal of Advanced Manufacturing Technology*. 27(5-6), 604-609.
- Jia, Q., Guo, W., Li, B., 2007. Study On The Effect Of Information Sharing Strategy To Complex Supply Chain System Based On Multi-Dimension View By Simulation. *Wireless Communications, Networking and Mobile Computing WiCom'07*, 4847-4850.
- Laux, J.S.K., Huang, G.Q., Mak, K.L., 2004. Impact of information sharing on inventory replenishment in divergent supply chains. *International Journal of Production Research*. 42(5), 919-941.
- Lehoux, N., D'Amours, S., Langevin, A., 2007. Cadre d'évaluation de la valeur de la collaboration: modélisation de la relation entre un fournisseur et son client pour le secteur des pâtes et papiers. *7e Congrès International de Génie Industriel*.
- Li, J., Sikora, R., Shaw, M.J., Tan, G.W., 2006. A strategic analysis of inter organizational information sharing. *Decision Support Systems*. 42(1), 251-266.
- Li, L., Zhang, H., 2008. Confidentiality and information sharing in supply chain coordination. *Management Science*. 54(8), 1467-1481.
- Mehrabi, A., Baboli, A., Campagne, J.P., 2007. Evaluer la valeur de partage d'information de délais dans une chaîne logistique avec l'algorithme génétique. *7e Congrès international de génie industriel*.
- Rached, M., Bahrour, Z., Baboli, A., Campagne, J.P., Zouari, B., 2009. A method to evaluate downstream and upstream information sharing using the genetic algorithm. *International Conference on Computers & Industrial Engineering CIE'39*. 1635-1640.
- Zhao, X., Qiu, M., 2007. Information Sharing in a Multi-Echelon Inventory System. *Tsinghua Science & Technology*. 12(4), 466-474.

**SPECIAL SESSION ON INTELLIGENT
VEHICLE CONTROLS & INTELLIGENT
TRANSPORTATION SYSTEMS**

FULL PAPERS

WHEELED-ROBOT NAVIGATION WITH VELOCITY UPDATING ON ROUGH TERRAINS

Farid García and Matías Alvarado

*Centro de Investigación y de Estudios Avanzados-IPN, Departamento de Computación
Av. Instituto Politécnico Nacional 2508, San Pedro Zacatenco, CP 07360, México DF, México
farid@computacion.cs.cinvestav.mx, matias@cs.cinvestav.mx*

Keywords: Autonomous Navigation, Roughness Recognition, Velocity Updating, Wheeled-Robots.

Abstract: For navigation on outdoor surfaces, usually having different kind of roughness and soft irregularities, this paper proposal is that a wheeled robot combines the gradient method for path planning, alongside it adjusts velocity based on a multi-layer fuzzy neural network; the network integrates information about the roughness and the soft slopes of the terrain to compute the navigation velocity. The implementation is simple and computationally low-cost. The experimental tests show the advantage in the performance of the robot by varying the velocity depending on the terrain features.

1 INTRODUCTION

Robotic autonomous navigation throughout outdoor terrains is highly complex. Obstacle detection and avoidance for no collision as well as the terrain features information for no slides are both required. Environment data must be quick and accurately processed by the robot's navigation systems for a right displacing. Besides, when information from human remote controllers is not quick available, the autonomous robots should be equipped for convenient reactions, particularly in front of unpredicted circumstances. Actually, by moving on outdoors, the autonomous robot's velocity control regarding the terrain features, beyond the obstacle location and avoidance, it has been few attended and it is a weakness for efficient and safe navigation nowadays.

For wheeled-robots navigation on terrains, it is necessary data about the surface features such that automated safe navigation is ensured. The feature which this work focuses is the surface roughness where the robot moves on. The robot's velocity during real navigation depends on the terrain roughness.

Outdoor autonomous robots are particularly relevant employed for terrain exploration missions. The terrain difficulties of soon system planets –like Mars– to move through soil, rocks and slopes, requires the usage of robots with the highest degree of autonomy to overcome such difficulties. In Earth

exploration missions where human lives may be in dangerous circumstances, the autonomous robots are as well required. For instance, search of explosive minas, active volcano craters exploration to determine the eruption risk.

Kelly and Stentz (1998) propose a navigation system for outdoors robots which includes perception, mapping and obstacle avoidance. Regarding the environment perception, Lambert et al. (2008) introduces a probabilistic modelling useful to avoid or to mitigate eventual collisions, which is used for updating a robot braking action. Selekwa et al. (2008) and Ward & Zelinsky (2000) addressed the navigation and path planning of an autonomous robot which varies the velocity according to the proximity of obstacles detected by infrared sensors.

So far, all the referred works on outdoor autonomous robots do not include in their proposals information about terrain surface roughness during navigation. In this work, two algorithms are implemented for robot autonomous navigation, one for path planning and the other for velocity updating regarding the terrains features. The present proposal is tested with a small wheeled-robot moving over outdoors terrains containing grass, ground, garden sand and soil, as the ones in Figure 1 and Figure 2. It is assumed that the robot can move on slopes with inclination angles less than 15 degrees; otherwise, the slopes are considered as obstacles, thus, the robot wraps them, in order to overcome them.

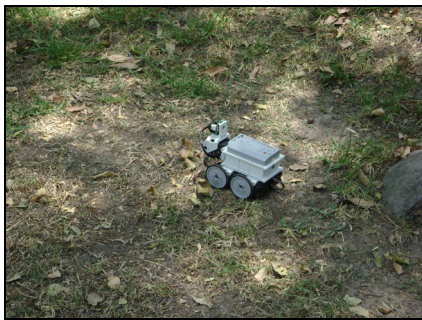


Figure 1: Robot's navigation outdoor surface.



Figure 2: Test environment.

Surface textures are captured via artificial vision, after image processing the estimation of the terrain roughness as well as the slopes inclinations are gotten. Then, the algorithms output indicates the velocity the robot can achieve. Bright and uniform lighting during navigation is required to guaranty consistent roughness recognition; therefore the presence of shadows, which treatment is a hard task to pattern recognition (Kahraman and Stegmann, 2006), is out of the scope of this work.

During outdoors navigation, human drivers estimate the convenient vehicle velocity by regarding their previous experience when driving on similar terrain textures. In other words, humans estimate how rough, in average, the terrain is, instead if specific texture details are recognized.

Human drivers that navigate on uneven terrains do not need to know about specific details but on the textures appearance average. The average recognition of ranges of textures as the humans learn is the behave experience to be mimicked and implemented to strength the robots' navigation abilities. The algorithms for path planning differ depending on the type of application, exploration on unknown terrains (Seraji and Howard, 2002), car navigation on roads (Sun et al., 2006), planet exploration (Seraji and Werger, 2007) or indoor navigation (Ward and Zelinsky, 2000), or if the

environment is either dynamic (Kim et al., 2007) or static (Wang and Liu, 2005).

For our purpose, the robot moves on the calculated path by adjusting its velocity depending on the terrain features. The path planning algorithm called *gradient method*, in static environments, recalculates the path in real time whenever an obstacle is found (Konolige, 2000). The gradient method is integrated for our navigation proposal.

The rest of the article is organized as follows: Section 2 summarizes the closest antecedents in the field of autonomous navigation; then, the method and architecture of the fuzzy neural network for speed updating, together with the gradient method for path planning are introduced. Section 3 describes the integration of both algorithms for wheeled-robot navigation, together with the tests and experimental results. Discussion in Section 4, then the paper ends with conclusions in Section 5.

2 VELOCITY UPDATING BY FUZZY NEURAL NETWORK

2.1 Terrain Roughness Recognition

The classification of terrain roughness has almost no received attention, and just recently is being a bit more attended. For instance, Larson et al. (2005) analysis the terrain roughness by means of spatial discrimination which then is (meta-) classified. Seraji and Howard (2002) assess the navigation strategy with the terrain's features of roughness, slopes and discontinuity. Ishigami et al. (2007) generate a path over a rough terrain with a terrain-based criterion function, and then the robot is controlled so as to move on the chosen path. In Brooks and Iagnemma (2009) do roughness recognition by using artificial vision, so recognition of novel textures is later to off-line recognition training from sample texture. Pereira et al. (2009) plotted maps of terrains incorporating roughness information that is based on the measurement of vibrations occurring in the suspension of the vehicle; this online method can recognize textures at the moment the vehicle passes over them, what is a limitation for velocity updating.

For velocity updating according to the terrain features, our proposal sets to imitate as human beings do. For safe navigation on irregular terrains, the human's velocity estimation is via imprecise but enough surface texture recognition. When a human driver notes a new texture, he uses his experience to

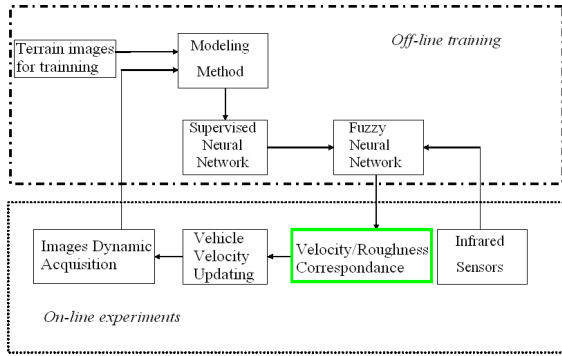


Figure 3: The proposal diagram.

estimate how rough the texture is, and then he decides the car driving speed without slide risks.

As illustrated in the diagram of Figure 3, in the first step, the terrain’s textures are neural-net-clustered in a roughness meta-class: a Supervised Neural Network (SNN) classifies textures forming the meta-class; then, a Fuzzy Neural Network (FNN) extends the supervised one, matches each terrain roughness with the corresponding robot velocity meanwhile the robot navigates safely.

Slopes are detected by two infrared sensors. One infrared sensor is located in the frontal part of the robot does parallel ray projection to the robot’s motion; the other sensor projects its ray directly to the floor perpendicular to the first sensor. The inclination angle of slopes is computed by trigonometric operations.

The off-line and on-line steps to adapt velocity regarding the terrains roughness and the inclination slopes while navigating are next described:

Roughness Identification (Off-line Training)

- 1) Select representative images of the terrain textures, where the robot moves on.
- 2) Characterize the texture using the Appearance Based Vision method which computes the principal components of the images distribution.
- 3) Train the SNN with the texture-roughness relationship previously established by the human expert driver.
- 4) Train the FNN to determine the velocity regarding the texture roughness as well as the inclination angle of slopes, according to an expert driver’s directives, make the fuzzy sets and the inference IF-THEN rules system.

Velocity Updating and Robot’s Motion (On-line Steps)

- 5) Acquisition of terrain images with the robot’s camera.

- 6) The SNN classifies the texture and assigns its roughness, this data is forwarded to the FNN.
- 7) The FNN inputs are both, the texture roughness and the slope inclination angle (to determine if the robot can pass on the slope, or should move around it).
- 8) With the texture roughness and slope inclination data, the FNN updates the velocity. The robot’s mechanical control system adjust the velocity.
- 9) The cycle is repeated as the robot moves, and the velocity is cycle updated.

2.2 The Fuzzy Neural Network

This section introduces the five-layer fuzzy neural network, whose output sets the velocity the robot can achieve safely. The terrain features recognition followed by the robot velocity tuning is shown in Figure 4. The roughness and slope input data are assessed and then used to adjust the robot’s velocity, that is the FNN output data, see Table 1. The FNN first layer inputs are the slope size and the texture roughness, the second layer sets the terms of input membership variables, the third sets the terms of the rule base, the fourth sets the term of output membership variables, and in the fifth one, the output is the robot’s velocity. The textures roughness is classified in three fuzzy sets, High (H), Medium (M) and Low (L). The inclination angles of slopes are classified in six fuzzy sets: Plain (PI), Slightly Plain (SP), Slightly Sloped (SS), Moderato Sloped (MS), High Slope (HS) and Very High (VH). The FNN output values are either: High Velocity (HV), Moderate Velocity (MV), Low Velocity (LV) or Stop (ST). Membership functions of the input and output variables terms denote the corresponding texture roughness, slope inclination and velocity, respectively.

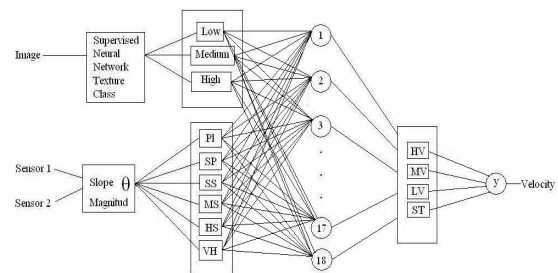


Figure 4: The Fuzzy Neural Network.

The fuzzy-making procedure maps the crisp input values to the linguistic fuzzy terms with membership values in [0,1]. In this work the trapezoid membership functions (MF) for texture

variable and the triangle MF for angle variable are respectively used. The FL inference rules governing the input - output relationship are in the Table 1. Taking X, Y, Z as variables of the respective predicates, the form of inference rules is:

IF *Slope angle* is X **AND** *Roughness* is Y **THEN** *Velocity* is Z.

The de-fuzzy procedure maps the fuzzy output from the inference mechanism to a crisp signal. When the robot finds a slope steeper than the allowed threshold, it stops, and evaluates which movement to make, whose decision concerns to path planning. The gradient method (Konolige, 2000) is integrated to present proposal.

2.3 The Gradient Method for Path Planning

The gradient method requires a discrete configuration of the navigation space in which the path cost function is sampled. At each point of the workspace, the gradient method uses a navigation function to generate a gradient field that represents the optimum path to the target point. The gradient of navigation function indicates the path direction with lowest cost, at each point in the navigation space; this optimum path to the target is continuously calculated, and is determined based on the length and the proximity to obstacles, in addition to any other criteria that may be chosen. By itself, the gradient method can lead the path with the lowest cost in static and completely unknown environments; this method is efficient for real time monitoring the movements of mobile robots equipped with laser beams.

3 THE NAVIGATION ALGORITHM

The robot autonomous navigation requires the concurrent operation of path planning and velocity estimation algorithms. The first step is to create a virtual map of the robot navigation space; hence the surface is divided into squares for providing the required detail level of space model. The next step is to calculate the optimal path between initial and goal locations using the gradient method.

After path planning, the texture recognition algorithm is turned on to determine the robot velocity. The roughness surface data in addition to information from sensors that measure the slopes

Table 1: The velocity updating fuzzy rules.

Rule No.	Input		Output
	<i>Slope angle</i>	<i>Roughness</i>	<i>Velocity</i>
1	Pl	L	HV
2	Pl	M	HV
3	Pl	H	HV
4	SP	L	MV
5	SP	M	HV
6	SP	H	HV
7	SS	L	MV
8	SS	M	MV
9	SS	H	HV
10	MS	L	LV
11	MS	M	MV
12	MS	H	MV
13	HS	L	LV
14	HS	M	LV
15	HS	H	MV
16	VH	L	ST
17	VH	M	ST
18	VH	H	LV

inclination are processed. Hence, the robot receives the instruction to move at the estimated velocity in the prior determined trajectory. If during the trip the sensors detect an obstacle or slopes with inclination greater than 15 degrees, the robot stops and the velocity estimation algorithm is turned off; the obstacle is registered and a new path to the goal location is recalculated. After that, the velocity estimation algorithm is turned on again, and the robot learns to move in the new trajectory at the estimated speed. Otherwise, i.e., if the robot does not find an obstacle on its path, then its speed is updated.

Note that the velocity estimation algorithm is not being executed all the time, but it is turned off when the robot finds an obstacle; at this circumstance, the camera records the obstacle images instead of surface texture. If the velocity estimation algorithm would not be turned off, the velocity would be estimated based on images of the obstacle texture, what is wrong; furthermore, in front of obstacle the robot should overcome the obstacle with specific movements and the velocity change is irrelevant.

The robot stops when it determines that has reached the goal location. The robot computes its location from the distance it has travelled since the initial location, by using odometry. The following list summarizes the robot navigation steps, see Figure 5:

1. Create a virtual map of robot space navigation, surface discretization,

2. Define the initial and goal locations of the robot,
3. Compute the path with the gradient method,
4. Artificial vision is turned on for texture recognition,
5. Velocity is estimated with data from textures and slopes of the surface,
6. The robot receives the order to advance along the path at the estimated velocity,
7. The robot velocity is updated when a change in texture is recognized, or if sensors detect a slope on the surface, or if both events occur,
8. If sensors detect an obstacle, or if slope inclination is greater than 15 degrees, then the robot stops and texture recognition algorithm is turned off, return to step 3; Otherwise, velocity is updated, return to step 7,
9. The robot stops when it has reached the goal location or destination.

3.1 Experimental Step

A car-like rover from the Bioloid robot transformer kit (ROBOTIS, 2010) is used, which uses a processing unit, four servomotors for power transmission to the wheels, two infrared sensors located in the robot front, and a wireless camera on top-front of the robot. The robot dimensions are 9.5 cm width per 15 cm length. In these experiments the FNN is trained with terrain textures from images in Figure 1.

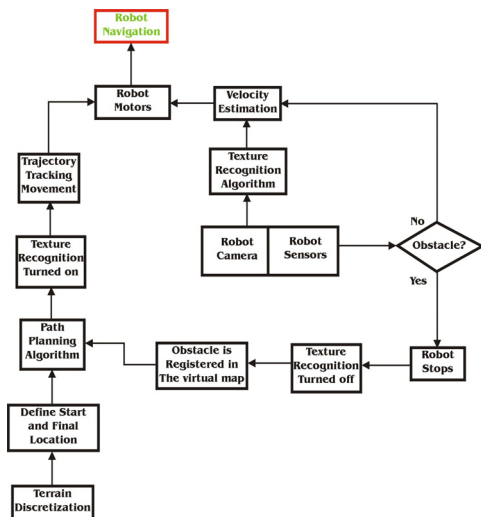


Figure 5: Path Planning and Velocity estimation algorithms running concurrently.

In this platform it is used a personal computer (PC) and the processor of the robot, to form a

master-slave architecture, wirelessly communicated. On the PC are implemented and executed the path planning and velocity estimation algorithms. The robot, on one hand, reports to the PC the sensors readings and wirelessly transmits the images captured by the camera. On the other hand, the robot performs the movements in accordance with instructions that the PC communicates it. The experiments are performed in the environment shown in Figure 2, whose area is 2.25 m², covered with dust, soil, dry leaves, branches and 2 cm-high grass; it contains rocks and small earth-mounds. The goal point is located 2.12 meters in a straight line from the initial location of the robot.

There were conducted 30 tests divided into three parts, using the path planning algorithm. In first part, the tests were performed at medium constant velocity 6.95 cm/s; in the second part at the maximum velocity the robot can reach 13.88 cm/s. The other tests were performed with velocity updating, combining path planning and velocity updating algorithms. Table 2 shows the results.

With velocity updating, both the detection of the robot environment and path planning are strengthened. By adjusting the velocity according to surface characteristics, safety increases and/or the travel time of the robot decreases. That is, if it detects that the surface is slippery then the robot slows down, although the robot spends more time to reach the goal location, the probability that the robot has an accident decreases.

3.2 Results and Navigation Performance

The common standards criterion to evaluate the performance of robots is (Dai et al., 2007), (Matthies et al., 1995): accurate estimation of the robot location, fast and accurate detection of the robot environment and reliable path planning for moving from one place to another without colliding with obstacles in unknown environments. This is true for this proposal and adds the following:

- The total time it takes the robot to make the route and,
- Comparing the distance travelled by the robot with the straight line distance between the initial and goal locations.

When the robot moves at medium constant velocity (6.94 cm/s), it runs 107.72% of the straight line distance between the initial and goal locations. The travel distance is increased because of the obstacle avoidance and the location estimation errors

Table 2: Autonomous navigation results with and without velocity updating.

	Constant velocity		With velocity updating
	6.94 cm/s	13.88 cm/s	
Navigation velocity	6.94 cm/s	13.88 cm/s	8.65 cm/s (average velocity)
Navigation distance	228.5 cm	111.29 cm	220.33 cm
Navigation time	43.10 s	19.08 s	32.55 s

generated by the slippage of wheels. On the other hand, when the robot moves at maximum constant velocity (13.88 cm/s) runs, on average, 52.46% of the path. The plausible explanation is that when the robot moves at maximum velocity the wheels slip more often and therefore the location estimation of the robot becomes very imprecise. With velocity updating the robot travels the 103.86% of the straight line distance between the initial and goal locations. In this case the distance is less than with medium constant velocity because the wheel slippage is less frequent, see Table 3. By updating the velocity the robot moves slower in areas that favour the slippage of wheels, for instance loose soil. Because of there are fewer slippages, the location estimation of the robot is more accurate and consequently the robot approaches the goal location.

The navigation time with velocity updating is 32.41% less and 41.38% higher than medium and maximum constant velocity, respectively. It is noted that with medium constant velocity the robot travels a path with good accuracy but spends more time doing the travelling. With maximum constant velocity the travelling is fast but the accuracy to traverse the path is very bad. With velocity updating performance is improved because the precision of the robot in the travelling of the trajectory is good and is performed in less time, i.e., the robot moves at optimum velocity, depending on surface characteristics, avoiding wheel slippage. With our approach the average velocity represents 62.13% of the maximum velocity the robot can reach.

4 DISCUSSION

Within the present approach, the robot moves on surfaces with different kinds of textures, to make navigation more versatile than such related works. The Martian surface can be considered as a special case of our approach because these surfaces are

Table 3: Percentage of travelled distances, with maximum and medium constant velocity; and with velocity updating.

	Constant Velocity		With velocity updating
	Medium	Maximum	
Percentage of distance travelled	107.72%	52.46%	103.86%

covered with sand and rocks, i.e., there is only one type of roughness. Actually, for the purpose of autonomous navigation on rough terrains is not a requisite to recognize textures at a high detail level. The high precision methods on details recognition are not the adequate but failed for supporting robots navigation –strongly some times. In addition, the detailed recognition of surfaces is computationally expensive, but a low consume of resources is recommended through autonomous navigation.

Our approach can be improved on the location estimation of the robot. So far, it has been used odometry only to calculate the robot location. Most of the works, if not all, that employ odometry use other tools to estimate the location of the robot such as electronic compasses (Seraji and Werger, 2007), sonar sensors (Dai et al., 2007), GPS (Matthies et al., 1995), among others. However, velocity updating reduces wheel slippage and the drift errors are small or occur less often.

On the other hand, the proposed algorithms are not limited to be applied to small vehicles. They can be extrapolated to other vehicles, depending on the particular characteristics, which define the appropriate rules of the vehicle operation. The algorithm is scalable to different vehicles by using as parameters their particular characteristics, such as weight, size and motor power, tires material and tire tread, among others.

In this proposal we claim that for velocity updating the experience of human drivers is mimicking by using the inference system of the fuzzy neural network, which model the operation of the vehicle based on the driver experience.

There are works that model the vehicles driving with differential equations (Nakamura et al., 2007), (Kim et al., 2008), (Ward and Iagnemma, 2008). But this approach is difficult because, in general, differential equations are nonlinear, and their solution is hard to obtain.

Within the algorithms testing, we have simulated the path of a truck. These tests consisted of placing a camera on the roof of the truck. The truck runs on various types of textures. During the truck trips, the camera recorded from a similar driver’s visual field

the crossed surfaces. Then, texture images are extracted from video recordings, which are processed by the algorithms.

The velocity updating results are encouraging. To process the 480×640-pixel images with the microprocessor Centrino Core 2 Duo at 2 GHz and 1.99 Gb RAM, the algorithms time spent is small, 0.3 seconds. It leads to conclude that vehicles with these computer capacities have enough time to react or to break on the next 5 meters, as soon as they are moving at 60 km/hr, which is a car maximum velocity in the city, and a standard speed on principal roads.

5 CONCLUSIONS

In this paper a proposal for wheeled robot navigation on outdoor surfaces with different kind of roughness and soft irregularities is presented. The robot integrates the path planning gradient method with a multi-layer fuzzy neural network in order to adjust velocity, by regarding the roughness and the slopes of the terrain. The artificial vision implementation is computationally low-cost. Wheeled-robot navigation becomes more efficient and safe because of the velocity updating. That is because, whenever the robots navigates, the velocity is updated by regarding the terrains characteristics, the wheel slippage is significant reduced, hence improving, the precision to achieve the goal location as well as the navigation time; thereafter, the risk that the robot suffers an accident is also decreased. On the opposite, without velocity updating it becomes more difficult the goal location approach as reported results show.

ACKNOWLEDGEMENTS

The authors would like to thank the financial support of CINVESTAV-IPN, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional. As well Farid García, scholarship no. 207029, would like to thank the financial support of CONACyT, Consejo Nacional de Ciencia y Tecnología.

REFERENCES

- Brooks, C.A. and Iagnemma, K. (2009). Visual Detection of Novel Terrain via Two-Class Classification. In: *Proceedings of the 2009 ACM Symposium on Applied Computing*, 1145-1150.
- Dai, X., Zhang, H. and Shi, Y. (2007). Autonomous Navigation for Wheeled Mobile Robots-A Survey. In: *Second International Conference on Innovative Computing, Information and Control*, September 5-7, Kumamoto, Japan, 2207-2210.
- Ishigami, G., Nagatani, K. and Yoshida, K. (2007). Path Planning for Planetary Exploration Rovers and Its Evaluation Based on Wheel Slip Dynamics. In: *IEEE International Conference on Robotics and Automation*, April 10-14, Roma, Italy, 2361-2366.
- Kahraman, F. and Stegmann, M.B. (2006). Towards Illumination-Invariant Localization of Faces Using Active Appearance Models. In: *7th Nordic Signal Processing Symposium*, June 7-9, Rejkjavik, Iceland, 4.
- Kelly, A. and Stentz, A. (1998). Rough Terrain Autonomous Mobility – Part 2: An Active Vision, Predictive Control Approach. *Autonomous Robots*, 5(2), 163-198.
- Kim, P.G., Park, C.G., Jong, Y.H., Yun, J.H., Mo, E.J., Kim, C.S., Jie, M.S., Hwang, S.C. and Lee, K.W. (2007). Obstacle Avoidance of a Mobile Robot Using Vision System and Ultrasonic Sensor. In: *Third International Conference on Intelligent Computing*, August 21-24, Qingdao, China, 545-453.
- Kim, Y.C., Min, K.D., Yun, K.H., Byun, Y.S. and Mok, J.K. (2008). Steering Control for Lateral Guidance for an All Wheel Steered Vehicle. In: *International Conference on Control, Automation and Systems*. October 14-17, Seoul, Korea, 24-28.
- Konolige, K. (2000). A Gradient Method for Realtime Robot Control. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. October 31 – November 5, Takamatsu, Japan, 639-646.
- Lambert, A., Gruyer, D., Pierre, G.S. and Ndjeng, A.N. (2008). Collision Probability Assessment for Speed Control. In: *11th International IEEE Conference on Intelligent Transportation Systems*. October 12-15, Beijing, China, 1043-1048.
- Larson, A.C., Voyles, R.M. and Demir, G.K. (2005). Terrain Classification Using Weakly-Structured Vehicle/Terrain Interaction. *Autonomous Robots*, 19(1), 41-52.
- Nakamura, S., Faragalli, M., Mizukami, N., Nakatani, I., Kunii, Y. And Kubota, T. (2007). Wheeled Robot with Movable Center of Mass for Traversing over Rough Terrain. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. October 29 – November 2, 1228-1233.
- Matthies, L., Gat, E., Harrison, R., Wilcox, B., Volpe, R. and Litwin, T. (1995). Mars Microover Navigation: Performance Evaluation and Enhancement. *Autonomous Robots*, 2(4), 291-311.
- Pereira, G.A.S., Pimenta, L.C.A., Chaimowicz, L., Fonseca, A.F., de Almeida, D.S.C., Correa, L.Q., Mesquita, R.C. and Campos, F.M. (2009). Robot Navigation in Multi-Terrain Outdoor Environments.

- International Journal of Robotic Research*, 28(6), 685-700.
- ROBOTIS Co., (2010). <http://www.robotis.com>.
- Selekwa, M.F., Dunlap, D.D., Shi, D. and Collins, E.G. (2008). Robot Navigation in Very Cluttered Environments by Preference-Based Fuzzy Behaviors. *Robotics and Autonomous Systems*, 53(3), 231-246.
- Seraji, H. and Howard, A. (2002). Behavior-Based Robot Navigation on Challenging Terrain: A Fuzzy Logic Approach. *IEEE Transactions on Robotics and Automation*, 18(3), 308-321.
- Seraji, H. and Werger, B. (2007). Theory and Experiments in SmartNav Rover Navigation. *Autonomous Robots*, 22(2), 165-182.
- Sun, Z., Bebis, G. and Miller, R. (2006). On-Road Vehicle Detection: A Review. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 28(5), 694-711.
- Wang, M. and Liu, J.N.K. (2005). Behavior-Blind Goal-Oriented Robot Navigation by Fuzzy Logic. In: *Proceedings of Knowledge-Based Intelligent Information and Engineering Systems*, 686-692.
- Ward, C.C. and Iagnemma, K. (2008). A Dynamic-Model-Based Wheel Slip Detector for Mobile Robots on Outdoor Terrain. *IEEE Transactions on Robotics*, 24(4), 821-831.
- Ward, K. and Zelinsky, A. (2000). Acquiring Mobile Robot Behaviors by Learning Trajectory Velocities. *Autonomous Robots*, 9(2), 113-133.

VEHICLE ROUTING TO MINIMIZE MIXED-FLEET FUEL CONSUMPTION AND ENVIRONMENTAL IMPACT

O. Gusikhin¹, P. MacNeille¹ and A. Cohn²

¹Ford Motor Company Research and Innovation Center, 2101 Village Road, Dearborn, MI 48121, U.S.A.

²University of Michigan, 2797 IOE Building, 1205 Beal Avenue, Ann Arbor, MI 48019, U.S.A.

Keywords: Fuel Economy, Heterogeneous Vehicle Routing Problem, Optimization.

Abstract: Efficient vehicle routing is critical to the operational profitability and customer satisfaction of vehicle fleet-related businesses, especially in light of increasing, and highly volatile, fuel prices. Growing pressures to reduce negative environmental impacts have suggested that a second metric (vehicle emissions) should also be considered in vehicle routing. Currently, the majority of existing tools use distance as a surrogate for cost. When considering a mixed fleet of multiple vehicle types, with individual vehicles within a fleet type also varying by age and vehicle health, this surrogate becomes significantly less accurate. Furthermore, using distance as a surrogate fails to capture the variations between city and highway driving, which are particularly striking for hybrid vehicles. We thus propose a new approach to the vehicle routing problem, specifically targeting applications with mixed fleets including clean-vehicle technologies, in recognition of the limitations of the existing approaches.

1 INTRODUCTION

Efficient vehicle routing is critical to the operational profitability and customer satisfaction of vehicle fleet-related businesses. The rising and highly-variable cost of fuel, as highlighted by the price spike in the summer of 2008, increases the importance of efficient vehicle routing. At the same time, growing environmental concerns suggest that cost is not the only metric of importance and that emissions should also be taken into account.

Currently, the majority of existing methods and software packages minimize travel distance as a surrogate for cost. Although there is a positive correlation between distance traveled and fuel consumed, it is not a perfect correlation. In particular, both fuel efficiency and emissions vary depending upon driving conditions (e.g. city vs. highway driving). This variability is even more pronounced in a heterogeneous fleet comprised of multiple vehicle types with a range of fuel-consumption and emissions characteristics.

In recent years, socio-economical pressures to reduce their fuel costs and carbon dioxide (CO₂) footprint have motivated many fleet operators to begin upgrading their fleets, focusing on clean-vehicle technologies and alternative fuels, including

flex-fuel, hybrid vehicles, and plug-in hybrid electric vehicles (PHEV). As a result, many commercial fleets are currently composed of a heterogeneous set of vehicles, with noticeable variations in fuel economy and emissions across vehicle type. Furthermore, there are variations across vehicles even within a given vehicle type or common set of capabilities, because newer vehicles typically exhibit better fuel economy and better emission control technologies than older vehicles.

Consider the following examples of highly heterogeneous fleets: Florida Power and Lighting (www.fpl.com), the leader in green fleet initiatives, has a fleet of approximately 2400 vehicles, with half of the fleet powered by biodiesel, 300 hybrids and plug-in hybrids now in service, and plans to convert one-third of the vehicles to hybrid by the end of 2010.

A key issue in incorporating the fuel efficiencies and emissions of heterogeneous fleets within the vehicle routing problem is this: *The differences in fuel efficiencies and emissions across vehicle types are not exclusively proportional to the distance traveled, but are also highly dependant on the driving cycle.* For instance, a hybrid vehicle takes advantage of the regenerative braking that occurs in stop-and-go driving environments to charge a battery which can then be used to power the vehicle

when traveling at slow speeds. On average, fuel consumption is reduced about 20% by regenerative braking (Chan and Chau, 2001). Conversely, in highway driving, hybrids provide minimal, if any, improvement in fuel efficiencies over their conventional counterparts.

The introduction of PHEVs, which use electrically-charged batteries for the initial portion of a trip and revert to gasoline-based power only when the battery has been depleted, presents even greater variability in terms of the correlations between distance and fuel utilization.

When we move beyond fuel costs to also consider the environmental impact of vehicle routing, the complexity grows further. Fuel consumption is effectively proportional to CO₂ emission, so fuel economy improvements are reflected in CO₂ reduction. Additionally, there may be requirements to minimize or eliminate conventional emissions such as volatile organic components (VOC) and nitrogen oxides (NO_x) emissions in highly populated areas, requiring the use of electric power or other clean alternative fuel options and thus limiting the feasible region of a vehicle routing problem.

In this paper, we consider ways to explicitly capture fuel consumption and emissions in a mixed-fleet vehicle routing program and analyze the opportunities for simultaneously reducing costs and negative environmental impacts. In Section 2, we review factors that influence fuel consumption and emissions and consider ways to estimate these metrics for a given route. In Section 3, we suggest a number of formulations for this new variation of the vehicle routing problem. We also outline a solution approach based on *composite variable modeling*. In Section 4, we provide a numerical example and analysis to highlight the benefits of our proposed approach and we then offer conclusions and suggested areas for future research in Section 5.

2 ESTIMATION OF FUEL ECONOMY & VEHICLE ENVIRONMENTAL IMPACT

Route planning is done by representing the road system as a graph in which intersections are nodes and road segments are arcs. To determine the best route from an origin node to a destination node, each arc is associated with a cost that represents distance, travel time, or fuel consumption. Then Dijkstra's algorithm (or an equivalent) is used to find the

lowest cost path which is then inversed mapped to the road system for visualization and navigation of the preferred route.

Total fuel cost along a given route is the product of the total consumption of each type of fuel and the per-unit cost for that fuel. Total fuel consumption along an arc is largely dependant on the vehicle specific load (VSL) opposing vehicle motion multiplied by the distance traveled against the VSL to give the total energy required to travel the arc. This energy can be readily converted into fuel. For example, about 0.003 gallons of gasoline or 0.002 kWh of electricity are needed to push against a pound of force over a one mile stretch of road. These factors will vary, however; with the efficiency of the energy conversion which depends on many factors.

The VSL depends largely on external factors that include the drive cycle, aerodynamic drag, rolling resistance, parasitic drag, and gradient drag. These factors in turn are dependant on several external factors including weather conditions, road and traffic conditions and topography. Most of the external factors can be reasonably well estimated using well known engineering formulas; however, the VSL also depends on the pattern of acceleration/deceleration that takes place along the branch. Driving pattern effects depend largely on complex interactions between the driver, the road, the powertrain, and traffic conditions. Thus, these effects are difficult to predict.

One approach to predicting these complex factors is by classifying road, traffic, and driver and drivetrain combinations into load effects. The US Environment Protection Agency (EPA) provides miles per gallon estimates for highway and city for all vehicles sold in the US in the last 15 years currently based on two standard driving cycles. Although these estimates may not be adequate to accurately forecast the specific fuel consumption, they nevertheless can provide a reasonable basis for the comparative analysis between different vehicles.

A more detailed classification is described in Brundell-Freij and Ericsson (Brundell-Freij and Ericsson, 2005), where a classification system has been developed based on extensive data collected from instrumented vehicles. Four variables relating to the road type were found to be significant: 1) occurrence and density of junctions controlled by traffic lights, 2) speed limit, 3) function of the street, and 4) the type of neighborhood. A large effect was attributed to the power-weight ratio of the vehicle, which presumably is descriptive of the drivers that choose a vehicle with a specific power-weight ratio.

A challenge in using instrumented vehicles is in the ability to collect adequate data. For example, in the Brundell-Freij and Ericsson study much of the data was collected in Lund, Sweden. This location has limited topography, so road gradient was not captured, although it is well known that road gradient is an important factor in situations with topographic relief. For example, (Tavares et al., 2009) develops a topography-based routing algorithm for waste collection vehicles in a mountainous area and demonstrates that the proposed approach allows reduction of fuel consumption despite increasing in the distance traveled. Also road design and traffic control policies may vary considerably between political jurisdictions, as may the mix of vehicles. One way to overcome these difficulties is to use vehicle and traffic modeling to determine the significant factors for classification.

Modeling tools such as the Powertrain System Analysis Toolkit (PSAT) (PSAT, 2008) or The MathWorks Simulink/SimDriveLine (Rose-Hulman, 2005) can be used to estimate the energy or fuel along a route given vehicle design parameters, external load factors such as road gradient and the driver's torque demand along the route. Vehicle design parameters can be obtained from vehicle manufacturers and external factors from published maps. Driver's torque demand involves the psychophysics of driving and has been simulated using software such as VISSIM or MISSION (PTV AG, 2009), (Wiedemann et al., 1991), (Busawon et al., 2006), (Noland and Quddus, 2006).

Vehicles in the fleet may be instrumented to collect actual fuel economy data along the branches they travel. The data may be recovered from the vehicle at a download site and stored in a database. Periodically the database may be used to automatically refine the costs assigned to a class of road segments, and to reclassify segments as needed.

3 VEHICLE ROUTING TO MINIMIZE FUEL CONSUMPTION

The Vehicle Routing Problem to Minimize Mixed-Fleet Fuel Consumption and Environmental Impact (VRPMF) belongs to the class of heterogeneous fleet vehicle routing problems (HVRP). (Baldacci et al., 2008) provides a comprehensive classification and review of the main approaches proposed for VRP with a heterogeneous fleet. Specifically, the problem

being considered in the paper represents a variant of HVRP with Vehicle-Dependent Routing Costs (HVRPD). They note that solution approaches to this difficult family of problems, both in the literature and in commercial applications, have predominantly been heuristic in nature. These are typically adaptations or extensions of solution techniques for traditional VRP and VRP with Time Windows.

In order to capture the complexities (and, in particular, the non-linearities) of VRPMF, we instead propose to leverage the use of *composite variable modeling* (CVM) to capture the complex real-world details associated with accurately modeling the fuel cost (and associated emissions) of a prescribed route. The idea behind CVM (Cohn, 2002), (Barlatt, et al., 2009) is to embed modeling complexity into the variable definition rather than capturing it explicitly in a model which may then become intractable. For example, in VRPMF, explicitly modeling the cost functions described in Section 2 within the framework of a traditional VRP would make an already difficult problem unsolvable. However, it is far easier to calculate the cost of a given route (for a given vehicle) off-line. We can then formulate a master problem in which each variable represents the assignment of a specific route to a specific vehicle. For a given route, we can compute the total fuel consumption for a given vehicle, and thus the total cost is just the sum of the chosen assignments. Similarly, a route pre-specifies all the customer demands that it meets, and thus we only need two sets of constraints. The first ensures that each vehicle is assigned to at most route and the second ensures that each customer demand is met exactly once.

The challenge, then, is to address the exponentially large number of potential variables. Clearly not all of the exponentially-large set of feasible routes (and their corresponding costs) can be generated. Even if they could, it would not be possible to solve the resulting exponentially-large set partitioning problem. Instead, *column generation* techniques (Desaulniers et al., 2005) (originally developed as part of Dantzig-Wolfe Decomposition) can be employed. The idea behind column generation for solving a linear program with an exponential number of variables is to identify candidate pivot variables for the simplex method not by pricing each variable's reduced cost directly, but rather by solving a secondary optimization problem (often called a *sub problem*) which seeks the feasible variable with the most negative reduced cost. If this yields a negative reduced cost variable, then the

simplex pivot occurs and the algorithm proceeds. If the most negative reduced cost variable is strictly non-negative, then a certificate of optimality is achieved and the algorithm terminates with a provably-optimal solution.

In Dantzig-Wolfe Decomposition, the inherent structure of a problem leads to a sub-problem that is pre-defined. Furthermore, it is itself a linear program and thus straightforward to solve. In CVM, because the variable definition is chosen specifically to overcome challenges of a traditional formulation, the sub problem reflects these challenges.

Perhaps the most closely related work to our proposed approach is that of Taillard (Taillard, 2005), who used a heuristic based on column generation techniques to solve HVRPD. Specifically, a large set of candidate routes were generated by solving separate homogeneous VRP problems for each fleet type. The final routes were then selected using a set partitioning formulation to ensure that all demands were met.

A key difference between our proposed approach and typical network-based routing problems is that the “cost” of a route cannot be computed simply by adding the individual arc costs (if so, the sub-problem would be a simple minimum cost flow problem). At first glance, it seems possible to formulate the sub problem as a network flow problem, where each node represents a customer or depot, each arc represents the driving from one node to another, and the cost associated with each arc can fully capture (based on off-line calculations) the cost of this driving. This is not quite true, however: The cost on a given arc is not independent of the other arcs that are also chosen for an individual vehicle’s route. This is because the fuel consumption on an arc depends on the starting conditions of the vehicle at the first node. If the battery is fully charged, it may be possible to complete most of the driving without relying on gasoline, and the resulting cost will be lower, whereas if the battery is depleted, the cost of the arc will be much higher.

Therefore, a more sophisticated approach to solving the sub-problem must be employed. For example, we could take a *multi-label shortest path* approach (Desrochers and Soumis, 1988), which is similar to Dijkstra’s shortest path algorithm, but with an added layer of complexity. Specifically, multiple metrics (not just cost) must be checked to determine whether a partial path can be pruned from consideration. One partial path dominates another only if it is less costly *and* covers the same amount of demand or more *and* has the same amount of remaining battery charge or more. Efficiently

solving this sub-problem is the key to successfully solving the master problem.

We conclude this section by noting that this approach has the added advantage of allowing the user to trade off between time and solution quality. Specifically, the solution quality continues to improve as each new candidate route is added to the master problem for consideration, but high-quality feasible solutions can nonetheless often be found early in the process. Furthermore, this approach naturally lends itself to a parallel implementation. At the highest level, a separate sub-problem can be solved, in parallel, for each vehicle. Furthermore, these individual sub-problems themselves can leverage a parallel architecture for efficient search.

4 ILLUSTRATIVE EXAMPLE

This section provides a simplified illustrative example of VRPMF. We consider a fleet of two vehicles: the first is a 2009 Ford Taurus front-wheel drive gasoline engine vehicle and the second is 2010 Ford Escape 4-wheel drive hybrid vehicle. Figure 1 shows estimates of the miles per gallon (MPG) values and environmental scores as provided by the US Environmental Protection Agency (EPA) at www.fueleconomy.gov. Note that the Taurus gets 18 MPG in city driving and 28 MPG in highway driving, while the Escape hybrid gets 27 MPG on the highway and 30 MPG in the city. The environmental impact of each vehicle can be evaluated by its carbon footprint and air pollution score. The carbon footprint measures greenhouse gas emissions (primarily CO₂) that in turn impact climate change. CO₂ emissions are closely linked to fuel consumption, since CO₂ is the ultimate end product of burning gasoline. The Air Pollution score represents the amount of health-damaging and smog-forming airborne pollutants (such as carbon monoxide, CO, and oxides of nitrogen, NO_x) that the vehicle emits on a scale from 0 (worst) to 10 (best). Note that there is little correlation between fuel consumption and these emissions; emissions primarily depend on the emission control technology. Taurus has an Air Pollution score of 6 and Escape Hybrid has a score of 8.

Suppose that we have eight customers distributed within a given geographical area as shown in Figure 2. The customers are labeled 1 to 8, while 0 is the depot. The driving distance between the depot and each of the customer sites is presented in Table 1. For this example, we assume that the distance data is symmetrical and shown as $X+Y$, where X is the

Table 1: Distance between depot and between customer's sites.

	0	1	2	3	4	5	6	7	8
0		14.5+ .9	19+ 1	23.9+ 24	0+ 11.6	0+ 9.8	0+ 13.6	12.5+ 10	15.5+ 1.1
1	14.5+ .9		4.2+ 1.5	10.2+ 1.8	0+ 6.3	0+ 10.3	0+ 12.1	12.2+ 2.6	0+ 17.9
2	19+ 1	4.2+ 1.5		5.7+ 2.2	25+ 9.1	4.2+ 11.6	11.4+ 4.7	7.7+ 2.9	20.2+ 2
3	23.9+ 24	10.2+ 1.8	5.7+ 2.2		0+ 6.7	0+ 10.6	0+ 6.5	0+ 3.8	14.3+ 1.5
4	0+ 11.6	0+ 6.3	2.5+ 9.1	0+ 6.7		0+ 4	14+ 6	17+ 7.5	0+ 11.6
5	0+ 9.8	0+ 10.3	4.2+ 11.6	0+ 10.6	0+ 4		0+ 3.8	0+ 8.6	0+ 7.7
6	0+ 13.6	0+ 12.1	11.4+ 4.7	0+ 6.5	14+ 6	0+ 3.8		0+ 4.7	7.4+ 4.1
7	12.5+ 10	12.2+ 2.6	7.7+ 2.9	0+ 3.8	17+ 7.5	0+ 8.6	0+ 4.7		12.5+ 2.5
8	15.5+ 1.1	0+ 17.9	20.2+ 2	14.3+ 1.5	0+ 11.6	0+ 7.7	7.4+ 4.1	12.5+ 2.5	



	
2009 Ford Taurus FWD	2010 Ford Escape Hybrid 4WD
EPA MPG City	
18	30
EPA MPG Hwy	
28	27
EPA Air Pollution Score	
6	8

Figure 1: Fleet Composition.

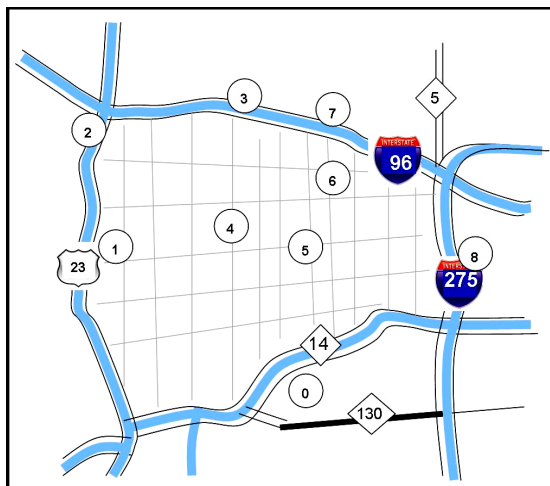


Figure 2: Vehicle Routing Problem.

number of highway miles and Y is the number of city miles.

Further, we assume that each service call requires approximately 90 minutes and that each service agent has to finish his/her route, starting and ending at the depot, within 8 hours.

We ignore breaks, and assume that there is no capacity limit on the routes other than the time limit. We begin by solving the traditional problem of minimizing the total fleet vehicle miles traveled (VMT). (In this small problem, this can be done by explicit enumeration). The total optimum VMT is 97.2. One vehicle visits customers 1, 2, 3, and 4 with a total travel distance of 47.3 miles, comprised of 22.9 city miles and 24.4 highway miles. The other vehicle visits customers 5, 6, 7, and 8 with a total travel distance of 49.9 miles, comprised of 21.9 city miles and 28 highway miles. The estimated travel time for the first route is 75 minutes and for the second route is 78 minutes. Note that in this variation of the problem, we do not differentiate between vehicles, as we are simply minimizing distance traveled.

For illustrative purposes, to estimate fuel consumption of each vehicle along the given routes we assume the estimated highway and city MPGs as defined in Figure 1. (Of course, real-world calculations are more complex as we discussed in section 2). As a result, the fuel consumption of the Taurus for the first route is 2.14 gallons and for the second route is 2.22 gallons. For the Escape Hybrid, the first route consumes 1.67 gallons and the second route consumes 1.77 gallons.

The optimal solution is to assign the Taurus to the first route, resulting in the consumption of 2.14 gallons, and to assign the Escape to the second route, with an estimated consumption of 1.67 gallons of gasoline. The total fleet fuel consumption for the given solution is 3.81 gallons of gasoline. The results are presented graphically in Figures 3 and 4.

We next reformulate the problem to take into account fuel economy, i.e. distinguishing in our optimization between highway and city driving, and treating the two vehicles separately, in recognition of their distinct characteristics. The optimal solution for this problem is presented in Figures 5 and 6.

Although the VMT increases (from 97.2 to 101.7), the fuel consumption decreases from 3.81 to 3.72 gallons. Note that the optimal routes have changed (see Figure 4). The route served by the Taurus covers customers 1, 2, 7, and 8 with a total length of 63.3 miles, but of this 54.4 miles is highway driving and only 8.9 miles is city driving. Conversely, the Escape Hybrid is assigned to a route that serves customers 5, 6, 3, and 4, using only city driving (for a total length of 38.4 miles).

Table 2 provides a comparative summary of the two solutions. In addition to the economical impact, this demonstrates how a reduction in fuel consumption also leads to a reduction in environmental impact from the fleet operations. First of all, the reduction of fuel consumption reduces the CO₂ emissions. In addition, the second solution shifts the time the Taurus spends in the city routes to the Escape Hybrid. As is shown in Figure 1, the Taurus has a lower EPA Air Pollution Score than the Escape Hybrid. Consequently the second solution also reduces health-damaging and smog-forming airborne pollutants along populated areas. We could also capture this explicitly in the objective function of our formulation, either by specifying constraints on the total Air Pollution Score (possibly weighted by the location of the route arcs), or by introducing weights in the objective function.

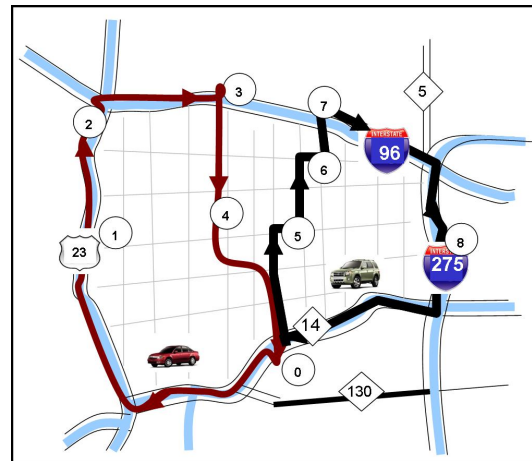


Figure 4: Solution 1 Map.

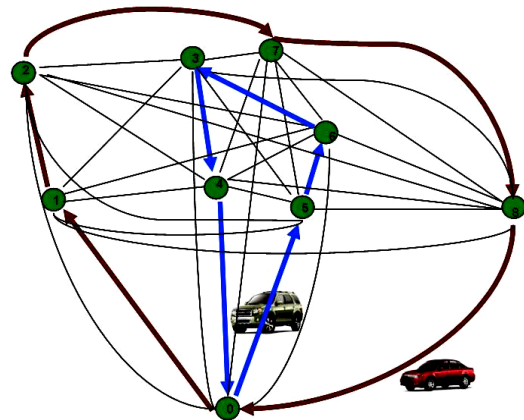


Figure 5: Solution 2 Graph.

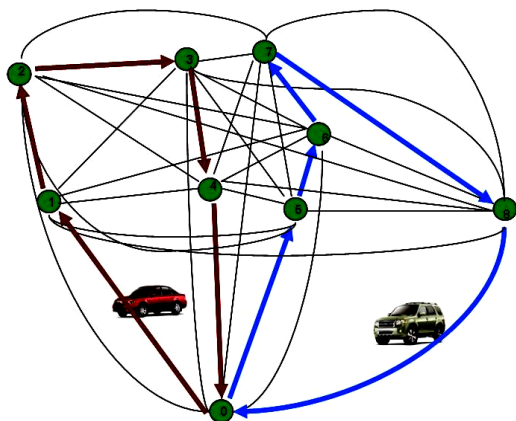


Figure 3: Solution 1 Graph.

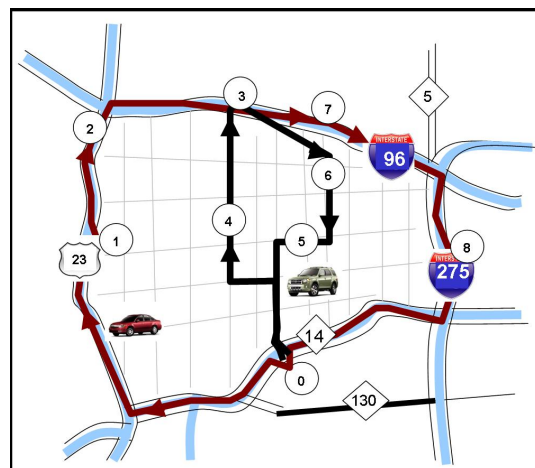






Figure 6: Solution 2 Map.

Table 2: Comparative summary of two VRT solutions.

Minimum Distance Optimization									
	Distance (miles)			Fuel (gallons)			Time (minutes)		
	Hwy	City	Total	Hwy	City	Total	Hwy	City	Total
	24.4	22.9	47.3	0.9	0.8	1.7	21	54	75
	28.0	21.9	49.9	1.0	1.2	2.2	25	53	78
Summary	52.4	44.8	97.2	1.9	2.0	3.9	46	107	153
Minimum Fuel Optimization									
	Distance (miles)			Fuel (gallons)			Time (minutes)		
	Hwy	City	Total	Hwy	City	Total	Hwy	City	Total
	54.4	8.9	63.3	1.9	0.5	2.4	48	26	74
		38.4	38.4		1.3	1.3		84	84
Summary	54.4	47.3	101.7	1.9	1.8	3.7	48	110	158

5 CONCLUSIONS

In this paper we discuss the importance of the heterogeneous fleet vehicle routing problem based on fuel consumption rather than just distance traveled. We describe the complex function that determines how much fuel a given route consumes, and argue that distance is an inadequate surrogate when multiple fleet types, especially varying across different technologies, are used. We provide a simple example to illustrate how minimizing total miles traveled can yield a very different solution than minimizing fuel consumption. We also discuss solution techniques, specifically based on the use of composite variable modeling, to solve this computationally challenging problem.

In the future, we propose to consider emissions as well as explicit fuel costs (which implicitly capture CO₂ emissions but not NO_x). We also suggest extending VRPMF to include variations and extensions such as those studied in the basic VRP, such as balancing routes, satisfying time windows, etc.

REFERENCES

Chan, C. C. and Chau, K.T.; Modern Electric Vehicle Technology. Oxford University Press Nov. 15, 2001.
 Brundell-Freij, K. and Ericsson, E.; Influence of street characteristics, driver category and car performance on urban driving patterns, Transportation Research Part D 10, 2005, pp. 213-229.
 Tavares, G., Zsigraiova, Z., Semiao, V., Carvalho, M.G , Optimization of MSW collection routes for minimum fuel consumption using 3D GIS modeling, Waste Management 29, 2009, pp. 1176-1185.
 PSAT Overview, January 2008, <http://www.transportation.anl.gov/pdfs/HV/412.pdf>
 Rose-Hulman Institute of Technology; Rose-Hulman Institute of Technology Students Design Hybrid

Vehicle Powertrain with Simulink and SimDriveLine, 2005; http://www.rose-hulman.edu/challengex/downloads/Rose-Hulman_user_story.pdf
 PTV AG, VISSIM – State-of-the-Art Multi-Modal Simulation; http://www.ptvag.com/fileadmin/files_ptvag.com/download/traffic/Broschures_Flyer/VISSIM/VISSIM_Brochure_e_2009_HiRes.pdf, 2009.
 R. Wiedemann, U. Reiter, Wiedemann Model-Microscopic Traffic Simulation The Simulation System Mission, 1991; http://www.ptvamerica.com/fileadmin/files_ptvamerica.com/library/1970s%20Wiedemann%20VISSIM%20car%20following.pdf
 Roshan Busawon, M. David Checkel, CALMOB6: A Fuel Economy and Emissions Tool for Transportation Planners, presented to the Best Practices in Urban Transportation Planning session of the 2006 Annual Conference of the Transportation Association of Canada, Charlettetown, Prince Edward Island, 2006.
 Robert B. Noland, Mohammed A. Quddus, Flow improvements and vehicle emissions: Effects of trip generation and emission control technology, Transportation Research Part D 11, 2006, pp. 1-14.
 Roberto Baldacci, Maria Battarra, and Daniele Vigo, Routing a Heterogeneous Fleet of Vehicles in B. Golden, et al. (Eds), The Vehicle Routing Problem: Latest Advances and New Challenges, Springer 2008, pp. 3-27.
 Cohn, A. “Composite Variable Modeling for Large-Scale Problems in Transportation and Logistics,” doctoral dissertation in Operations Research, Massachusetts Institute of Technology, advised by Dr. Cynthia Barnhart, April 2002.
 Barlatt A.,Cohn A.,FradkinY., Gusikhin O.; Morford C., Using composite variable modeling to achieve realism and tractability in production planning: An example from automotive stamping, IIE Transactions, Volume 41, Issue 5, 2009 , pp. 421 – 436.
 Desaulniers, G., J. Desrosiers, and M. Solomon, editors. “A Primer in Column Generation.” Springer US, 2005.
 Taillard E. D., A heuristic column generation method for the heterogeneous VRP, Operations Research -- Recherche opérationnelle 33 (1), 1999, pp. 1-14. (Publication CRT-96-03, Centre de recherche sur les transports, Université de Montréal, 1996. Revision 11.2005)
 Desrochers, M. and F. Soumis. A Generalization Generalized Permanent Labeling Algorithm for the Shortest Path Problem with Time Windows. INFOR 1988 26:191 – 212.

POSTERS

MODEL-BASED FEATURE EXTRACTION FOR ASSESSMENT OF DRIVER-RELATED FATIGUE

Damián Alvarez, Álvaro Orozco

Grupo de Investigación en Control e Instrumentación, Universidad Tecnológica de Pereira, La Julita, Pereira, Colombia
damianalvarez@ohm.utp.edu.co, aaog@utp.edu.co

Augusto Salazar

Grupo de Investigación en Percepción y Control Inteligente, Universidad Nacional Sede Manizales
La Nubia, Manizales, Colombia
aesalazarj@unal.edu.co

Keywords: Driver fatigue.

Abstract: One of the most major causes of road crashes is the fatigue. In this paper it is shown a methodology for Driver Fatigue assessment based on computer vision (CV). CV is used to characterize different visual responses of the driver while driving and suffering from fatigue. Some of the visual responses are the eyelid and lips movements. The proposed Methodology uses an active appearance model (AAM) to adjust the facial model *Candide3* from images sequences where spatial measures can be computed. These measures include the eye closeness and the mouth openness. Results show that with the measures computed it's possible efficiently extract some discriminant parameters related to driver fatigue state. For example, the PERCLOS, the AECS, and the YawnFrec. Finally, an experimental framework is designed in order to compare the performance of the proposed method with psychological signal-based methods.

1 INTRODUCTION

Driving for long periods of time can decrease alert and performance of the driver who could start to suffer from fatigue (Ting et al., 2008). Fatigue and lack of sleep have been identified as the most frequent causes of traffic accidents in the road. For instance, the driver who is under these conditions is risking not only his/her own life but also other people lives. In order to prevent these kind of accidents, in the last decades a huge effort has been made to develop of monitoring systems that detects the fatigue in the driver and warned him/her by using different techniques. However, an efficiently system for fatigue assessment is still being an important issue to solve.

2 THEORETICAL BACKGROUND

Fatigue. Is the state of alteration in both the awareness level and the perception level of the person, this state affects psychomotor processes, such as speed of reaction, attention level, and making decisions that are crucial for the safe development of an activity. Fatigue is not the same as sleep, but induction of sleep

could occur with fatigue. Fatigue can be caused by physical effort, emotional stress, lack of sleep, or an unspecified disorder.

Driver Fatigue. Is a state of reduced mental alertness, which impairs performance of a range of cognitive and psychomotor tasks, including driving (Saroj and Lal, 2001). The driver fatigue can be sub-divided into two groups: related to sleep and related to tasks. The first one, is caused by lack of sleep (this is the category select in this work), while the second one is caused by distracting tasks while driving.

Candide Model. It is a parameterized facial mask, that has been specifically developed for model-based coding of human faces. Different versions of this mask has been developed, the most current version is the third, which has been implemented mainly to simplify the animation of MPEG-4 facial animation parameters (Ahlberg, 2001).

2.1 Psychological Test for Attention Measurement

Works that deal with the assessment of driver-related fatigue require experimental frameworks where the methodology can be contrasted and validated. For ex-

ample, attention test (such as interviews or computer based test), which are basically monitoring task that required psychomotor reactions. One test of attention most commonly used in drivers is the test of attention variables TOVA. It's considered the gold standard within this kind of tests. It consists in 22-minutes of psychomotor tasks where a subject needs to hold the attention and respond to a random stimulus. Another example, is the PVT test (psychomotor vigilance test), that measures the visual reaction time (RT) by mean of portable devices. This test is another common tool of measuring the fatigue in performance and lack of sleep studies. The PVT is a 10 minutes length test, like the TOVA test requires responding to a stimulus as fast as possible.

Others examples of tests that can be used in these kind of studies are those tests that aim to evaluate the ability to focus the attention. One example of such test is the Stroop task. In general any test that measures the RT is useful in these studies, because the sleep restriction and deprivation can be the main cause of an increase in the RT.

2.2 Classification of Driver Monitoring Systems

In order to reduce road crashes, enormous efforts have been done to develop driver monitoring systems. These systems can be classify in three classes (Vural et al., 2007). 1) Studies that analyze measures of the driver's performance. In this kind of systems the car is equipped with measurement devices that indicate if the person is driving as usual or not. These systems has the drawback that cannot be adapted to the driver habits. 2) Work that are related to the measurement of physiological signals. These methods provide good indicators of fatigue. However, these are invasive methods that could interfere with the driving task. 3) Works focus on detection of visual responses that present the driver based on CV. The CV based methods can be successful in order to assess fatigue states. However, so far just one visual response has been used and that could be the main drawback in the CV based methods (Zhu et al., 2004).

Measurement of fatigue indicators is a significant problem due to the absence of direct measures, which are not directly relate to the fatigue but to the effects of fatigue. The only direct measure is the self report. However there are problems related with its use because the emotional influence in the person (Wang et al., 2006). In the literature there are different studies related with the measurement of performance, physiological, of perception, among others. Although the technologies applied to fatigue assessment have

evolve, the search for a fiable fatigue indicator still ahead (Saroj and Lal, 2001).

2.3 Visual Responses and Fatigue-Related Parameters

In order to detect driver fatigue it is required to measure different visual responses. Simultaneous measures will provide a less ambiguous scenary that just one measure. Some of the visual responses are: the eyelids, the head pose, and the facial expressions. From these visual responses fatigue-related parameters are computed. For example, head-position-dependent parameters and head-position-independent parameters. the last will be used in this work, more specifically are computed: the percentage of eye closing in time (PERCLOS), the eye closing average speed (AECS), and the yawning frequency in time (YawnFrec) (Zhu et al., 2004).

PERCLOS and AECS. These measures are characteristic of the eyelid movement. The PERCLOS has been already validated and it has been found that is the most appropriated parameters to assess the driver-related fatigue (Dinges et al., 1998). The AECS is a good indicator of fatigue. This parameter is defined as the time taken to completely close the eyes. The eye openness is characterized by pupil's shape. It can be measure by taking the ellipse axis relationship. This feature over the time is used to computed the PERCLOS (Zhu et al., 2004). In (Ji and Yang, 2002), it has been shown that the AECS in a tired person is definitely different from a rested person.

YawnFrec. A tired person is characterized by expressing less facial expressions because there is minimal activity of facial muscles, but it also show more open mouth. The mouth openness can be measure by detecting the lips movements whether the features around it deviate from its closed configuration. The mouth openness is characterized by the proportion between the height and width to computed the YawnFrec (Zhu et al., 2004).

3 FATIGUE ASSESSMENT BASED ON COMPUTER VISION

People in a state of fatigue show some visual responses that are easily observable from changes in their facial features like eyes and mouth. CV has different non-invasive techniques for monitoring of drivers (Wang et al., 2006). As Zhang (Zhang and Zhang, 2006) reported, the CV-based systems for monitoring drivers are the most promising commer-

cial application for assessment of driver-related fatigue (Zhang and Zhang, 2006), (Dong and Wu, 2005), (Saeed et al., 2007), (Zhu et al., 2004), (Ji and Yang, 2002). However, CV is a challenging research area due to the different factors such as: facial expressions complexity, fast eye and head movements, and changes in illumination conditions, etc. These factors difficult the development of robust and real time implementations of these kind of systems.

Basic Requirements for A CV-Based System. According to (Dong and Wu, 2005), and (Smith et al., 2003) a monitoring system to detect fatigue should fulfill the following requirements: 1) fully automatic. 2) to be based on only quantitative features, such as, the eye openness and closeness rate. 3) to work under changing illumination conditions. 4) to work under occlusion conditions that are frequent when head is moved from the reference point. 5) real-time. 6) non-invasive, without physical contact of the driver. 7) before an accident it should detect and warn the early occurrence of sleep.

Steps in A Visual System for Detection of Fatigue.

First, near infrared image sequences were acquired. Second, mathematical morphology is used to improve the images quality and the success rate in the following steps. Third, the Haar algorithm is used to detect the face and then the Gabor algorithm is used to detect the person's eyes. Then the distance between the eyes is used reference to adjust the Candide model to the person's face. Fourth the AAM technique is used to track the movements of the face. Finally the feature extraction stage is performed using a model.

4 CHARACTERIZATION OF FACE MOVEMENTS

The most important points to characterize a face depend on the application and the facial model. In this work the movements of the eyes and mouth are considered. In these sense, a whole face model is used instead of a partial model of the eyes, as used in (Saeed et al., 2007).

In order to get a quite realistic fitting of model and detect the driver fatigue it's need to define the adjustable vertices of the model Candide. These vertices belong to the eyes and mouth ROIs where the parameters PERCLOS, AECS, and YawnFrec are computed. Additionally it is desirable that the movements of selected vertices are controlled by facial action units, these movements are part of the list of changes that are presented when there is a movement of the ROI encoded in terms of AUVs (Ahlberg, 2001).

Facial Motion Tracking. Previous to the computation of the parameters of fatigue the AAM models are used to adjust the face model to different subject faces. The AAM models are adjusted by machine learning algorithms from available features (Cootes et al., 1995). Then an alignment process is run to adjust the AAM model (vertices update) to the input images. To train the AMM an graphical user interface (GUI) was developed, were the following steps are done. 1) Scaling of the model to the face based on the inter-eye distance, 2) Rotations in axes X, Y, and Z to locate the model in the subject position, 3) Shape changes of the model regions are evaluated (control by shaped units) or movements in regions of the model (with vector control units of action, focusing on the mouth openness AUV11 and eye closeness AUV6) are evaluated too. 4) Points in the boundary of the face and eyes and mouth are reviewed and re-located. 5) The positions of the vertices of the fitted model are stored for each image in the sequence. Once the training algorithm is completed, a fitting algorithm is used to adjust the AAM to minimize the fitting error. For example, the decreasing gradient optimization can be used as described in Eq. 1, with the model A_0 and the input image $I(x)$. Once the model is adjusted the extraction of fatigue features can start.

$$FittingError = \sum^x [I(x) - A_0(x)]^2 \quad (1)$$

Eyelid and Lips Movements Characterization. To compute the PERCLOS and AECS in (Ji and Yang, 2002) is proposed to continually track the pupil and measure the eye closeness cumulatively over time, using the ratio of the vertices of the pupil ellipse. A single eye closure is defined as the difference between two periods of time in which the pupil size is 20% or less of the normal size. The closing eye speed is defined as the time period in which the pupil size is between 80% and 20% of the nominal size of the pupil.

In order to computed these two parameters more accurately, an average time is used of all measurements taken on a defined range (Ji and Yang, 2002).

To make these measures it is apply the methodology described in (Ji and Yang, 2002), but different from this, eye closeness is computed using the eye vertices which are defined in the model Candide 3. Specifically, we used some vertices of the eyelids. Thus, the eye closeness is calculated as:

$$c_{re} = \frac{d(v_{98} - v_{100}) + d(v_{54} - v_{55}) + d(v_{106} - v_{108})}{3} \quad (2)$$

$$c_{le} = \frac{d(v_{105} - v_{107}) + d(v_{21} - v_{22}) + d(v_{97} - v_{99})}{3} \quad (3)$$

From Eq. 2 and 3, when the eye closeness is less than or equal to 20% of the maximum distance between eyelids is considered that the eyes are closed. According to the work developed in (Dong and Wu, 2005) if the eyes are closed for 5 consecutive frames it can be considered that the driver is falling asleep.

In order to compute the rate of mouth opening, it is necessary to know the degree of the mouth openness, which is represented by the relation between the mouth's height and width. The graphical representation of the mouth openness in a period of time is known as YawnFrec. To compute the mouth openness the vertices (right, left, upper, lower), of the mouth in the model Candide 3 are used. In this sense, the mouth openness is computed as follows:

$$OpenMouth = \frac{d(v_7 - v_8)}{d(v_{64} - v_{31})} \quad (4)$$

5 EXPERIMENTAL FRAMEWORK

Two different videos were acquired with 320x340 resolution. The subjects were instructed to blink and yawn in different head positions range from 45°y – 45°. The subject 1 blinked 35 times and yawned 5 times. The subject 2 blinked 24 and yawned 4 times. The system was able to identify the total numbers of yawns and blinks for each subject. This means a 100% accuracy in detection of eye closeness and mouth openness. Indeed the parameters PERCLOS, AECS, and Yawnfrec could be determined every time. In order to validate the presented methodology, an experimental framework is proposed. First, a commercial driving simulator is going to be used and three different physiological tests will be set up (the PVT test, the Stroop test, and the RT test). These tests are embedded in the software PEBL (Open source Psychology Software), available at "http://pebl.sf.net". From these tests some basic measures will be made, in similar ambient and physical conditions to those presented in the computer-vision-based methodology in order to compare results and validate the parameters.

6 CONCLUSIONS

In this paper was proposed a characterization methodology based on models to assess fatigue. Two variables are measured by means of computer vision. The eye closing range and the mouth opening range. These measures are the base to calculate the parameters PERCLOS, AECS y YawnFrec. It has been found that the proposed methodology is practical and

reliable within the previously described conditions. Also an experimental framework based on psychological measures was defined in order to validate the proposed methodology. As future work, it is proposed to complement the above methodology with estimation of head's position in order to compute additional parameters. This will provide more information to the assessment of the driver fatigue.

REFERENCES

- Ahlberg, J. (2001.). Candide-3 an updated parameterized face. Report No. LiTH-ISY-R-2326, Dept. of Electrical Engineering, Linköping University, Sweden.
- Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J. (1995). Active shape models-their training and application. *Comput. Vis. Image Underst.*, 61(1):38–59.
- Dinges, D. F., Mallis, M., Maislin, G., and Powell, J. W. (1998). Evaluation of techniques for ocular measurement as an index of fatigue and the basis for alertness management.
- Dong, W. and Wu, X. (2005). Driver fatigue detection based on the distance of eyelid. *IEEE Int. Workshop VLSI Design and Video Tech Suzhou*, pp 365-368.
- Ji, Q. and Yang, X. (2002). Real-time eye, gaze, and face pose tracking for monitoring driver vigilance. *Real-Time Imaging*, 8(5):357–377.
- Saeed, I., Wang, A., Senaratne, R., and Halgamuge, S. (2007). Using the active appearance model to detect driver fatigue. In *Information and Automation for Sustainability, 2007. ICIAFS 2007. Third International Conference on*, pages 124–128.
- Saroj, K. and Lal, A. (2001.). A critical review of the psychophysiology of driver fatigue. *Biological Psychology* 55 173194.
- Smith, P., Shah, M., and Vitoria, N. (2003.). Determining driver visual attention with one camera. *Trans. on Intelligent Transportation Systems*, 4(4):205–218.
- Ting, P., Hwang, J., Doong, J., and Jeng, M. (2008.). Driver fatigue and highway driving: A simulator study. *Physiology and Behavior* 94 448-453.
- Vural, E., Cetin, M., Ercil, A., Littlewort, G., Bartlett, M., and Movellan, J. (2007). Drowsy driver detection through facial movement analysis. pages 6–18.
- Wang, Q., Yang, J., Ren, M., and Zheng, Y. (2006). Driver fatigue detection: A survey. *Proceedings 6th World Congr. Intell. Control Autom., Dalian, China*, pp. 8587-8591.
- Zhang, Z. and Zhang, J. (2006). Driver fatigue detection based intelligent vehicle control. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 1262–1265.
- Zhu, Z., Ji, Q., and Lan, P. (2004). Real time non-intrusive monitoring and prediction of driver fatigue. *IEEE Trans. Veh. Technol.*, 53:1052–1068.

AUTHOR INDEX

Adenauer, J.	98	Greggio, N.	36
Agostini, A.	160	Grigorie, L.	114
Alonso-Quesada, S.	133	Grigorie, T.	5, 13
Alvarado, M.	277	Grzechca, W.	259
Alvarez, D.	295	Gu, D.	80
Baboli, A.	269	Gusikhin, O.	285
Bahroun, Z.	269	Hahn, A.	249
Balic, J.	265	Hähner, J.	28
Belouaer, L.	154	Hamzaoui, A.	92
Benbrahim, M.	92	Han, K.	234
Bernardino, A.	36	Hawarah, L.	147
Berruet, P.	244	Henning, K.	104
Beth, C.	249	Hurling, B.	28
Betta, A.	92	Jacomino, M.	147
Botez, R.	5, 13, 114	Jedermann, R.	20
Bouamama, S.	86	Jeschke, S.	104
Bourki, A.	169	Kamenik, J.	249
Bouzid, M.	154	Kanso, M.	244
Braga, L.	255	Kastner, M.	230
Bremer, H.	230	Keller, H.	217
Brox, M.	213	Kościelnik, D.	182
Buniyamin, N.	125	Kottenstette, N.	53
Busch, J.	249	Koutsoukos, X.	53
Cabrera, A.	213	Kulas, A.	249
Campagne, J.	269	Kunze, R.	104
Celaya, E.	160	Lam, J.	141
Cohn, A.	255, 285	Lang, W.	20
Coulm, M.	169	LeBlanc, H.	53
Cramer, E.	249	Libatique, N.	193
Cruz Jr., J.	193	Machado, J.	199
Culaba, A.	193	MacNeille, P.	285
Cus, F.	265	Mamou, M.	5, 13, 114
Dayrit, F.	193	Masolkin, S.	74
DeSouza, G.	234	Matthes, J.	217
Dong, Y.	234	Mébaraki, Y.	5, 13, 114
Du, B.	141	Mekki, A.	45
Dzaack, J.	98	Miskowicz, M.	182
Essounbouli, N.	92	Mouaddib, A.	154
Eyisi, E.	53	Nagy, M.	174
Ferrara, P.	230	Necsulescu, D.	221
Ferreira, C.	209	Nikitin, S.	174
García, F.	277	Omar, R.	80
Gattringer, H.	230	Ommen, D.	249
Ghazel, M.	45	Orozco, Á.	295
Ghedira, K.	86	Palafox-Albarrán, J.	20

AUTHOR INDEX (CONT.)

Philippe, J.	244
Ploix, S.	147
Popov, A.	5, 13, 114
Promyslov, V.	74
Pruner, E.	221
Rached, M.	269
Ramakers, R.	104
Ramsauer, M.	230
Roetting, M.	98
Rolet, P.	169
Salazar, A.	295
Sánchez-Solano, S.	213
Santos, J.	255
Santos-Victor, J.	36
Sariff, N.	125
Sasiadek, J.	221
Seabra, E.	199
Sen, M.	133
Serra, G.	209
Shu, Z.	141
Siarry, P.	86
Sima, V.	187
Smairi, N.	86
Sohraiee, S.	65
Şuşu, A.	238
Sztipanovits, J.	53
Tan, R.	193
Tangonan, G.	193
Terzian, V.	174
Teytaud, O.	169
Toguyeni, A.	45
Tomforde, S.	28
Toro, E.	213
Vayssière, P.	169
Waibel, P.	217
Wegerich, A.	98
Zhang, L.	221
Znak, V.	226
Zouari, B.	269
Zuperl, U.	265

Proceedings Indexed by:

