ICINCO 2007

FOURTH INTERNATIONAL CONFERENCE ON INFORMATICS IN CONTROL, AUTOMATION AND ROBOTICS

Proceedings

Signal Processing, Systems Modeling and Control

ANGERS, FRANCE · MAY 9-12, 2007

ORGANIZED BY

INSTICC

IN COOPERATION WITH

aaai

CO-SPONSORED BY

IFAC

GdR MACS

CNRS

INSTICC PRESS

ICINCO 2007

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

Volume SPSMC

Angers, France

May 9 – 12, 2007

Co-organized by
INSTICC – Institute for Systems and Technologies of Information, Control and Communication
and
University of Angers

Co-sponsored by
IFAC - International Federation of Automatic Control
GDR MACS - Groupe de Recherche "Modélisation, Analyse et Conduite des Systèmes dynamiques
CNRS - Centre National de la Recherche Scientifique
and
EEA – Club des Enseignants en Electronique, Electrotechnique et Automatique

In Cooperation with
AAAI – Association for the Advancement of Artificial Intelligence

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

Printed in Portugal

ISBN: 978-972-8865-84-9

Depósito Legal: 257879/07

http://www.icinco.org

secretariat@icinco.org

BRIEF CONTENTS

INVITED SPEAKERS

Dimitar Filev

The Ford Motor Company

U.S.A.

Mark W. Spong

University of Illinois at Urbana-Champaign

U.S.A.

Patrick Millot

Université de Valenciennes

France

SPECIAL SESSION CHAIRS

Samir Ladaci

IRCCyN

France

Jean-Louis Boimond

LISA

France

Jean Jacques Loiseau

IRCCyN Nantes

France

Oleg Gusikhin

Ford Research & Adv. Engineering

U.S.A.

ORGANIZING AND STEERING COMMITTEES

Conference Co-chairs

Jean-Louis Ferrier, University of Angers, France

Joaquim Filipe, Polytechnic Institute of Setúbal / INSTICC, Portugal

Program Co-chairs

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

Janan Zaytoon, CReSTIC, URCA, France

Proceedings Production

Andreia Costa, INSTICC, Portugal

Bruno Encarnação, INSTICC, Portugal

Vitor Pedrosa, INSTICC, Portugal

CD-ROM Production

Paulo Brito, INSTICC, Portugal

Webdesigner and Graphics Production

Marina Carvalho, INSTICC, Portugal

Secretariat and Webmaster

Marina Carvalho, INSTICC, Portugal

PROGRAM COMMITTEE

Eugenio Aguirre, University of Granada, Spain

Arturo Hernandez Aguirre, Centre for Research in Mathematics, Mexico

Frank Allgower, University of Stuttgart, Germany

Fouad AL-Sunni, KFUPM, Saudi Arabia

Bala Amavasai, Sheffield Hallam University, U.K.

Francesco Amigoni, Politecnico di Milano, Italy

Yacine Amirat, University Paris 12, France

Nicolas Andreff, LASMEA, France

Stefan Andrei, National University of Singapore, Singapore

Plamen Angelov, Lancaster University, U.K.

Luis Antunes, GUESS/Universidade de Lisboa, Portugal

Peter Arato, Budapest University of Technology and Economics, Hungary

Helder Araújo, University of Coimbra, Portugal

Gustavo Arroyo-Figueroa, Instituto de Investigaciones Electricas, Mexico

Marco Antonio Arteaga, Universidad Nacional Autonoma de Mexico, Mexico

Vijanth Sagayan Asirvadam, University Technology Petronas, Malaysia

Nikos Aspragathos, University of Patras, Greece

Robert Babuska, TU Delft, The Netherlands

Ruth Bars, Budapest University of Technology and Economics, Hungary

Karsten Berns, University Kaiserslautern, Germany

Robert Bicker, University of Newcastle upon Tyne, U.K.

Stjepan Bogdan, University of Zagreb, Faculty of EE&C, Croatia

Patrick Boucher, SUPELEC, France

Alan Bowling, University of Notre Dame, U.S.A.

Edmund Burke, University of Nottingham, U.K.

Kevin Burn, University of Sunderland, U.K.

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

Luis M. Camarinha-Matos, New University of Lisbon, Portugal

Marco Campi, University of Brescia, Italy

Marc Carreras, University of Girona, Spain

Jorge Martins de Carvalho, FEUP, Portugal

Alicia Casals, Technical University of Catalonia, Spain

Alessandro Casavola, University of Calabria, Italy

Christos Cassandras, Boston University, U.S.A.

Riccardo Cassinis, University of Brescia, Italy

Raja Chatila, LAAS-CNRS, France

Tongwen Chen, University of Alberta, Canada

YangQuan Chen, Utah State University, U.S.A.

Albert M. K. Cheng, University of Houston, U.S.A.

Graziano Chesi, University of Hong Kong, China

Sung-Bae Cho, Yonsei University, Korea

Ryszard S. Choras, University of Technology & Agriculture, Poland

Carlos Coello Coello, CINVESTAV-IPN, Mexico

Patrizio Colaneri, Politecnico di Milano, Italy

António Dourado Correia, University of Coimbra, Portugal

Yechiel Crispin, Embry-Riddle University, U.S.A.

Keshav Dahal, University of Bradford, U.K.

Mariolino De Cecco, DIMS - University of Trento, Italy

Bart De Schutter, Delft University of Technology, The Netherlands

Angel P. del Pobil, Universitat Jaume I, Spain

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

Rüdiger Dillmann, University of Karlsruhe, Germany

Feng Ding, Southern Yangtze University, China

Denis Dochain, Université Catholique de Louvain, Belgium

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

Alexandre Dolgui, Ecole des Mines de Saint Etienne, France

PROGRAM COMMITTEE (CONT.)

Marco Dorigo, Université Libre de Bruxelles, Belgium

Petr Ekel, Pontifical Catholic University of Minas Gerais, Brazil

Heinz-Hermann Erbe, TU Berlin, Germany

Gerardo Espinosa-Perez, Universidad Nacional Autonoma de Mexico, Mexico

Simon Fabri, University of Malta, Malta

Sergej Fatikow, University of Oldenburg, Germany

Jean-Marc Faure, Ecole Normale Superieure de Cachan, France

Jean-Louis Ferrier, Université d'Angers, France

Florin Gheorghe Filip, The Romanian Academy & The National Institute for R&D in Informatics (ICI), Romania

Georg Frey, University of Kaiserslautern, Germany

Manel Frigola, Technical University of Catalonia (UPC), Spain

Colin Fyfe, University of paisley, U.K.

Dragan Gamberger, Rudjer Boskovic Institute, Croatia

Leonardo Garrido, Tecnológico de Monterrey, Mexico

Ryszard Gessing, Silesian University of Technology, Poland

Lazea Gheorghe, Technical University of Cluj-Napoca, Romania

Maria Gini, University of Minnesota, U.S.A.

Alessandro Giua, University of Cagliari, Italy

Luis Gomes, Universidade Nova de Lisboa, Portugal

John Gray, University of Salford, U.K.

Dongbing Gu, University of Essex, U.K.

Jason Gu, Dalhousie University, Canada

José J. Guerrero, Universidad de Zaragoza, Spain

Jatinder (Jeet) Gupta, University of Alabama in Huntsville, U.S.A.

Thomas Gustafsson, Luleå University of Technology, Sweden

Maki K. Habib, Saga University, Japan

Hani Hagras, University of Essex, U.K.

Wolfgang Halang, Fernuniversitaet, Germany

J. Hallam, University of Southern Denmark, Denmark

Riad Hammoud, Delphi Electronics & Safety, U.S.A.

Uwe D. Hanebeck, Institute of Computer Science and Engineering, Germany

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

Robert Harrison, The University of Sheffield, U.K.

Vincent Hayward, McGill Univ., Canada

Dominik Henrich, University of Bayreuth, Germany

Francisco Herrera, University of Granada, Spain

Victor Hinostroza, University of Ciudad Juarez, Mexico

Weng Ho, National University of Singapore, Singapore

Wladyslaw Homenda, Warsaw University of Technology, Poland

Alamgir Hossain, Bradford University, U.K.

Dean Hougen, University of Oklahoma, U.S.A.

Amir Hussain, University of Stirling, U.K.

Seth Hutchinson, University of Illinois, U.S.A.

Atsushi Imiya, IMIT Chiba Uni, Japan

Sirkka-Liisa Jämsä-Jounela, Helsinki University of Technology, Finland

Ray Jarvis, Monash University, Australia

Odest Jenkins, Brown University, U.S.A.

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

Ivan Kalaykov, Örebro University, Sweden

Dimitrios Karras, Chalkis Institute of Technology, Greece

Dusko Katic, Mihailo Pupin Institute, Serbia

Graham Kendall, The University of Nottingham, U.K.

Uwe Kiencke, University of Karlsruhe (TH), Germany

Jozef Korbicz, University of Zielona Gora, Poland

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

PROGRAM COMMITTEE (CONT.)

Bart Kosko, University of Southern California, U.S.A.

George L. Kovács, Hungarian Academy of Sciences, Hungary

Krzysztof Kozlowski, Poznan University of Technology, Poland

Gerhard Kraetzschmar, Fraunhofer Institute for Autonomous Intelligent Systems, Germany

Cecilia Laschi, Scuola Superiore Sant'Anna, Italy

Loo Hay Lee, National University of Singapore, Singapore

Soo-Young Lee, KAIST, Korea

Graham Leedham, University of New South Wales (Asia), Singapore

Cees van Leeuwen, RIKEN BSI, Japan

Kauko Leiviskä, University of Oulu, Finland

Kang Li, Queen's University Belfast, U.K.

Yangmin Li, University of Macau, China

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

Cheng-Yuan Liou, National Taiwan University, Taiwan

Vincenzo Lippiello, Università Federico II di Napoli, Italy

Honghai Liu, University of Portsmouth, U.K.

Luís Seabra Lopes, Universidade de Aveiro, Portugal

Brian Lovell, The University of Queensland, Australia

Peter Luh, University of Connecticut, U.S.A.

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

N. P. Mahalik, Gwangju Institute of Science and Technology, Korea

Bruno Maione, Politecnico di Bari, Italy

Frederic Maire, Queensland University of Technology, Australia

Om Malik, University of Calgary, Canada

Danilo Mandic, Imperial College, U.K.

Jacek Mandziuk, Warsaw University of Technology, Poland

Hervé Marchand, INRIA, France

Philippe Martinet, LASMEA, France

Aleix Martinez, Ohio State University, U.S.A.

Aníbal Matos, Faculdade de Engenharia da Universidade do Porto (FEUP), Portugal

Rene V. Mayorga, University of Regina, Canada

Barry McCollum, Queen's University Belfast, U.K.

Ken McGarry, University of Sunderland, U.K.

Gerard McKee, The University of Reading, U.K.

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

Basil Mertzios, (1)Thessaloniki Institute of Technology, (2) Democritus University, Greece

José Mireles Jr., Universidad Autonoma de Ciudad Juarez, Mexico

Sushmita Mitra, Indian Statistical Institute, India

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

Rafael Muñoz-Salinas, University of Cordoba, Spain

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

Ould Khessal Nadir, Okanagan College, Canada

Fazel Naghdy, University of Wollongong, Australia

Tomoharu Nakashima, Osaka Prefecture University, Japan

Andreas Nearchou, University of Patras, Greece

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

Sergiu Nedevschi, Technical University of Cluj-Napoca, Romania

Maria Neves, Instituto Superior de Engenharia do Porto, Portugal

Hendrik Nijmeijer, Eindhoven University of Technology, The Netherlands

Juan A. Nolazco-Flores, ITESM, Campus Monterrey, Mexico

Urbano Nunes, University of Coimbra, Portugal

Gustavo Olague, CICESE, Mexico

PROGRAM COMMITTEE (CONT.)

José Valente de Oliveira, Universidade do Algarve, Portugal

Andrzej Ordys, Kingston University in London, Faculty of Engineering, U.K.

Djamila Ouelhadj, University of Nottingham, ASAP GROUP (Automated Scheduling, Optimisation and Planning), U.K.

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

Christos Panayiotou, University of Cyprus, Cyprus

Evangelos Papadopoulos, NTUA, Greece

Panos Pardalos, University of Florida, U.S.A.

Michel Parent, INRIA, France

Thomas Parisini, University of Trieste, Italy

Igor Paromtchik, RIKEN, Japan

Gabriella Pasi, Università degli Studi di Milano Bicocca, Italy

Witold Pedrycz, University of Alberta, Canada

Carlos Eduardo Pereira, Federal University of Rio Grande do Sul - UFRGS, Brazil

Maria Petrou, Imperial College, U.K.

J. Norberto Pires, University of Coimbra, Portugal

Marios Polycarpou, University of Cyprus, Cyprus

Marie-Noëlle Pons, CNRS, France

Libor Preucil, Czech Technical University in Prague, Czech Republic

Joseba Quevedo, Technical University of Catalonia, Spain

Robert Reynolds, Wayne State University, U.S.A.

A. Fernando Ribeiro, Universidade do Minho, Portugal

Bernardete Ribeiro, University of Coimbra, Portugal

Robert Richardson, University of Manchester, U.K.

John Ringwood, National University of Ireland (NUI), Maynooth, Ireland

Rodney Roberts, Florida State University, U.S.A.

Kurt Rohloff, BBN Technologies, U.S.A.

Juha Röning, University of Oulu, Finland

Agostinho Rosa, IST, Portugal

Hubert Roth, University Siegen, Germany

António Ruano, CSI, Portugal

Carlos Sagüés, University of Zaragoza, Spain

Mehmet Sahinkaya, University of Bath, U.K.

Antonio Sala, Universidad Politecnica de Valencia, Spain

Abdel-Badeeh Salem, Ain Shams University, Egypt

Ricardo Sanz, Universidad Politécnica de Madrid, Spain

Medha Sarkar, Middle Tennessee State University, U.S.A.

Nilanjan Sarkar, Vanderbilt University, U.S.A.

Jurek Sasiadek, Carleton University, Canada

Daniel Sbarbaro, Universidad de Concepcion, Chile

Carsten Scherer, Delft University of Technology, The Netherlands

Carla Seatzu, University of Cagliari, Italy

Klaus Schilling, University Würzburg, Germany

Yang Shi, University of Saskatchewan, Canada

Michael Short, University of Leicester, U.K.

Chi-Ren Shyu, University of Missouri-Columbia, U.S.A.

Bruno Siciliano, Università di Napoli Federico II, Italy

João Silva Sequeira, Instituto Superior Técnico, Portugal

Silvio Simani, University of Ferrara, Italy

Amanda Sharkey, University of Sheffield, U.K.

Michael Small, Hong Kong Polytechnic University, China

Burkhard Stadlmann, University of Applied Sciences Wels, Austria

Tarasiewicz Stanislaw, Université Laval, Canada

Aleksandar Stankovic, Northeastern University, U.S.A.

Raúl Suárez, Universitat Politecnica de Catalunya (UPC), Spain

PROGRAM COMMITTEE (CONT.)

Ryszard Tadeusiewicz, AGH University of Science and Technology, Poland

Tianhao Tang, Shanghai Maritime University, China

Adriana Tapus, University of Southern California, U.S.A.

József K. Tar, Budapest Tech Polytechnical Institution, Hungary

Daniel Thalmann, EPFL, Switzerland

Gui Yun Tian, University of Huddersfield, U.K.

Antonios Tsourdos, Cranfield University, U.K.

Nikos Tsourveloudis, Technical University of Crete, Greece

Ivan Tyukin, RIKEN Brain Science Institute, Japan

Masaru Uchiyama, Tohoku University, Japan

Nicolas Kemper Valverde, Universidad Nacional Autónoma de México, Mexico

Marc Van Hulle, K. U. Leuven, Belgium

Annamaria R. Varkonyi-Koczy, Budapest University of Technology and Economics, Hungary

Luigi Villani, Università di Napoli Federico II, Italy

Markus Vincze, Technische Universität Wien, Austria

Bernardo Wagner, University of Hannover, Germany

Axel Walthelm, sepp.med gmbh, Germany

Lipo Wang, Nanyang Technological University, Singapore

Alfredo Weitzenfeld, ITAM, Mexico

Dirk Wollherr, Technische Universität München, Germany

Sangchul Won, Pohang University of Science and Technology, Korea

Kainam Thomas Wong, The Hong Kong Polytechnic University, China

Jeremy Wyatt, University of Birmingham, U.K.

Alex Yakovlev, University of Newcastle, U.K.

Hujun Yin, University of Manchester, U.K.

Xinghuo Yu, Royal Melbourne Institute of Techology, Australia

Du Zhang, California State University, U.S.A.

Janusz Zalewski, Florida Gulf Coast University, U.S.A.

Marek Zaremba, Universite du Quebec, Canada

Dayong Zhou, University of Oklahoma, U.S.A.

Argyrios Zolotas, Loughborough University, U.K.

Albert Zomaya, The University of Sydney, Austrália

AUXILIARY REVIEWERS

Rudwan Abdullah, University of Stirling, U.K.

Luca Baglivo, University of Padova, Italy

Prasanna Balaprakash, IRIDIA, Université Libre de Bruxelles, Belgium

João Balsa, Universidade de Lisboa, Portugal

Alejandra Barrera, ITAM, Mexico

Frederik Beutler, Intelligent Sensor-Actuator-Systems Laboratory - Universität Karlsruhe (TH), Germany

Alecio Binotto, CETA SENAI-RS, Brazil

Nizar Bouguila, Concordia University, Canada

Dietrich Brunn, Intelligent Sensor-Actuator-Systems Laboratory - Universität Karlsruhe (TH), Germany

Maria Paola Cabasino, Dip.to Ingegneria Elettrica ed Elettronica Universita' di Cagliari, Italy

Joao Paulo Caldeira, EST-IPS, Portugal

Aneesh Chauhan, Universidade de Aveiro, Portugal

Paulo Gomes da Costa, FEUP, Portugal

Xevi Cufi, University of Girona, Spain

Sérgio Reis Cunha, FEUP, Portugal

Paul Dawson, Boise State University, U.S.A.

Mahmood Elfandi, Elfateh University, Libya

AUXILIARY REVIEWERS (CONT.)

Michele Folgheraiter, Politecnico di Milano, Italy

Diamantino Freitas, FEUP, Portugal

Reinhard Gahleitner, University of Applied Sciences Wels, Austria

Nils Hagge, Leibniz Universität Hannover, Institute for Systems Engineering, Germany

Onur Hamsici, The Ohio State Univeristy, U.S.A.

Renato Ventura Bayan Henriques, UFRGS, Brazil

Matthias Hentschel, Leibniz Universität Hannover, Institute for Systems Engineering, Germany

Marco Huber, Intelligent Sensor-Actuator-Systems Laboratory - Universität Karlsruhe (TH), Germany

Markus Kemper, University of Oldenbvurg, Germany

Vesa Klumpp, Intelligent Sensor-Actuator-Systems Laboratory - Universität Karlsruhe (TH), Germany

Daniel Lecking, Leibniz Universität Hannover, Institute for Systems Engineering, Germany

Gonzalo Lopez-Nicolas, University of Zaragoza, Spain

Cristian Mahulea, University of Zaragoza, Spain

Cristian Mahulea, Dep.to Informática e Ingeneiría de Sistemas Centro Politécnico Superior, Spain

Nikolay Manyakov, K. U. Leuven, Belgium

Antonio Muñoz, University of Zaragoza, Spain

Ana C. Murillo, University of Zaragoza, Spain

Andreas Neacrhou, University of Patras, Greece

Marco Montes de Oca, IRIDIA, Université Libre de Bruxelles, Belgium

Sorin Olaru, Supelec, France

Karl Pauwels, K. U. Leuven, Belgium

Luis Puig, University of Zaragoza, Spain

Ana Respício, Universidade de Lisboa, Portugal

Pere Ridao, University of Girona, Spain

Kathrin Roberts, Intelligent Sensor-Actuator-Systems Laboratory - Universität Karlsruhe (TH), Germany

Paulo Lopes dos Santos, FEUP, Portugal

Felix Sawo, Intelligent Sensor-Actuator-Systems Laboratory - Universität Karlsruhe (TH), Germany

Frederico Schaf, UFRGS, Brazil

Oliver Schrempf, Intelligent Sensor-Actuator-Systems Laboratory - Universität Karlsruhe (TH), Germany

Torsten Sievers, University of Oldenbvurg, Germany

Razvan Solea, Institute of Systems and Robotics, University of Coimbra, Portugal

Wolfgang Steiner, University of Applied Sciences Wels, Austria

Christian Stolle, University of Oldenbvurg, Germany

Alina Tarau, Delft University of Technology, The Netherlands

Rui Tavares, University of Evora, Portugal

Paulo Trigo, ISEL, Portugal

Haralambos Valsamos, University of Patras, Greece

José Luis Villarroel, University of Zaragoza, Spain

Yunhua Wang, University of Oklahoma, U.S.A.

Florian Weißel, Intelligent Sensor-Actuator-Systems Laboratory - Universität Karlsruhe (TH), Germany

Jiann-Ming Wu, National Dong Hwa University, Taiwan

Oliver Wulf, Leibniz Universität Hannover, Institute for Systems Engineering, Germany

Ali Zayed, Seventh of April University, Libya

Yan Zhai, University of Oklahoma, U.S.A.

SELECTED PAPERS BOOK

A number of selected papers presented at ICINCO 2007 will be published by Springer, in a book entitled Informatics in Control, Automation and Robotics IV. This selection will be done by the conference co-chairs and program co-chairs, among the papers actually presented at the conference, based on a rigorous review by the ICINCO 2007 program committee members.

FOREWORD

Welcome to the 4th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2007) held at the University of Angers. 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 require Information Technology support.

In this year Conference Program we have included oral presentations (full papers and short papers) as well as posters, organized in three simultaneous tracks: "Intelligent Control Systems and Optimization", "Robotics and Automation" and "Systems Modeling, Signal Processing and Control". Furthermore, ICINCO 2007 includes 2 satellite workshops and 3 plenary keynote lectures, given by internationally recognized researchers

The two satellite workshops that are held in conjunction with ICINCO 2007 are: Third International Workshop on Multi-Agent Robotic Systems (MARS 2007) and Third International Workshop on Artificial Neural Networks and Intelligent Information Processing (ANNIIP 2007).

As additional points of interest, it is worth mentioning that the Conference Program includes a plenary panel subject to the theme "Practical Applications of Intelligent Control and Robotics" and 3 Special Sessions focused on very specialized topics.

ICINCO has received 435 paper submissions, not including workshops, from more than 50 countries, in all continents. To evaluate each submission, a double blind paper review was performed by the program committee, whose members are researchers in one of ICINCO main topic areas. Finally, only 263 papers are published in these proceedings and presented at the conference; of these, 195 papers were selected for oral presentation (52 full papers and 143 short papers) and 68 papers were selected for poster presentation. The global acceptance ratio was 60,4% and the full paper acceptance ratio was 11,9%. After the conference, some authors will be invited to publish extended versions of their papers in a journal and a short list of about thirty papers will be included in a book that will be published by Springer with the best papers of ICINCO 2007.

In order to promote the development of research and professional networks the conference includes in its social program a Town Hall Reception in the evening of May 9 (Wednesday) and a Conference and Workshops Social Event & Banquet in the evening of May 10 (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

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 secretariat; 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 2008, next May, at the Hotel Tivoli Ocean Park, Funchal, Madeira, Portugal.

Janan Zaytoon

CReSTIC, URCA, France

Juan Andrade Cetto

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

Jean-Louis Ferrier

Université d'Angers, France

Joaquim Filipe

Polytechnic Institute of Setúbal / INSTICC, Portugal

CONTENTS

SHORT PAPERS

POSTERS

SPECIAL SESSION ON FRACTIONAL ORDER SYSTEMS

INVITED SPEAKERS

KEYNOTE LECTURES

REAL TIME DIAGNOSTICS, PROGNOSTICS, & PROCESS MODELING

Dimitar Filev
The Ford Motor Company
U.S.A.

Abstract: Practical and theoretical problems related to the design of real time diagnostics, prognostics, & process modeling systems are discussed. Major algorithms for autonomous monitoring of machine health in industrial networks are proposed and relevant architectures for incorporation of intelligent prognostics within plant floor information systems are reviewed. Special attention is given to the practical realization of real time structure and parameter learning algorithms. Links between statistical process control and real time modeling based on the evolving system paradigm are analyzed relative to the design of soft sensing algorithms. Examples and case studies of industrial implementation of aforementioned concepts are presented.

BRIEF BIOGRAPHY

Dr. Dimitar P. Filev is a Senior Technical Leader, Intelligent Control & Information Systems with Ford Motor Company specializing in industrial intelligent systems and technologies for control, diagnostics and decision making. He is conducting research in systems theory and applications, modeling of complex systems, intelligent modeling and control and he has published 3 books, and over 160 articles in refereed journals and conference proceedings. He holds15 granted U.S. patents and numerous foreign patents in the area of industrial intelligent systems Dr. Filev is a recipient of the '95 Award for Excellence of MCB University Press and was awarded 4 times with the Henry Ford Technology Award for development and implementation of advanced intelligent control technologies. He is Associate Editor of Int. J. of General Systems and Int. J. of Approximate Reasoning. He is a member of the Board of Governors of the IEEE Systems, Man & Cybernetics Society and President of the North American Fuzzy Information Processing Society (NAFIPS). Dr. Filev received his PhD. degree in Electrical Engineering from the Czech Technical University in Prague in 1979.

SYNCHRONIZATION OF MULTI-AGENT SYSTEMS

Mark W. Spong
Donald Biggar Willett Professor of Engineering
Professor of Electrical and Computer Engineering
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
U.S.A.

Abstract: There is currently great interest in the control of multi-agent networked systems. Applications include mobile sensor networks, teleoperation, synchronization of oscillators, UAV's and coordination of multiple robots. In this talk we consider the output synchronization of networked dynamic agents using passivity theory and considering the graph topology of the inter-agent communication. We provide a coupling control law that results in output synchronization and we discuss the extension to state synchronization in addition to output synchronization. We also consider the extension of these ideas to systems with time delay in communication among agents and obtain results showing synchronization for arbitrary time delay. We will present applications of our results in synchronization of Kuramoto oscillators and in bilateral teleoperators.

BRIEF BIOGRAPHY

Mark W. Spong received the B.A. degree, magna cum laude and Phi Beta Kappa, in mathematics and physics from Hiram College, Hiram, Ohio in 1975, the M.S. degree in mathematics from New Mexico State University in 1977, and the M.S. and D.Sc. degrees in systems science and mathematics in 1979 and 1981, respectively, from Washington University in St. Louis. Since 1984 he has been at the University of Illinois at Urbana-Champaign where he is currently a Donald Biggar Willett Distinguished Professor of Engineering, Professor of Electrical and Computer Engineering, and Director of the Center for Autonomous Engineering Systems and Robotics. Dr. Spong is Past President of the IEEE Control Systems Society and a Fellow of the IEEE. Dr. Spong's main research interests are in robotics, mechatronics, and nonlinear control theory. He has published more than 200 technical articles in control and robotics and is co-author of four books. His recent awards include the Senior U.S. Scientist Research Award from the Alexander von Humboldt Foundation, the Distinguished Member Award from the IEEE Control Systems Society, the John R. Ragazzini and O. Hugo Schuck Awards from the American Automatic Control Council, and the IEEE Third Millennium Medal.

TOWARD HUMAN-MACHINE COOPERATION

Patrick Millot
Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielle et Humaine
Université de Valenciennes
France

Abstract: In human machine systems human activities are mainly oriented toward decision-making: monitoring and fault detection, fault anticipation, diagnosis and prognosis, and fault prevention and recovery. The objectives combine the human-machine system performances (production quantity and quality) as well as the global system safety. In this context human operators may have a double role: (1) a negative role as they may perform unsafe or erroneous actions on the process, (2) a positive role as they can detect, prevent or recover an unsafe process behavior due to an other operator or to automated decision makers. Two approachs to these questions are combined in a pluridisciplinary research way : (1) human engineering which aims at designing dedicated assistance tools for human operators and at integrating them into human activities through a human machine cooperation, (2) cognitive psychology and ergonomics analysing the human activities, the need for such tools and their use. This paper focuses on the concept of cooperation and proposes a framework for implementation. Examples in Air Traffic Control and in Telecommunication networks illustrate these concepts.

BRIEF BIOGRAPHY

Born in 53 he received a PhD in Automatic Control (79) an is Docteur d'Etat es Sciences (87). He is full Professor at the University of Valenciennes since 89. He conducts research on Automation Sciences, Artificial Intelligence, Supervisory Control, Human Machine Systems, Human Reliability with applications to production telecommunication and transport systems (Air Traffic Control, Car Traffic, Trains Metro.). His scientific production covers about 175 publications, collective books, conference proceedings. Research Director of 35 PhD students and 9 HDR since 89, reviewer of 50 PhD Thesis and 9 HDR from other universities. Head of the research group "Human Machine Systems" in LAMIH since 87 till 04 (25 researchers). Vice-head then head of LAMIH between 96 and 05 (222 researchers and engineers). Vice Chairman of the University of Valenciennes since October 05 in charge of research.

Scientific head or Member of the scientific board or Manager of several regional research groups on Supervisory Control (GRAISYHM 96-02) on Transport System Safety (GRRT since 87,pôle ST2 since 01 with 80 researchers of 10 labs). Member of the French Council of the Universities (96-03), member of the scientific board of the french national research group in Automation Sciences supported by CNRS (96-01). Partner of several European projects and netwoks (HCM networks 93-96, 2 projects since 02 on Urban Guided Transport Management Systems and the Network of Excellence EURNEX since 04). Member of the IFAC Technical Committee 4.5 Human Machine Systems since 00. IPC member of several International Conferences and Journals.

SIGNAL PROCESSING, SYSTEMS MODELING AND CONTROL

FULL PAPERS

INNER AND OUTER APPROXIMATION OF CAPTURE BASIN USING INTERVAL ANALYSIS

Mehdi Lhommeau[1], Luc Jaulin[2] and Laurent Hardouin[1]
[1]*Laboratoire d'Ingénierie des Systèmes Automatisés, Université d'Angers, 62 av. Notre Dame du Lac, 49000 Angers, France*
[2]E^{3}I^2, *ENSIETA, 2 rue Franois Verny, 29806 Brest Cédex 09, France*
mehdi.lhommeau@univ-angers.fr, luc.jaulin@ensieta.fr, laurent.hardouin@istia.univ-angers.fr

Keywords: Interval Analysis, Viability theory, Capture basin.

Abstract: This paper proposes a new approach to solve the problem of computing the capture basin $\mathbb{C}$ of a target $\mathbb{T}$. The capture basin corresponds to the set of initial states such that the target is reached in finite time before possibly leaving of constrained set. We present an algorithm, based on interval analysis, able to characterize an inner and an outer approximation $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$ of the capture basin. The resulting algorithm is illustrated on the Zermelo problem.

1 INTRODUCTION AND NOTATIONS

Consider a nonlinear continuous-time system

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)),\ \mathbf{x}(t) \in \mathbb{R}^n,\ \mathbf{u}(t) \in \mathbb{R}^m \quad (1)$$

We shall assume that the function $\mathbf{f}$ is sufficiently regular to guarantee that for all piecewise continuous function $\mathbf{u}(.)$ the solution of $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$ is unique. The state vector $\mathbf{x}(t)$ is not allowed to exit a given compact set $\mathbb{K} \subset \mathbb{R}^n$ and the input $\mathbf{u}(t)$ should belong to a given compact set $\mathbb{U} \subset \mathbb{R}^m$.
Define the flow function $\phi(t, \mathbf{x}_0, \mathbf{u})$ as the solution of $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ for the initial vector $\mathbf{x}_0$ and for the input function $\mathbf{u}$. The path from t_1 to t_2 is defined by

$$\phi([t_1, t_2], \mathbf{x}_0, \mathbf{u}) \stackrel{\text{def}}{=} \big\{\mathbf{x} \in \mathbb{R}^n, \exists t \in [t_1, t_2], \mathbf{x} = \phi(t, \mathbf{x}_0, \mathbf{u})\big\}. \quad (2)$$

Define a target set $\mathbb{T} \subset \mathbb{K} \subset \mathbb{R}^n$ as a closed set we would like to reach for one $t \geq 0$. The *capture basin* $\mathbb{C}$ of $\mathbb{T}$ is the set of initial states $x \in \mathbb{K}$ for which there exists an admissible control $\mathbf{u} \in \mathcal{F}([0,t] \to \mathbb{U})$ and a finite time $t \geq 0$ such that the trajectory $\phi([0,t], \mathbf{x}_0, \mathbf{u})$ with the dynamic f under the control $\mathbf{u}$ lives in $\mathbb{K}$ and reaches $\mathbb{T}$ at time t :

$$\mathbb{C} \stackrel{\text{def}}{=} \big\{\mathbf{x}_0 \in \mathbb{K}, \exists t \geq 0, \exists \mathbf{u} \in \mathcal{F}([0,t] \to \mathbb{U}), \phi(t, \mathbf{x}_0, \mathbf{u}) \in \mathbb{T} \text{ and } \phi([0,t], \mathbf{x}_0, \mathbf{u}) \subset \mathbb{K}\big\}. \quad (3)$$

The aim of the paper is to provide an algorithm able to compute an inner and an outer approximation of $\mathbb{C}$, i.e., to find two subsets $\mathbb{C}^-$ and $\mathbb{C}^+$ such that

$$\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+.$$

Our contribution is twofold. First we shall introduce interval analysis in the context of viability problem (Aubin, 1991; Saint-Pierre, 1994; Cruck et al., 2001). Second, we shall provide the first algorithm able to compute a garanteed inner and an outer approximation for capture basins.

2 INTERVAL ANALYSIS

The interval theory was born in the 60's aiming rigorous computations using finite precision computers (see (Moore, 1966)). Since its birth, it has been developed and it proposed today orignal algorithms for solving problems independently to the finite precision of computers computations, although reliable computations using finite precision remains one important advantage of the interval based algorithms (Kearfott and Kreinovich, 1996).
An *interval* $[x]$ is a closed and connected subset of $\mathbb{R}$. A *box* $[\mathbf{x}]$ of $\mathbb{R}^n$ is a Cartesian product of n intervals. The set of all boxes of $\mathbb{R}^n$ is denoted by $\mathbb{IR}^n$. Note that $\mathbb{R}^n =]-\infty, \infty[\times \cdots \times]-\infty, \infty[$ is an element of $\mathbb{IR}^n$.

Basic operations on real numbers or vectors can be extended to intervals in a natural way.

Example 1 *If* $[t]=[t_1,t_2]$ *is an interval and* $[\mathbf{x}]=\left[x_1^-,x_1^+\right]\times\left[x_2^-,x_2^+\right]$ *is a box, then the product* $[t]*[\mathbf{x}]$ *is defined as follows*

$$[t_1,t_2]*\left(\begin{bmatrix}x_1^-,x_1^+\\x_2^-,x_2^+\end{bmatrix}\right)=\left(\begin{bmatrix}[t_1,t_2]*\left[x_1^-,x_1^+\right]\\[t_1,t_2]*\left[x_2^-,x_2^+\right]\end{bmatrix}\right)=$$
$$\left(\begin{array}{l}[\min(t_1x_1^-,t_1x_1^+,t_2x_1^-,t_2\,x_1^+),\\[\min(t_1x_2^-,t_1x_2^+,t_2x_2^-,t_2x_2^+),\\ \qquad\max(t_1x_1^-,t_1x_1^+,t_2x_1^-,t_2x_1^+)]\\ \qquad\max(t_1x_2^-,t_1x_2^+,t_2x_2^-,t_2x_2^+)]\end{array}\right).$$

2.1 Inclusion Function

The function $[\mathbf{f}](.):\mathbb{IR}^n\rightarrow\mathbb{IR}^p$ is an *inclusion function* of a function $\mathbf{f}:\mathbb{R}^n\rightarrow\mathbb{R}^p$ if

$$\forall[\mathbf{x}]\in\mathbb{IR}^n,\mathbf{f}([\mathbf{x}])\triangleq\{\mathbf{f}(\mathbf{x})\mid\mathbf{x}\in[\mathbf{x}]\}\ \subset\ [\mathbf{f}]([\mathbf{x}]).$$

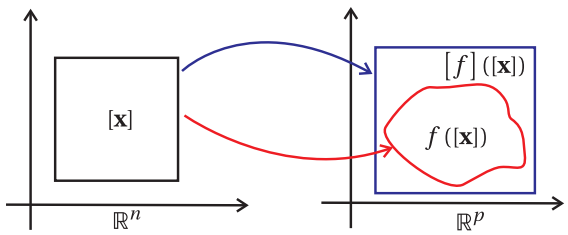


Illustration of inclusion function.

Interval computation makes it possible to obtain inclusion functions of a large class of nonlinear functions, as illustrated by the following example.

Example 2 *If* $\mathbf{f}(x_1,x_2)\triangleq((1-0.01x_2)x_1;(-1+0.02x_1)x_2)$, *a methodology to obtain an enclosure of the image set* $\mathbf{f}([10,20],[40,50])$ *is as follows:*

$$\mathbf{f}\left(\begin{bmatrix}[40,50]\\[10,20]\end{bmatrix}\right)\subset\left(\begin{array}{l}(1-0.01*[40,50])*[10,20]\\(-1+0.02*[10,20])*[40,50]\end{array}\right)$$
$$=\left(\begin{array}{l}(1-[0.4,0.5])*[10,20]\\(-1+[0.2,0.4])*[40,50]\end{array}\right)$$
$$=\left(\begin{array}{l}[0.5,0.6]*[10,20]\\[-0.8,-0.6]*[40,50]\end{array}\right)=\left(\begin{array}{l}([5,12])\\([-40,-24])\end{array}\right).$$

This methodology can easily be applied for any box $[x_1]\times[x_2]$ and the resulting algorithm corresponds to an inclusion function for $\mathbf{f}$.

The *interval union* $[\mathbf{x}]\sqcup[\mathbf{y}]$ of two boxes $[\mathbf{x}]$ and $[\mathbf{y}]$ is the smallest box which contains the union $[\mathbf{x}]\cup[\mathbf{y}]$. The *width* $w([\mathbf{x}])$ of a box $[\mathbf{x}]$ is the length of its largest side.

The ε-*inflation* of a box $[\mathbf{x}]=[x_1^-,x_1^+]\times\cdots\times[x_n^-,x_n^+]$ is defined by

$$\text{inflate}\,([\mathbf{x}],\varepsilon)\triangleq[x_1^--\varepsilon,x_1^++\varepsilon]\times\cdots$$
$$\cdots\times[x_n^--\varepsilon,x_n^++\varepsilon].\quad(4)$$

2.2 Picard Theorem

Interval analysis for ordinary differential equations were introduced by Moore (Moore, 1966) (See (Nedialkov et al., 1999) for a description and a bibliography on this topic). These methods provide numerically reliable enclosures of the exact solution of differential equations. These techniques are based on Picard Theorem.

Theorem 1 *Let* t_1 *be a positive real number. Assume that* $\mathbf{x}(0)$ *is known to belong to the box* $[\mathbf{x}](0)$. *Assume that* $\mathbf{u}(t)\in[\mathbf{u}]$ *for all* $t\in[0,t_1]$. *Let* $[\mathbf{w}]$ *be a box (that is expected to enclose the path* $\mathbf{x}(\tau),\tau\in[0,t_1]$*). If*

$$[\mathbf{x}](0)+[0,t_1]*[\mathbf{f}]([\mathbf{w}],[\mathbf{u}])\subset[\mathbf{w}],\quad(5)$$

where $[\mathbf{f}]([\mathbf{x}].[\mathbf{u}])$ *is an inclusion function of* $\mathbf{f}(\mathbf{x},\mathbf{u})$, *then, for all* $t\in[0,t_1]$

$$\mathbf{x}([0,t_1])\subset[\mathbf{x}](0)+[0,t_1]*[\mathbf{f}]([\mathbf{w}],[\mathbf{u}]).\quad(6)$$

2.3 Interval Flow

Definition: The *inclusion function of the flow* is a function

$$[\phi]:\left\{\begin{array}{ccc}\mathbb{IR}\times\mathbb{IR}^n\times\mathbb{IR}^m&\rightarrow&\mathbb{IR}^n\\([t],[\mathbf{x}],[\mathbf{u}])&\rightarrow&[\phi]([t],[\mathbf{x}],[\mathbf{u}])\end{array}\right.$$

such that

$$\forall t\in[t],\forall\mathbf{x}\in[\mathbf{x}],\mathbf{u}\in\mathcal{F}([t]\rightarrow[\mathbf{u}]),\phi(t,\mathbf{x},\mathbf{u})\in[\phi]([t],[\mathbf{x}],[\mathbf{u}])$$

Using Theorem 1, one can build an algorithm computing an enclosure $[\mathbf{x}]([t])$ for the path $\mathbf{x}([t])=\{\mathbf{x}(t),t\in[t]\}$ from an enclosure $[\mathbf{x}]$ for $\mathbf{x}(0)$. The principle of this algorithm is illustrated by Figure 1.

Comments : The interval $[t]=[t_1,t_2]$ is such that $t_1\geq0$. Step 2 computes an estimation $[\hat{\mathbf{x}}](t_2)$ for the domain of all $\mathbf{x}(t_1)$ consistent with the fact that $\mathbf{x}(0)\in[\mathbf{x}]$. Note that, at this level, it is not certain that $[\hat{\mathbf{x}}](t_2)$ contains $\mathbf{x}(t_2)$. Step 3 computes the smallest box $[\mathbf{v}]$ containing $[\mathbf{x}](t_1)$ and $[\hat{\mathbf{x}}](t_2)$. At Step 4, $[\mathbf{v}]$ is inflated (see (4)) to provide a good candidate for $[\mathbf{w}]$. α and β are small positive numbers. Step 5 checks the condition of Theorem 1. If the condition is not satisfied, no bounds can be computed for $\mathbf{x}(t_2)$ and $\mathbb{R}^n$ is returned. Otherwise, Step 8 computes a box containing $\mathbf{x}(t_2)$ using theorem 1. ■

Algorithm 1: Inclusion function $[\phi.]$

Data: $[t]=[t_1,t_2],[\mathbf{x}](t_1),[\mathbf{u}]$
Result: $[\mathbf{x}](t_2)$
1 **begin**
2 $\quad[\hat{\mathbf{x}}](t_2) := [\mathbf{x}](t_1)+(t_2-t_1)*[\mathbf{f}]([\mathbf{x}](t_1),[\mathbf{u}])$;
3 $\quad[\mathbf{v}] := [\mathbf{x}](t_1)\sqcup[\hat{\mathbf{x}}](t_2)$;
4 $\quad[\mathbf{w}] := \text{inflate}([\mathbf{v}],\alpha.w([\mathbf{v}])+\beta)$;
5 $\quad$**if** $[\mathbf{x}](t_1)+[0,t_2-t_1]*[\mathbf{f}]([\mathbf{w}],[\mathbf{u}]) \not\subseteq [\mathbf{w}]$ **then**
6 $\quad\quad[\mathbf{x}](t_2) := \mathbb{R}^n$
7 $\quad\quad$return
8 $\quad[\mathbf{x}](t_2) := [\mathbf{x}](t_1)+(t_2-t_1)*[\mathbf{f}]([\mathbf{w}],[\mathbf{u}])$;
9 **end**

The algorithm to we gave to compute the interval flow is very conservative. The pessimism can drastically be reduced by using the Lohner method (Lohner, 1987).

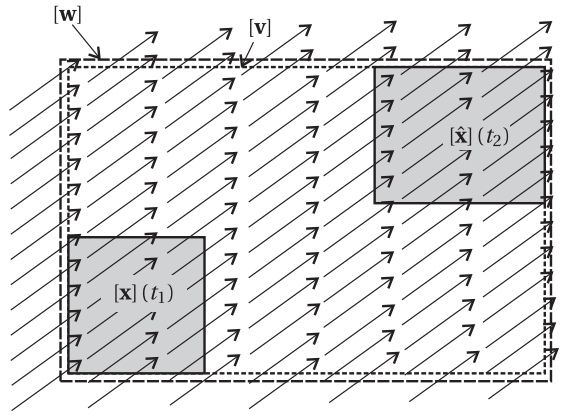


Figure 1: Principle of algorithm $[\phi]$.

3 ALGORITHM

This section presents an algorithm to compute an inner and an outer approximation of the capture basin. It is based on Theorem 2.

Theorem 2 *If* $\mathbb{C}^-$ *and* $\mathbb{C}^+$ *are such that* $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+ \subset \mathbb{K}$, *if* $[\mathbf{x}]$ *is a box and if* $\mathbf{u} \in \mathcal{F}([0,t] \to \mathbb{U})$, *then*

(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $\left(\phi(t,[\mathbf{x}],\mathbf{u}) \subset \mathbb{C}^- \wedge \phi([0,t],[\mathbf{x}],\mathbf{u}) \subset \mathbb{K}\right) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $\phi(t,[\mathbf{x}],\mathbb{U}) \cap \mathbb{C}^+ = \emptyset \wedge \phi(t,[\mathbf{x}],\mathbb{U}) \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

Proof : (*i*) and (*ii*) are due to the inclusion $\mathbb{T} \subset \mathbb{C} \subset \mathbb{K}$. Since $\mathbb{T} \subset \mathbb{C}^- \subset \mathbb{C}$, (*iii*) is a consequence of the definition of the capture basin (see (3)). The proof of (*iv*) is easily obtained by considering (3) and in view of fact that $\mathbb{C} \subset \mathbb{C}^+ \subset \mathbb{K}$.

Finally, a simple but efficient bisection algorithm is then easily constructed. It is summarized in Algorithm 2. The algorithm computes both an inner and outer approximation of the capture basin $\mathbb{C}$. In what follows, we shall assume that the set $\mathbb{U}$ of feasible input vectors is a box $[\mathbf{u}]$. The box $[\mathbf{x}]$ to be given as an input argument for ENCLOSE should contain set $\mathbb{K}$.

Comments. Steps 4 and 7 uses Theorem 2, (i)-(iii) to inflate $\mathbb{C}^-$. Steps 5 and 8 uses Theorem 2, (ii)-(iv) to deflate $\mathbb{C}^+$.

Algorithm 2: ENCLOSE.

Data: $\mathbb{K},\mathbb{T},[\mathbf{x}]$
Result: $\mathbb{C}^-,\mathbb{C}^+$
begin
1 $\quad\mathbb{C}^- \leftarrow \emptyset; \mathbb{C}^+ \leftarrow [\mathbf{x}]; \mathcal{L} \leftarrow \{[\mathbf{x}]\}$;
2 $\quad$**while** $\mathcal{L} \neq \emptyset$ **do**
3 $\quad\quad$pop the largest box $[\mathbf{x}]$ from $\mathcal{L}$;
4 $\quad\quad$**if** $[\mathbf{x}] \subset \mathbb{T}$ **then**
$\quad\quad\quad\mathbb{C}^- \leftarrow \mathbb{C}^- \cup [\mathbf{x}]$;
5 $\quad\quad$**else if** $[\mathbf{x}] \cap \mathbb{K} = \emptyset$ **then**
$\quad\quad\quad\mathbb{C}^+ \leftarrow \mathbb{C}^+ \backslash [\mathbf{x}]$;
6 $\quad\quad$take $t \geq 0$ and $\mathbf{u} \in \mathbb{U}$
7 $\quad\quad$**if** $[\phi](t,[\mathbf{x}],\mathbf{u}) \subset \mathbb{C}^-$ and $[\phi]([0,t],[\mathbf{x}],\mathbf{u}) \subset \mathbb{K}$ **then**
$\quad\quad\quad\mathbb{C}^- \leftarrow \mathbb{C}^- \cup [\mathbf{x}]$;
8 $\quad\quad$**else if** $[\phi](t,[\mathbf{x}],\mathbf{u}) \cap \mathbb{C}^+ = \emptyset$ and $[\phi](t,[\mathbf{x}],\mathbb{U}) \cap \mathbb{K} = \emptyset$ **then**
$\quad\quad\quad\mathbb{C}^+ \leftarrow \mathbb{C}^+ \backslash [\mathbf{x}]$;
9 $\quad\quad$**else if** $w([\mathbf{x}]) \geq \varepsilon$ **then**
$\quad\quad\quad$bisect $[\mathbf{x}]$ and store the two resulting
$\quad\quad\quad$boxes into $\mathcal{L}$;
end

where

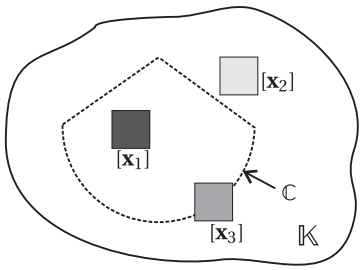
ε : ENCLOSE stops the bisecting procedure when the precision is reached ;

$\mathbb{C}^-$: Subpaving (list of nonoverlapping boxes) representing an inner approximation of the capture basin, that is the boxes inside the capture basin $\mathbb{C}$;

$\mathbb{C}^+$: Subpaving representing the outer approximation of the capture basin, that is the boxes outside $\mathbb{C}$ and the boxes for which no conclusion could be reached ;

These subpavings provide the following bracketing of the solution set :

$$\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+.$$



1. Solution $[\mathbf{x}_1] \subset \mathbb{C}^-$
2. Non solution $[\mathbf{x}_2] \subset \mathbb{C}^+$
3. Undefined $[\mathbf{x}_3] \subset \mathbb{C}^+$

Figure 2: Two dimensional exemple of ENCLOSE algorithm.

4 EXPERIMENTATIONS

This section presents an application of Algorithm 2. The algorithm has been implemented in $C++$ using Profil/BIAS interval library and executed on a PentiumM 1.4Ghz processor. As an illustration of the algorithm we consider the Zermelo problem (Bryson and Ho, 1975; Cardaliaguet et al., 1997). In control theory, Zermelo has described the problem of a boat which wants to reach an island from the bank of a river with strong currents. The magnitude and direction of the currents are known as a function of position. Let $f(x_1,x_2)$ be the water current of the river at position (x_1,x_2). The method for computing the expression of the speed vector field of two dimensional flows can be found in (Batchelor, 2000). In our example the dynamic is nonlinear,

$$f(x_1,x_2) \triangleq \left(1+\frac{x_2^2-x_1^2}{(x_1^2+x_2^2)^2}\ ,\ \frac{-2x_1x_2}{(x_1^2+x_2^2)^2}\right).$$

The speed vector field associated to the dynamic of the currents is represented on Figure 3.
Let $\mathbb{T} \triangleq \mathcal{B}(0,r)$ with $r=1$ be the island and we set $\mathbb{K} = [-8,8]\times[-4,4]$, where $\mathbb{K}$ represents the river. The boat has his own dynamic. He can sail in any direction at a speed v. Figure 4 presents the two-dimensional boat. Then, the global dynamic is given by

$$\begin{cases} x_1'(t) &= 1+\dfrac{x_2^2-x_1^2}{(x_1^2+x_2^2)^2}+v\cos(\theta) \\ x_2'(t) &= \dfrac{-2x_1x_2}{(x_1^2+x_2^2)^2}+v\sin(\theta) \end{cases},$$

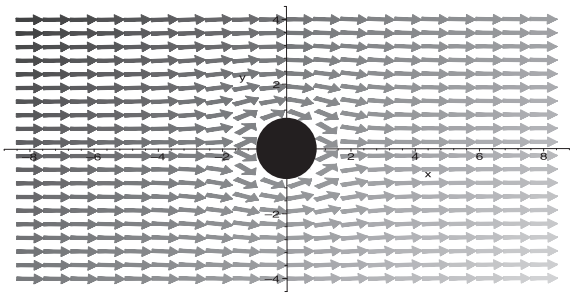


Figure 3: Vector field of the currents.

where the controls $0 \le v \le 0.8$ and $\theta \in [-\pi,\pi]$.

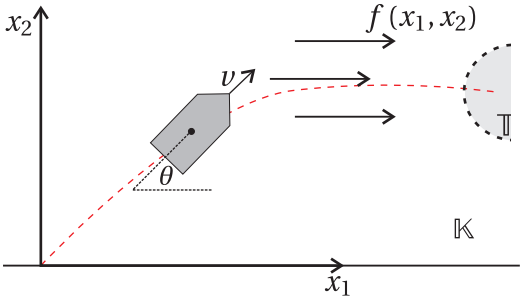


Figure 4: Zermelo's problem.

Figure 5 shows the result of the ENCLOSE algorithm, where the circle delimits the border of the target $\mathbb{T}$. Then, $\mathbb{C}^-$ corresponds to the union of all dark grey boxes and $\mathbb{C}^+$ corresponds to the union of both grey and light grey boxes. Thus, we have the following inclusion relation :

$$\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+.$$

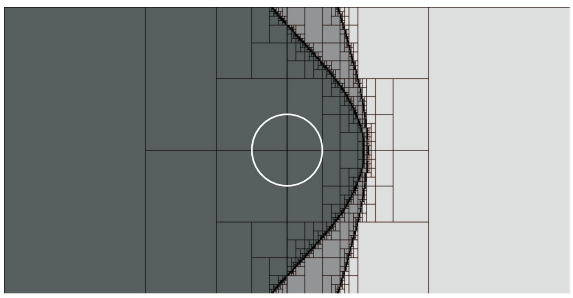


Figure 5: Two dimensional exemple of ENCLOSE algorithm.

5 CONCLUSION

In this paper, a new approach to deal with capture basin problems is presented. This approach uses interval analysis to compute an inner an outer approximation of the capture basin for a given target. To fill out this work, different perspectives appear. It could be interesting to tackle problems in significantly larger dimensions. The limitation is mainly due to the bisections involved in the interval algorithms that makes the complexity exponential with respect to the number of variables. Constraint propagation techniques (L. Jaulin, M. Kieffer, O. Didrit, E. Walter, 2001) make it possible to push back this frontier and to deal with high dimensional problems (with more than 1000 variables for instance). In the future, we plan to combine our algorithm with graph theory and guaranteed numerical integration (Nedialkov et al., 1999; Delanoue, 2006) to compute a guaranteed control **u**.

ACKNOWLEDGEMENTS

The authors wish to thank N. Delanoue for many helpful comments and valuable discussions

REFERENCES

Aubin, J. (1991). *Viability theory*. Birkhäuser, Boston.

Batchelor, G.-K. (2000). *An introduction to fluid dynamics*. Cambridge university press.

Bryson, A. E. and Ho, Y.-C. (1975). *Applied optimal control : optimization, estimation, and control*. Halsted Press.

Cardaliaguet, P., Quincampoix, M., and Saint-Pierre, P. (1997). Optimal times for constrained nonlinear control problems without local controllability. *Applied Mathematics and Optimization*, 36:21–42.

Cruck, E., Moitie, R., and Seube, N. (2001). Estimation of basins of attraction for uncertain systems with affine and lipschitz dynamics. *Dynamics and Control*, 11(3):211–227.

Delanoue, N. (2006). *Algorithmes numriques pour l'analyse topologique*. PhD dissertation, Université d'Angers, ISTIA, France. Available at `www.istia.univ-angers.fr/~delanoue/`.

Kearfott, R. B. and Kreinovich, V., editors (1996). *Applications of Interval Computations*. Kluwer, Dordrecht, the Netherlands.

L. Jaulin, M. Kieffer, O. Didrit, E. Walter (2001). *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer-Verlag, London.

Lohner, R. (1987). Enclosing the solutions of ordinary initial and boundary value problems. In Kaucher, E., Kulisch, U., and Ullrich, C., editors, *Computer Arithmetic: Scientific Computation and Programming Languages*, pages 255–286. BG Teubner, Stuttgart, Germany.

Moore, R. E. (1966). *Interval Analysis*. Prentice-Hall, Englewood Cliffs, NJ.

Nedialkov, N.-S., Jackson, K.-R., and Corliss, G.-F. (1999). Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105:21–68.

Saint-Pierre, P. (1994). Approximation of the viability kernel. *Applied Mathematics & Optimization*, 29:187–209.

BLIND TWO-THERMOCOUPLE SENSOR CHARACTERISATION

Peter C. Hung, Seán F. McLoone
Department of Electronic Engineering, National Unviersity of Ireland Maynooth, Maynooth, Co. Kildare, Ireland
phung@eeng.nuim.ie, s.mcloone@ieee.org

George W. Irwin, Robert J. Kee
Virtual Engineering Centre, Queen's University Belfast, Belfast, Northern Ireland, BT9 5HN
g.irwin@qub.ac.uk, r.kee@qub.ac.uk

Keywords: Sensor, system identification, thermocouple, blind deconvolution.

Abstract: Thermocouples are one of the most popular devices for temperature measurement in many mechatronic implementations. However, large wire diameters are required to withstand harsh environments and consequently the sensor bandwidth is reduced. This paper describes a novel algorithmic compensation technique based on blind deconvolution to address this loss of high frequency signal components using the outputs from two thermocouples. In particular, a cross-relation blind deconvolution for parameter estimation is proposed. A feature of this approach, unlike previous methods, is that no *a priori* assumption is made about the time constant ratio of the two thermocouples. The advantages, including small estimation variance, are highlighted using results from simulation studies.

1 INTRODUCTION

There is a growing trend towards the integration of different types of sensors and actuators with information processing (Isermann, 2005). Commercial and industrial applications increasingly demand dynamic temperature measurement when advanced mechatronic components are incorporated. In the automotive industry for example, accurate and reliable measurement of exhaust gas temperature is required for the regeneration of diesel particulate filters (DPF), and for the evaluation of the combustion performance of internal combustion engines (Kee and Blair, 1994).

Fast response temperature measurement can be performed using techniques such as Coherent Anti-Stokes Spectroscopy, Laser-Induced Fluorescence and Infrared Pyrometry. However, these are expensive, difficult to calibrate and maintain and are therefore impractical for wide-scale deployment outside the laboratory (Hung *et al.*, 2005a). Thermocouples are widely used for temperature measurement due to their high permissible working limit and good linear temperature dependence. In addition, their low cost, robustness, ease of installation and reliability means that there are many situations in which thermocouples are indeed the only suitable choice. Unfortunately, their design involves a compromise between robustness and speed of response which poses major problems when measuring temperature fluctuations with high frequency signal components.

To remove the effect of the sensor on the measured quantity in such conditions, compensation of the thermocouple measurement is desirable. Usually, this compensation involves two stages: thermocouple characterisation followed by temperature reconstruction. Reconstruction is a process of restoring the unknown fluid temperature from thermocouple outputs using either software techniques or hardware. This paper will concentrate on the first stage, since effective and reliable characterisation is essential for achieving satisfactory temperature reconstruction.

In an attempt to improve existing characterisation of thermocouples, this paper proposes a novel technique based on the cross-relation method (Liu *et al.*, 1993) from the field of blind deconvolution put forward by Sato (1975). Compared to other algorithms, simulations suggest

that the proposed method gives estimations with lower variance even in environments with moderate amount of noise.

This paper is organised as follows: Section 2 introduces the background of two-thermocouple characterisation. Section 3 proposes the cross-relation method and shows how it can be applied to this problem. Simulation results are presented in Section 4 while conclusions follow in Section 5.

2 DIFFERENCE EQUATION SENSOR CHARACTERISATION

2.1 Thermocouple Modelling

Assuming some criteria regarding to the construction of thermocouples are satisfied (Forney and Fralick, 1994; Tagawa and Ohta, 1997), a first-order lag model with time constant τ and unity gain can represent the frequency response of a fine-wire thermocouple (Petit, 1982). This simplified model can be written mathematically as

$$T_{\text{fluid}}(t) = T(t) + \tau \dot{T}(t) . \qquad (1)$$

Here the original fluid temperature T_{fluid} can be reconstructed if τ, the thermocouple output $T(t)$ and its derivative are available. In practice, this direct approach is infeasible as $T(t)$ contains noise and its derivative is difficult to estimate accurately. More importantly, it is generally not possible to obtain a reliable *a priori* estimate of τ, related to their thermocouple bandwidth ω_B

$$\tau = \frac{1}{\omega_B}, \qquad (2)$$

which, in turn, is a function of thermocouple wire diameter d and fluid velocity v

$$\omega_B \propto \sqrt{\frac{v}{d^3}} . \qquad (3)$$

Hence, τ varies as a function of operating conditions. Clearly, a single-thermocouple does not provide sufficient information for *in situ* estimation. Equation (3) highlights the fundamental trade-off that exists when using thermocouples. Large wire diameters are usually employed to withstand harsh environments such as engine combustion systems, but these results in thermocouples with low bandwidth, typically ω_B < 1 Hz. In these situations high frequency temperature transients are lost with the thermocouple output significantly attenuated and phase-shifted compared to T_{fluid}. Consequently, appropriate compensation of the thermocouple measurement is needed to restore the high frequency fluctuations.

2.2 Two-Thermocouple Sensor Characterisation

In 1936 Pfriem suggested using two thermocouples with different time constants to obtain *in situ* sensor characterisation. Since then, various thermocouple compensation techniques incorporating this idea have been proposed in an attempt to achieve accurate and robust fluid temperature compensation (Forney and Fralick, 1994; Tagawa and Ohta, 1997; Kee *et al.*, 1999; Hung *et al.*, 2003, 2005a, 2005b). However, the performance of all these algorithms deteriorates rapidly with increasing noise power, and many are susceptible to singularities and sensitive to offsets (Kee *et al.*, 2006). It would be very useful from the implementation point of view to know when the characterisations are not reliable.

Some of these two-thermocouple methods rely on the restrictive assumption that the ratio of the thermocouple time constants α ($\alpha < 1$ by definition) is known *a priori*. Hung *et al.* (2003, 2005a, 2005b) develop difference equation methods that do not require any *a priori* assumption about the time constant ratio.

The equivalent discrete time representation for the thermocouple model (2) is:

$$T(k) = aT(k-1) + bT_{\text{fluid}}(k-1) , \qquad (4)$$

where a and b are difference equation ARX parameters and k is the sample instant. The discrete time equivalent of α is defined as

$$\beta = b_2 / b_1 , \quad \beta < 1 . \qquad (5)$$

Here subscripts 1 and 2 are used to distinguish between signals from different thermocouples. Assuming ZOHs and a sampling interval τ_s, the parameters of the discrete and continuous time thermocouple models are related by

$$a = \exp(-\tau_s / \tau), \quad b = 1 - a \,. \tag{6}$$

Since two sets of (4) are available from each thermocouple outputs $T_1(k)$ and $T_2(k)$, a beta model (Hung, *et al.*, 2005) can be formulated by eliminating T_{fluid} from (4) to become

$$\Delta T_2^k = \beta \Delta T_1^k + b_2 \Delta T_{12}^{k-1}, \tag{7}$$

where the pseudo-sensor output ΔT_2^k and inputs ΔT_1^k and ΔT_{12}^{k-1} are defined as

$$\begin{aligned} \Delta T_1^k &= T_1(k) - T_1(k-1) \\ \Delta T_2^k &= T_2(k) - T_2(k-1) \\ \Delta T_{12}^{k-1} &= T_1(k-1) - T_2(k-1). \end{aligned} \tag{8}$$

For an *M*-sample data set (7) can be expressed in ARX vector form

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\theta}, \tag{9}$$

with $\mathbf{Y} = \Delta\mathbf{T}_2^k$, $\mathbf{X} = [\Delta\mathbf{T}_1^k \ \Delta\mathbf{T}_{12}^{k-1}]$, and $\boldsymbol{\theta} = [\beta \ b_2]^T$. Here $\Delta\mathbf{T}_1^k$, $\Delta\mathbf{T}_2^k$ and $\Delta\mathbf{T}_{12}^{k-1}$ are vectors containing *M*-1 samples of the corresponding composite signals ΔT_1^k, ΔT_2^k and ΔT_{12}^{k-1}.

Due to the form of the composite input and output signals, the noise terms in the **X** and **Y** data blocks are no longer independent. The result is that conventional least-squares and total least-squares both generate biased parameter estimates even when the measurement noise on the thermocouples is independent. It has been shown that generalised total least-squares (GTLS) on the other hand, can produce unbiased parameter estimate $\hat{\boldsymbol{\theta}}$ that outperforms other difference equation based methods. One of the reasons can be traced back to the use of β, which enhanced the model stability during parameter estimation (McLoone *et al.*, 2006).

Unfortunately, the β−GTLS approach occasionally returns unreasonable $\hat{\boldsymbol{\theta}}$ estimates as will be illustrated in Section 4. This is caused by the sensitivity of GTLS to violations in the underlying theoretical assumptions with composite signals (Huffel and Vandewalle, 1991), plus ill-conditioning of the noise correlation matrix. The blind deconvolution approach is considered here to isolate these invalid $\hat{\boldsymbol{\theta}}$.

3 BLIND SENSOR CHARACTERISATION

One of the best known deterministic blind deconvolution approaches is the method of cross-relation (CR) proposed by Liu *et al.* (1993). Such techniques exploit the information provided by output measurements from multiple systems of known structure but unknown parameters, for the same input signal.

This new approach to characterisation of thermocouples is completely different from those in Section 2. As commutation is a fundamental assumption for the method of cross-relation, the thermocouple models are both assumed to be linear. This is reasonably realistic as long as the thermocouples concerned are used within well-defined temperature ranges. Nonetheless, linearisation can easily be carried out using either the data capture hardware or software, even if the thermocouple response is nonlinear. Further, the approach requires constant model parameters, therefore the fluid flow velocity v is assumed to be constant, such that the two thermocouple time constants τ_1 and τ_2 are time-invariant.

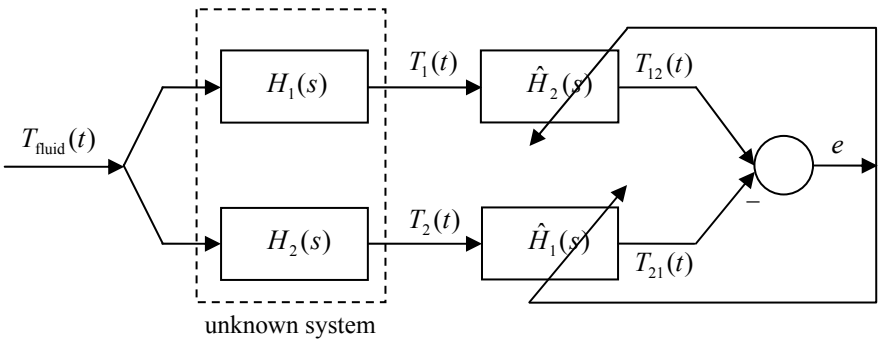


Figure 1: Two-thermocouple cross-relation characterisation.

3.1 Two-Thermocouple Sensor Characterisation

By exploiting the commutative relationship between linear systems, a novel two-themocouple characterisation scheme is proposed as follows. Since the fluid temperature T_{fluid} is unknown, the two thermocouple output signals T_1 and T_2 are passed through two different synthetic thermocouples as shown in Fig. 1. These are also modelled by (1) and can be expressed in first-order transfer function as:

$$\hat{H}_1(s) = \frac{1}{1+s\hat{\tau}_1}, \quad \hat{H}_2(s) = \frac{1}{1+s\hat{\tau}_2}, \tag{10}$$

where $\hat{H}$ is the estimate of the thermocouple transfer function H. The unknown thermocouple time constant parameters can then be estimated as $\hat{\tau}_1$ and $\hat{\tau}_2$ using the cross-relation method, illustrated in Fig. 1. Here the cross-relation error signal, $e = T_{12}(t) - T_{21}(t)$ is used to define a mean-square-error cost function

$$\begin{aligned} J_2(\hat{\tau}_1, \hat{\tau}_2) &= E\{e^2\} \\ &= E\{[T_{12}(t) - T_{21}(t)]^2\}, \forall\ \hat{\tau}_1, \hat{\tau}_2 . \end{aligned} \tag{11}$$

Equation (11) is then minimised with respect to $\hat{\tau}_1$ and $\hat{\tau}_2$ to yield the estimates of the unknown thermocouple time constants. Clearly, the cross-relation cost function $J_2(\hat{\tau}_1, \hat{\tau}_2)$ is zero when $\hat{\tau}_1 = \tau_1$ and $\hat{\tau}_2 = \tau_2$. In practice it will not be possible to obtain an exact match between T_{12} and T_{21} due to measurement noise and other factors such as thermocouple modelling inaccuracy and violations of the assumption that the two thermocouples are experiencing identical environmental conditions.

Xiu *et al.* (1995) suggest that one of the necessary conditions for multiple finite-impulse-response channels to be identifiable is that their transfer function polynomial do not share common roots. Applying this condition to the two-thermocouple characterisation problem corresponds to requiring that the time constants, and hence the diameters (3), of the thermocouples are different, that is

$$\tau_1 \neq \tau_2 \quad \Rightarrow \quad d_1 \neq d_2 . \tag{12}$$

Not surprsingly, this requirement is consistent with all other two-thermocouple characterisation techniques mentioned in Section 2. Thus, cross-relation deconvolution converts the problem of sensor characterisation into an optimisation one.

3.2 Cost Function

A 3-D surface plot and a contour map of a typical $J_2(\hat{\tau}_1, \hat{\tau}_2)$ cost function are shown in Figs. 2 and 3. Unfortunately, $J_2(\hat{\tau}_1, \hat{\tau}_2)$ is not quadratic and cannot therefore be minimised using linear least-squares. Fig. 3 shows that the cross-relation cost function is highly non-quadratic away from the minimum corresponding to the value of the true time constants.

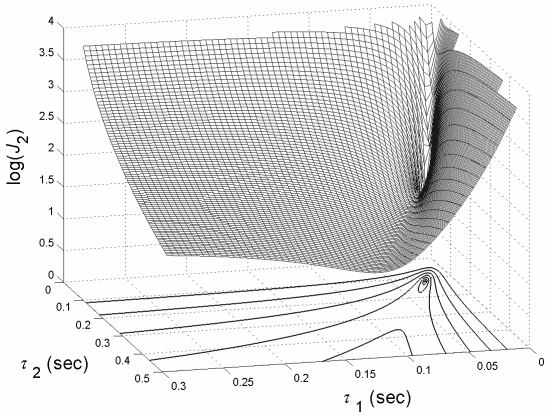


Figure 2: Three-dimensional plot of log(J_2).

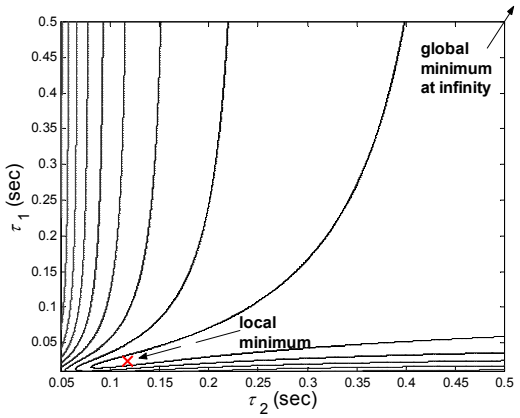


Figure 3: Contour plot of J_2 (cross: local minimum).

More importantly, the cost function has a second minimum when both time constant values approach infinity. Under these conditions, both low-pass filters (10) take infinite amounts of time to respond.

In other words, they are effectively open-circuited and their differences will always be zero. The existence of this minimum applies regardless of the noise conditions or any violations of the modelling assumptions. The minimum at infinity is thus in fact the global minimum, while the true time constant value is located at a local minimum. In the absence of noise, $J_2 = 0$ at both the global and local minima.

In addition, the narrow basin of attraction of the desired local minimum coupled with the global minimum at infinity has serious implications for optimisation complexity since search bounds have to be carefully selected to avoid divergence of gradient search algorithms to the global minimum. Consequently, in this study a robust, but inefficient, grid based search has been adopted to avoid these issues. To reduce the associated computational load different step sizes are used for each time constant. Noting from Fig. 3 that, at least locally,

$$\frac{\partial J_2}{\partial \tau_1} > \frac{\partial J_2}{\partial \tau_2}, \tag{13}$$

it can be concluded that the cost function is more sensitive to changes in the smaller thermocouple time constant; hence greater accuracy is required in estimating this value.

4 SIMULATION RESULTS

A MATLAB® simulation of a two-thermocouple probe system (Fig. 4) was used to evaluate the performance of the proposed cross-relation (CR) blind sensor characterisation. Thermocouples 1 and 2 were modelled as first-order low-pass filters according to (1) with time constants $\tau_1 = 23.8$ and $\tau_2 = 116.8$ ms respectively. The simulated fluid temperature was varied sinusoidally according to

$$T_{\text{fluid}}(t) = 16.5\sin(20\pi t) + 50.5\,, \tag{14}$$

and the resulting temperature measurements sampled every 2 ms. Each simulation ran for 5 s.

The level of zero-mean white Gaussian measurement noise added to the thermocouple signals is described by the noise level L_e, defined as

$$L_e = \frac{\text{var}(n_i)}{\text{var}(T_{\text{fluid}})} \cdot 100\%, \qquad i = 1, 2, \tag{15}$$

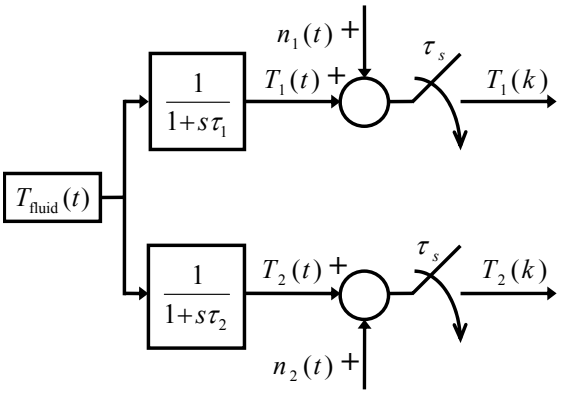


Figure 4: Simulated two-thermocouple measurement system.

where n_1 and n_2 are the noises added to the thermocouples. For a given L_e, the algorithm performance was assessed in terms of percentage estimation errors:

$$e = \frac{\tau - \hat{\tau}}{\tau} \cdot 100\%\,. \tag{16}$$

To reduce the time required for completing the simulation, the following search ranges and intervals (13) were chosen for the cross-relation (CR) algorithm:

$$\begin{aligned} &10 < \hat{\tau}_1 < 30 \text{ ms; at every } 0.5 \text{ ms,} \\ &100 < \hat{\tau}_2 < 130 \text{ ms; at every } 2.5 \text{ ms.} \end{aligned} \tag{17}$$

Of particular importance was the removal of the first 1000 data samples before computing $J_2(\hat{\tau}_1, \hat{\tau}_2)$, using the remaining 1500 sets of CR outputs T_{12} and T_{21}. This was required to eliminate the effect of transients on parameter estimation accuracy during each iteration of CR simulation (Fig. 1). The number of samples removed was estimated to exceed the 98% settling time for the system (i.e. five times the largest time constant τ_2) which equated to about 0.6 s or 300 samples.

The resulting means and standard deviations of the parameter estimation error (16), for both β–GTLS (Section 2.2) and CR (Section 3.1) algorithms are shown in Fig. 5. Note results for $\hat{\tau}_2$ are similar to those illustrated for $\hat{\tau}_1$ and are thus omitted.

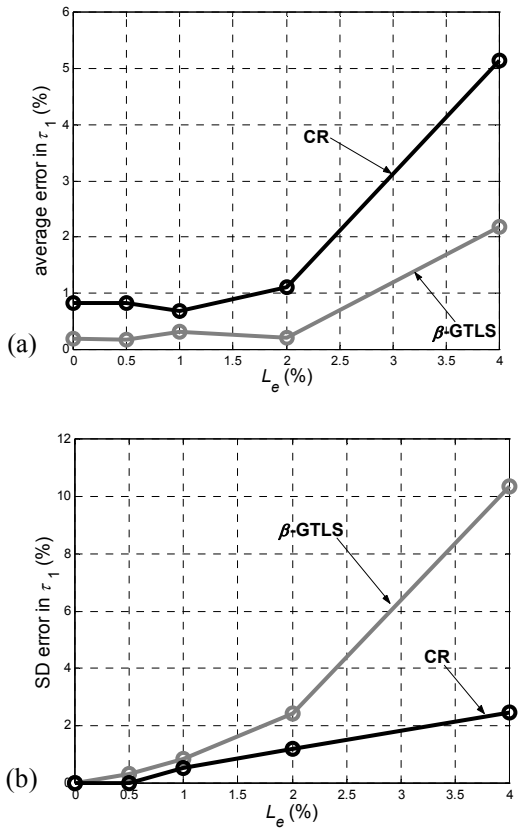


Figure 5: (a) Means and (b) standard deviations of e of $\hat{\tau}_1$ averaged over 100 Monte-Carlo runs.

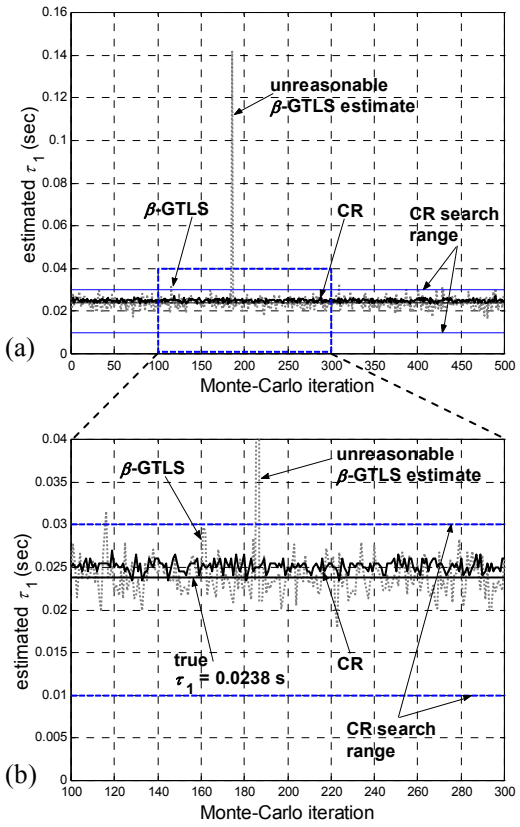


Figure 6: 500 Monte-Carlo runs of $\hat{\tau}_1$ of $\beta-$GTLS and CR, where (b) is a magnified version of (a).

These results suggset that CR produces biased parameter estimates since their expected mean errors are greater than that of $\beta-$GTLS . However, the estimation standard deviations of CR are less than that of $\beta-$GTLS .

With regard to the search intervals taken for CR, two issues need to be considered when looking at the graphs. Firstly, a major contribution to the CR bias comes from the low resolution of the search grid used. Since, when $\tau_1 = 23.8$ ms, an interval of 0.5 ms represents an 'artificial' estimation bias of up to 2.1%. This can be reduced if a finer search grid is employed, at the expense of increasing the already heavy computation load. Similarly, the CR standard deviation errors may be 2.1% larger than the reported values because of the finite resolution employed, although this is unlikely due to the intrinsic noise-filtering capability of CR.

The noise-resilient property of CR compared to GTLS is further highlighted in Fig. 6, where 500 Monte-Carlo simulations were performed. It can be seen that one unreasonable $\hat{\tau}_1$ value was returned by $\beta-$GTLS while the CR approach is well-behaved, although its estimate is asymptotically biased. Hence, CR can be used to verify whether a GTLS estimate is genuine or corrupted by signal outliers, improving the overall reliability of sensor characterisation.

5 CONCLUSIONS

A novel cross-relation (CR) sensor characterisation method has been presented. It does not require *a priori* knowledge of the thermocouple time constant ratio α , as required in many other characterisation algorithms. CR is more noise-tolerant in the sense of reduced parameter estimation variance when compared to the alternatives such as $\beta-$GTLS . The robustness arises because the CR process involves passing each thermocouple output through a first-order block, which removes, at least partially,

measurement noise during identification. As a result, CR can be employed to verify estimation validity, thereby increasing the overall reliability of other characterisation methods.

The computational complexity of CR, due to the inefficient grid based search used in this study, means that it is most appropriate for offline sensor characterisation. Further investigations include ways to speed up the computation and reduce the estimation bias.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the financial support of the Virtual Engineering Centre, Queen's University Belfast, *http://www.vec.qub.ac.uk*.

REFERENCES

Forney, L. J., Fralick G. C., 1994. Two wire thermocouple: Frequency response in constant flow. *Rev. Sci. Instrum.*, 65, pp 3252-3257.

Hung, P., McLoone, S., Irwin G., Kee, R., 2003. A Total Least Squares Approach to Sensor Characterisations. *Proc. 13th IFAC Symposium on Sys. Id.*, Rotterdam, The Netherlands, pp 337-342.

Hung, P. C., McLoone, S., Irwin G., Kee, R., 2005a. A difference equation approach to two-thermocouple sensor characterisation in constant velocity flow environments. *Rev. Sci. Instrum.*, 76, Paper No. 024902.

Hung, P. C., McLoone, S., Irwin G., Kee, R., 2005b. Unbiased thermocouple sensor characterisation in variable flow environments. *Proc. 16th IFAC World Congress*, Prague, Czech Republic.

Isermann, R., 2005. Mechatronic Systems – Innovative Products with Embedded Control. *Proc. 16th IFAC World Congress*, Prague, Czech Republic.

Kee, R. J., Blair, G. P., 1994. Acceleration test method for a high performance two-stroke racing engine. *Proc. SAE Motorsports Conference*, Detroit, MI, Paper No. 942478.

Kee, R. J, O'Reilly, P. G., Fleck, R., McEntee, P. T., 1999. Measurement of Exhaust Gas Temperature in a High Performance Two-Stroke Engine. *SAE Trans. J. Engines*, 107, Paper No. 983072.

Kee, J. K., Hung, P., Fleck, B., Irwin, G., Kenny, R., Gaynor, J., McLoone, S., 2006. Fast response exhaust gas temperature measurement in IC Engines. *SAE 2006 World Congress*, Detroit, MI, Paper No. 2006-01-1319.

Liu, H., Xu, G., Tong, L., 1993. A deterministic approach to blind identification of multichannel FIR systems. *Proc. 27th Asilomar Conference on Signals, Systems and Computers*, Asilomar, CA, pp. 581-584.

McLoone, S., Hung, P., Irwin, G., Kee, R., 2006. Exploiting *A Priori* Time Constant Ratio Information in Difference Equation Two-Thermocouple Sensor Characterisation. *IEEE Sensors J.*, 6, pp. 1627-1637.

Pfriem, H., 1936. Zue messung verandelisher temperaturen von ogasen und flussigkeiten. *Forsch. Geb. Ingenieurwes*, 7, pp. 85-92.

Petit, C., Gajan, P., Lecordier, J. C., Paranthoen, P., 1982. Frequency response of fine wire thermocouple. *J. Physics Part E*, 15, pp. 760-764.

Sato, Y., 1975. A method of self-recovering equalization for multilevel amplitude modulation systems. *IEEE Trans. in Communications*, 23, pp. 679-682.

Tagawa, M., Ohta, Y., 1997. Two-Thermocouple Probe for Fluctuating Temperature Measurement in Combustion – Rational Estimation of Mean and Fluctuating Time Constants. *Combustion and Flame*, 109, pp 549-560.

Xu, G., Liu, H., Tong, L., Kailath, T., 1995. A least-squares approach to blind channel identification. *IEEE Trans. on Signal Processing*, 43, pp. 2982-2993.

Van Huffel S., Vandewalle, J., 1991. The Total Least Squares Problem: Computational Aspects and Analysis, SIAM, Philadelphia, 1st edition.

CONJUGATE GRADIENT TECHNIQUES FOR MULTICHANNEL ADAPTIVE FILTERING

Lino García Morales and Fernando Juan Berenguer Císcar
Escuela Superior Politécnica,Universidad Europea de Madrid, Tajo S/N, Villaviciosa de Odón, Madrid, Spain
lino.garcia@uem.es, fjuan.berenguer@uem.es

Keywords: Multichannel Adaptive Filtering, System Identification, Optimization Method, Conjugate Gradient, Partitioned Frequency-Domain Adaptive Filtering.

Abstract: The conjugate gradient is the most popular optimization method for solving large systems of linear equations. In a system identification problem, for example, where very large impulse response is involved, it is necessary to apply a particular strategy which diminishes the delay, while improving the convergence time. In this paper we propose a new scheme which combines frequency-domain adaptive filtering with a conjugate gradient technique in order to solve a high order multichannel adaptive filter, while being delayless and guaranteeing a very short convergence time.

1 INTRODUCTION

The multichannel adaptive filtering problem's solution depends on the correlation between the channels, the number of channels and the order and nature of the impulse responses involved in the system. The multichannel acoustic echo cancellation (MAEC) application, for example, can be seen as a system identification problem with extremely large impulse responses (depending on the environment and its reverberation time, the echo paths can be characterized by FIR filters with thousands of taps).

In these cases a multirate adaptive scheme such a partitioned block frequency-domain adaptive filter (PBFDAF) (Páez and García, 1992) is a good alternative and is widely used in commercial systems nowadays. However, the convergence speed may not be fast enough under certain circumstances.

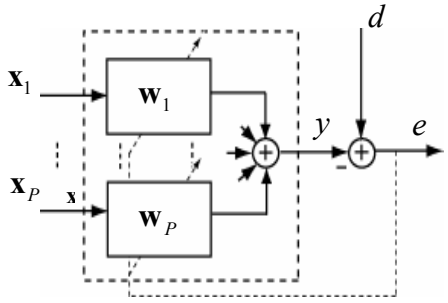


Figure 1: Multichannel Adaptive Filtering.

Figure 1 shows the working framework, where $\mathbf{x}_p$ represents the p channel input signal, d the desired signal, y the output of adaptive filter and e the error signal we try to minimize. In typical scenarios, the filter input signals $\mathbf{x}_p$, $p = 1,\ldots,P$ (where P is a number of channels), are highly correlated which further reduces the overall convergence of the adaptive filter coefficients w_{pm}, $m = 1,\ldots,L$ (L is the filter length),

$$y[n] = \sum_{p=1}^{P}\sum_{m=1}^{L} x_p[n-m]\,w_{pm} . \qquad (1)$$

The mean square error (MSE) minimization of the multichannel signal with respect to the filter coefficients is equivalent to the Wiener-Hopf equation

$$\mathbf{R}\mathbf{w} = \mathbf{r} . \qquad (2)$$

$\mathbf{R}$ represents the autocorrelation matrix and $\mathbf{r}$ the cross-correlation vector between the input and the desired signals. Both are a priori time-domain statistical unknown variables, although can be estimated iteratively from $\mathbf{x}$ and d.

$\mathbf{R} = E\{\mathbf{x}\mathbf{x}^H\}$ and $\mathbf{r} = E\{\mathbf{x}d^*\}$, with $\mathbf{x} = \begin{bmatrix} \mathbf{x}_1^T & \dots & \mathbf{x}_P^T \end{bmatrix}^T$; $\mathbf{w} = \begin{bmatrix} \mathbf{w}_1^T & \dots & \mathbf{w}_P^T \end{bmatrix}^T$ and $\mathbf{w}_p = \begin{bmatrix} w_{p1} & \dots & w_{pL} \end{bmatrix}^T$. In the notation we are using a for scalar, $\mathbf{a}$ for vector and $\mathbf{A}$ for matrix; $\mathbf{a}$, $\mathbf{A}$ denotes vector and matrix respectively in a frequency-domain: $\mathbf{a} = \mathbf{F}\mathbf{a}$, $\mathbf{A} = \mathbf{F}\mathbf{A}$. $\mathbf{F}$ represents the discrete Fourier transform (DFT) matrix defined as $\mathbf{F}_{kl} = e^{-j2\pi kl/M}$, with $k,l = 0,\dots,M-1$, $j = \sqrt{-1}$ and $\mathbf{F}^{-1}$ as its inverse. Of course, in the final implementation, the DFT matrix is substituted by much more efficient fast Fourier transforms (FFT). Here $(.)^T$ denotes transpose operator and $(.)^H = \left((.)^T\right)^*$ the Hermitian operator (conjugate transpose).

The conjugate gradient (CG) method is efficient to obtain the solution to (2), however, a big delay is introduced (noted that the system order is $LP \times LP$). In order to reduce this convergence speed problem we propose a new algorithm which employs much more powerful CG optimization techniques, but keeping the frequency block partition strategy to allow computationally realistic low latency situations. The paper is organized as follows: Section 2 reviews the Multichannel PBFDAF approach and its implementation. Section 3 develops the Multichannel Conjugate Gradient Partitioned Frequency Domain Adaptive Filter algorithm (PBFDAF–CG). Results of the new approach are presented in Section 4 and 5 followed by conclusions.

2 PBFDAF

The PBFDAF was developed to deal efficiently with such situations. The PBFDAF is a more efficient implementation of Least Mean Square (LMS) algorithm in the frequency-domain. It reduces the computational burden and user-delay bounded. In general, the PBFDAF is widely used due to be good trade-off between speed, computational complexity and overall latency. However, when working with long impulse response, as the acoustic impulse responses (AIR) used in MAEC, the convergence properties provided by the algorithm may not be enough. Besides, the multichannel adaptive filter is structurally more difficult, in general, than the single channel case (Benesty and Huang, 2003).

This technique makes a sequential partition of the impulse response in the time-domain prior to a frequency-domain implementation of the filtering operation. This time segmentation allows setting up individual coefficient updating strategies concerning different sections of the adaptive canceller, thus avoiding the need for disabling the adaptation in the complete filter. The adaptive algorithm is based on the frequency-domain adaptive filter (FDAF) for every section of the filter (Shink, 1992).

The main idea of frequency-domain adaptive filter is to frequency transform the input signal in order to work with matrix multiplications instead of dealing with slow convolutions. The frequency-domain transform employs one or more DFTs and can be seen as a pre-processing block that generates decorrelated output signals.

In the more general FDAF case, the output of the filter in the time domain (1) can be seen as a direct frequency-domain translation of the block LMS (BLMS) algorithm. In the PBFDAF case, the filter is partitioned transversally in an equivalent structure. Partitioning $\mathbf{w}_p$ in Q segments (K length) we obtain

$$y[n] = \sum_{p=1}^{P}\sum_{q=1}^{Q}\sum_{m=0}^{K-1} x_p[n-qK-m]\, w_{p(qK+m)} \qquad (3)$$

Where the total filter length L, for each channel, is a multiple of the length of each segment $L = QK$, $K \le L$. Thus, using the appropriate data sectioning procedure, the Q linear convolutions (per channel) of the filter can be independently carried out in the frequency-domain with a total delay of K samples instead of the QK samples needed in standard FDAF implementations.

Figure 2 shows the block diagram of the algorithm using the overlap-save method. In the frequency domain with matrix notation, equation (3) can be expressed as

$$\mathbf{Y} = \mathbf{X} \otimes \mathbf{W}. \qquad (4)$$

Where $\mathbf{X} = \mathbf{F}\mathbf{X}$ represents a matrix of dimensions $M \times Q \times P$ which contains the Fourier transform of the Q partitions and P channels of the input signal matrix $\mathbf{X}$.

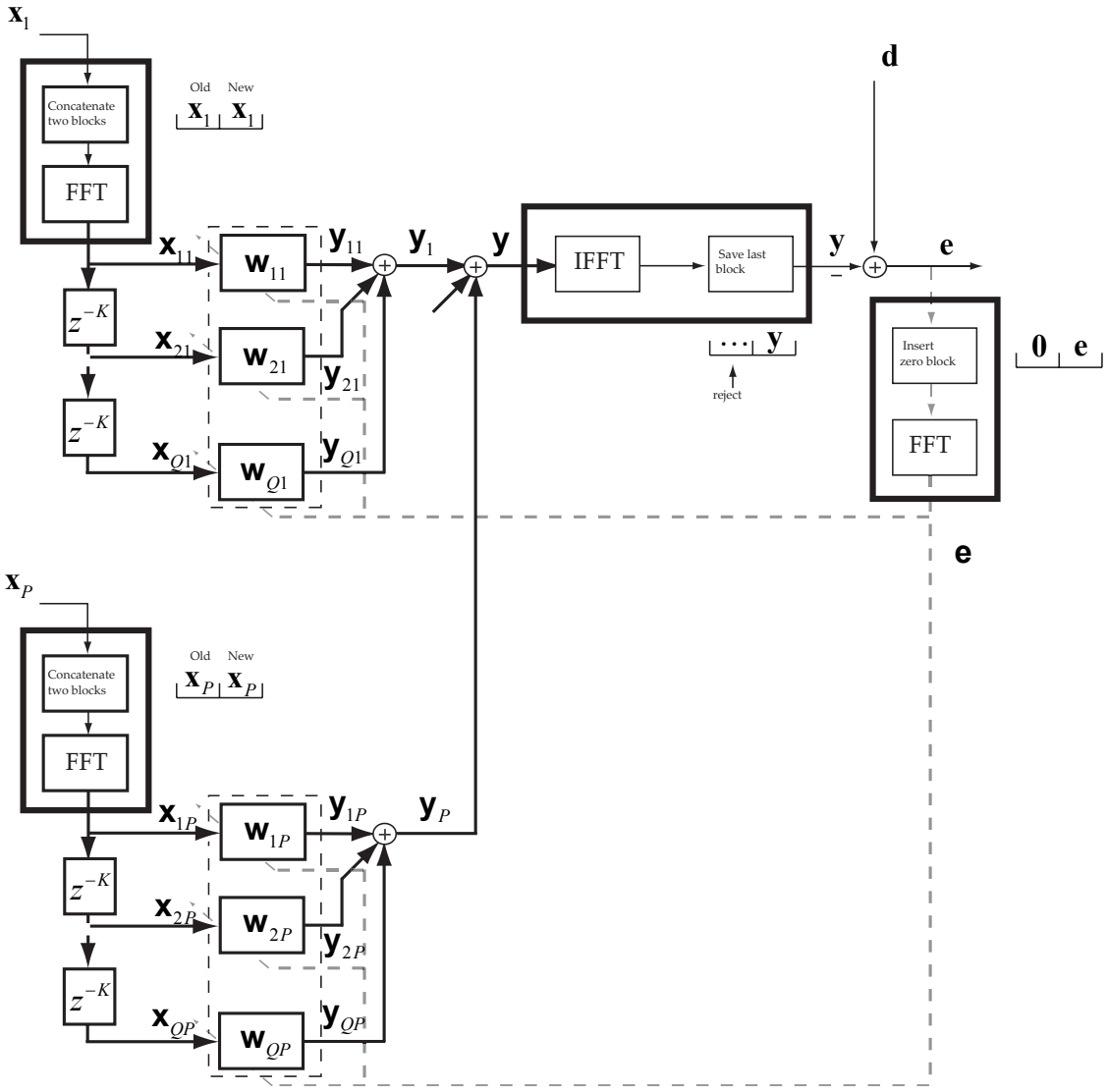


Figure 2: Multichannel PBFDAF (Overlap-Save method).

Being $\mathbf{X}$, $2K \times P$-dimensional (supposing 50% overlapping between the new block and the previous one).

It should be taken into account that the algorithm adapts every K samples. $\mathbf{W}$ represents the filter coefficient matrix adapted in the frequency-domain (also $M \times Q \times P$-dimensional) while the $\otimes$ operator multiplies each of the elements one by one; which in (4) represents a circular convolution.

The output vector $\mathbf{y}$ can be obtained as the double sum (rows) of the $\mathbf{Y}$ matrix. First we obtain a $M \times P$ matrix which contains the output of each channel in the frequency-domain $\mathbf{y}_p$, $p = 1, \ldots, P$, and secondly, adding all the outputs we obtain the output of the whole system $\mathbf{y}$. Finally, the output in the time-domain is obtained by using

$$\mathbf{y} = \text{last } K \text{ components of } \mathbf{F}^{-1}\mathbf{y}. \qquad (5)$$

Notice that the sums are performed prior to the time-domain translation. In this way we reduce $(P-1)(Q-1)$ FFTs in the complete filtering process. As in any adaptive system the error can be obtained as

$$\mathbf{e} = \mathbf{d} - \mathbf{y}, \qquad (6)$$
$$\mathbf{d} = \left[d[mK] \quad \ldots \quad d[(m+1)K - 1] \right]^T.$$

The error in the frequency-domain (for the actualization of the filter coefficients) can be obtained as

$$\mathbf{e} = \mathbf{F}\begin{bmatrix} \mathbf{0}_{K\times 1} \\ \mathbf{e} \end{bmatrix}. \quad (7)$$

As we can see, a block of K zeros is added to ensure a correct linear convolution implementation. In the same way, for the block gradient estimation, it is necessary to employ the same error vector in the frequency-domain for each partition q and channel p.

This can be achieved by generating an error matrix $\mathbf{E}$ with dimensions $M \times Q \times P$ which contains replicas of the error vector, defined in (7), of dimensions P and Q ($\mathbf{E} \leftarrow \mathbf{e}$ in the notation). The actualization of the weights is performed as

$$\mathbf{W}[m+1] = \mathbf{W}[m] + 2\mu[m] \otimes \mathbf{G}[m]. \quad (8)$$

The instantaneous gradient is estimated as

$$\mathbf{G} = -\mathbf{X}^{*} \otimes \mathbf{E}. \quad (9)$$

This is the unconstrained version of the algorithm which saves two FFTs from the computational burden at the cost of decreasing the convergence speed. As we are trying to improve specifically this parameter we have implemented the constrained version which basically makes a gradient projection. The gradient matrix is transformed into the time-domain and is transformed back into the frequency-domain using only the first K elements of $\mathbf{G}$ as

$$\mathbf{G} = \mathbf{F}\begin{bmatrix} \mathbf{G} \\ \mathbf{0}_{K\times Q\times P} \end{bmatrix}. \quad (10)$$

3 PBFDAF–CG

CG algorithm is a technique originally developed to minimize quadratic functions, as (2), which was later adapted for the general case (Luenberger, 1984). Its main advantage is its speed as it converges in a finite number of steps. In the first iteration it starts estimating the gradient, as in the steepest descent (SD) method, and from there it builds successive directions that create a set of mutually conjugate vectors with respect to the positively defined Hessian (in our case, the auto-correlation matrix $\mathbf{R}$ in the frequency-domain).

In each *m*-block iteration the conjugate gradient algorithm will iterate $k = 1, \ldots, \min(N, K)$ times; where N represent the memory of the gradient estimation, $N \le K$. In a practical system the algorithm is stopped when it reaches a user-determined MSE level. To apply this conjugate gradient approach to the PBFDAF algorithm the weight actualization equation (8) must be modified as

$$\mathbf{w}[m+1] = \mathbf{w}[m] + \alpha \mathbf{v}[m]. \quad (11)$$

Where $\mathbf{w}$ is the coefficient vector of dimension $MQP \times 1$ which results from rearranging matrix $\mathbf{W}$ (in the notation $\mathbf{w} \leftarrow \mathbf{W}$). $\mathbf{v}$ is a finite $\mathbf{R}$-conjugated vector set which satisfies $\mathbf{v}_i^H \mathbf{R} \mathbf{v}_j = 0, \forall i \neq j$. The $\mathbf{R}$-conjugacy property is useful as the linear independency of the conjugate vector set allows expanding the $\mathbf{w}^{\bullet}$ solution as

$$\mathbf{w}^{\bullet} = \alpha_0 \mathbf{v}_0 + \cdots + \alpha_k \mathbf{v}_k = \sum_{k=0}^{K-1} \alpha_k \mathbf{v}_k. \quad (12)$$

Starting at any point $\mathbf{w}_0$ of the weighting space, we can define $\mathbf{v}_0 = -\mathbf{g}_0$ being $\mathbf{g}_0 \leftarrow \bar{\mathbf{G}}_0$, $\bar{\mathbf{G}}_0 = \nabla(\mathbf{W}_0)$, $\mathbf{p}_0 \leftarrow \bar{\mathbf{P}}_0$, $\bar{\mathbf{P}}_0 = \nabla(\mathbf{W}_0 - \bar{\mathbf{G}}_0)$.

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{v}_k \quad (13)$$

$$\alpha_k = \frac{\mathbf{g}_k^H \mathbf{v}_k}{\mathbf{v}_k^H (\mathbf{g}_k - \mathbf{p}_k)} \quad (14)$$

$$\begin{aligned} \mathbf{g}_{k+1} \leftarrow \bar{\mathbf{G}}_{k+1},\ \bar{\mathbf{G}}_{k+1} = \nabla(\mathbf{W}_{k+1}) \\ \mathbf{p}_{k+1} \leftarrow \bar{\mathbf{P}}_{k+1},\ \bar{\mathbf{P}}_{k+1} = \nabla(\mathbf{W}_{k+1} - \bar{\mathbf{G}}_{k+1}) \end{aligned} \quad (15)$$

$$\mathbf{v}_{k+1} = -\mathbf{g}_{k+1} + \beta_k \mathbf{v}_k \quad (16)$$

$$\beta_k^{HS} = \frac{\mathbf{g}_{k+1}^H (\mathbf{g}_{k+1} - \mathbf{g}_k)}{\mathbf{v}_k^H (\mathbf{g}_{k+1} - \mathbf{g}_k)} \quad (17)$$

Where $\mathbf{p}_k$ represents the gradient estimated in $\mathbf{w}_k - \mathbf{g}_k$. For that, it is necessary to evaluate $\mathbf{Y} = \mathbf{X} \otimes \left(\mathbf{W} - \bar{\mathbf{G}} \right)$, (5), (6), (7) and (9). In order to be able to generate nonzero direction vectors which are conjugate to the initial negative gradient vector, a gradient estimation is necessary (Boray and Srinath, 1992). This gradient estimation is obtained by averaging the instantaneous gradient estimates over N past values. The ∇ operator is an averaging gradient estimation with the current weights and N past inputs $\mathbf{X}$ and $\mathbf{d}$,

$$\bar{\mathbf{G}}_k = \nabla \left(\mathbf{W}_k \right) = \frac{2}{N} \sum_{n=0}^{N-1} \mathbf{G}_{k-n} \Big|_{\mathbf{W}_k, \mathbf{X}_{k-n}, \mathbf{d}_{k-n}} . \quad (18)$$

This alternative approach does not require knowing neither the Hessian nor the employment of a linear search. Notice that all the operations (13-17) are vector operations that keep the computational complexity low. The equation (17) is known as the Hestenes-Stiefel method but there are different approaches for calculating β_k: Fletcher-Reeves (19), Polar-Ribière (20) and Dai-Yuan (21) methods.

$$\beta_k^{FR} = \frac{\mathbf{g}_{k+1}^H \mathbf{g}_{k+1}}{\mathbf{g}_k^H \mathbf{g}_k} \quad (19)$$

$$\beta_k^{PR} = \frac{\mathbf{g}_{k+1}^H \left(\mathbf{g}_{k+1} - \mathbf{g}_k \right)}{\mathbf{g}_k^H \mathbf{g}_k} \quad (20)$$

$$\beta_k^{DY} = \frac{\mathbf{g}_{k+1}^H \mathbf{g}_{k+1}}{\mathbf{v}_k^H \left(\mathbf{g}_{k+1} - \mathbf{g}_k \right)} \quad (21)$$

The constant β_k is chosen to provide $\mathbf{R}$-conjugacy for the vector $\mathbf{v}_k$ with respect to the previous direction vectors $\mathbf{v}_{k-1}, \ldots, \mathbf{v}_0$. Instability occurs whenever β_k exceeds unity.

In this approach, the successive directions are not guaranteed to be conjugate to each other, even when one uses the exact value of the gradient at each iteration. To ensure the algorithm stability the gradient can be initialized forcing $\beta_k = 1$ in (16) when $\beta_k > 1$.

4 COMPUTATIONAL COST

Table 1 shows a comparative analysis for both algorithms in terms of operations number (multiplications, sums) clustered by functionality. Note that constants A, B and C, in the PBFDAF computational burden estimation, are used as reference for the number of operations in PBFDAF–CG. For one iteration (k = 1), the computational cost of the PBFDAF–CG is 40 times higher than the PBFDAF.

5 SIMULATION EXAMPLES

MAEC application is a good example of complex system identification because has to deal with very long adaptive filters in order to achieve good results. The scenario employed in our tests simulates two small chambers imitating a typical teleconference environment. Following an acoustic opening approach, both chambers can be acoustically connected by means of linear arrays of microphones and loudspeakers. Details of this configuration follow. Room dimensions are [2000 2440 2700] mm.

The impulse responses are calculated using the image method (Allen and Berkley, 1979) with an expected reverberation time of 70ms (reflection coefficients [0.8 0.8; 0.5 0.5; 0.6 0.6]). The speech source, microphones and loudspeakers are situated as in Figure 3. In the emitting room, the source is located in [1000 500 1000] and the microphones in [{800 900 1000 1100 1200} 2000 750]. Notice that the microphone separation is only 10 cm, which would be a worse case scenario that provides with highly correlated signals. In the reception room the loudspeakers are situated in [{500 750 1000 1250 1500} 100 750] and the microphone in [1000 2000 750].

The directivity patterns of the loudspeakers ([elevation 0º, azimuth -90º, aperture beam 180º]) and the microphones ([0º 90º 180º]) are modified so that they are face to face. We are considering $P = 5$ channels as it is a realistic situation for home applications; enough for obtaining good spatial localization and significantly more complex than the stereo case.

Table 1: Computational Cost Comparative ($O = PQM$).

Alg.\Op.	Gradient Estimation and Convolution	Updating	Constrained Version
PBFDAF	$A = (P+2)O\log_2 O + P(Q(M+1)+1) + K + O$	$B = 9O$	$C = 2O\log_2 O$
PBFDAF–CG	$(((N(A+1)+1)+1)2+1)(k+1)$	$(13O+2)k$	$2CN(k+1)$

The source is a male speech recorded in an anechoic chamber at a sampling rate of 16 kHz and the background noise in the local room has a power of -40 dB of SNR.

Figure 4 shows the constrained PBFDAF algorithm behaviour. For equation (8) we are using a power normalizing expression as

$$\mu[m] = \frac{\mu}{\mathbf{U}[m] + \delta}, \qquad (22)$$

$$\mathbf{U}[m] = (1-\lambda)\mathbf{U}[m-1] + \lambda|\mathbf{X}|^2. \qquad (23)$$

Where $\mu[m]$ is a matrix of dimensions $M \times Q \times P$, μ is the step size, λ is an averaging factor, and δ is a constant to avoid stability problems. In our case $\mu = 0.025$, $\lambda = 0.25$ and $\delta = 0.5$.

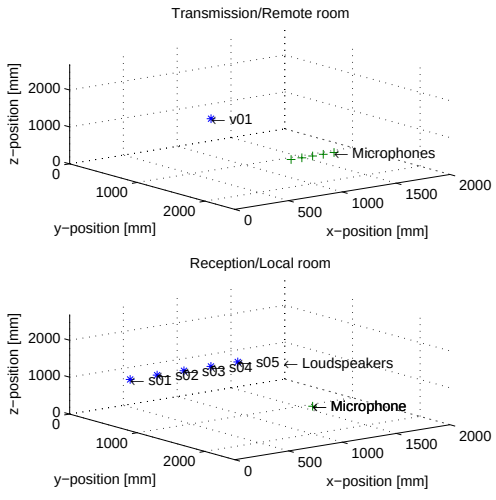


Figure 3: Working environment for the tests.

Figure 5 shows the result of using the PBFDAF-CG algorithm with the Hestenes-Stiefel method where the difference in convergence can be observed. A maximum of $N = \lfloor \sqrt{K} \rfloor$ or when MSE below -45 dB is employed.

For both algorithms we use $Q = 8$ partitions, $L = 1024$ taps, $K = L/Q = 128$ taps for each partition. The length of the FFTs is $M = 2K = 256$. Working with sample rate of 16 kHz means 8 ms of latency (although a delayless approach already has been studied) (Bendel and Burshtein, 2001). Again in both cases the algorithm uses the overlap-save method (50% overlapping).

The upper part of the figures show the echo signal d (black) and the residual error e (grey).

The centre shows the MSE (dB) and the lower picture the misalignment (also in dB) obtained as $\varepsilon = \|\mathbf{h} - \mathbf{w}\| / \|\mathbf{h}\|$, being $\mathbf{h}$ the unknown impulse response and $\mathbf{w} = \left[\mathbf{w}_1^T \ \ldots \ \mathbf{w}_P^T\right]^T$ the estimation.

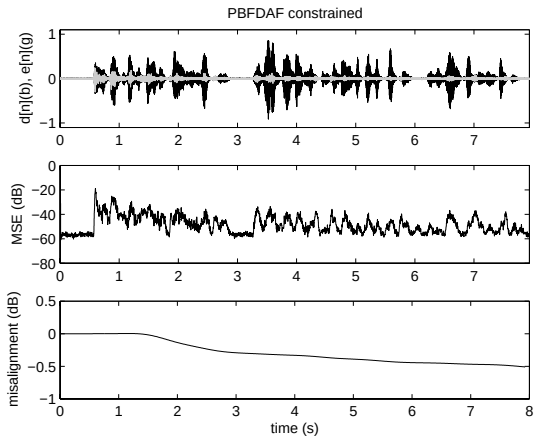


Figure 4: PBFDAF Constrained.

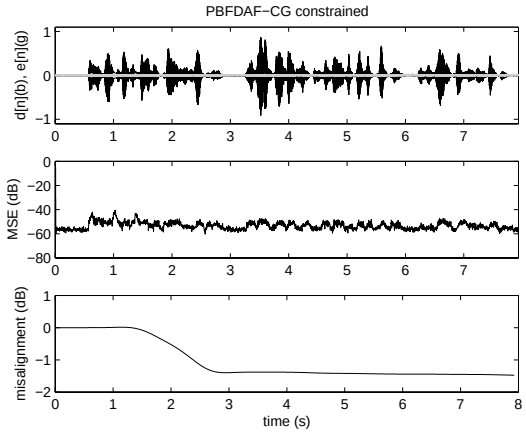


Figure 5: PBFDAF–CG Constrained.

The speech input signal to MAEC application is an inappropriate perturbation signal due to a nonstationary character. The speech waveform contains segments of voiced (quasi-periodic) sounds, such as "e," unvoiced or fricative (noiselike) sounds, such as "g," and silence.

Besides it is possible a double-talk situations (when the speech of the at least two talkers arrives simultaneously at the canceller) that made identification much more problematic than it might appear at first glance.

A much more conditioned application is an adaptive multichannel measure of impulse response. In this case, it is possible to select the best perturbation signal, with the appropriate SNR, for system identification and adapt until the error signal falls below a MSE setting threshold.

The maximum length sequences (MLS) are pseudorandom binary signals which autocorrelation function is approximately an impulse.

In an industrial case it is probably the most convenient method to use because it is simple and allows system identification without perturbing the system operation or stopping the plant (Aguado and Martínez, 2003). In this case it is necessary superimpose the perturbation signal to the input system with a power enough to identify the system while guaranty the optimal functioning.

6 CONCLUSIONS

The PBFDAF algorithm is widely used in multichannel adaptive filtering applications such as MAEC commercial systems with good results (in general for stereo case).

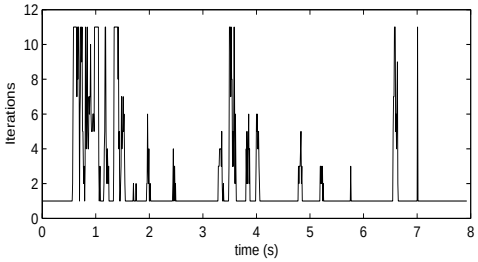


Figure 6: PBFDAF–CG iterations versus time.

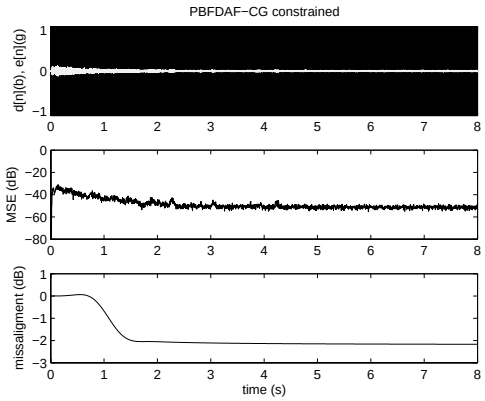


Figure 7: PBFDAF–CG Constrained (MLS).

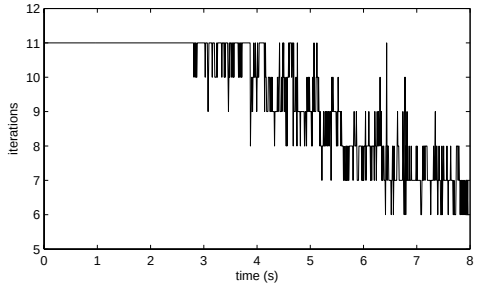


Figure 8: PBFDAF–CG iterations versus time (MLS).

However, especially when working in multichannel, high reverberation environments (like teleconference) its convergence may not be fast enough. In this article we have presented a novel algorithm: PFDAF–CG; based on the same structure, but using much more powerful CG techniques to speed up the convergence time and improve the MSE and misalignment performance.

As shown in the results, the proposed algorithm improves a MSE and misalignment performance, and converges a lot faster than its counterpart while keeping the computational convergence relatively low, because all the operations are performed between vectors in the frequency-domain. We are working on better gradient estimation methods in order to reduce computational cost. Besides, it is possible to arrive to a compromise between

complexity and speed modifying the maximum number of iterations.

Figure 6 shows the PBFDAF–CG iterations versus time. The total number of iterations for this experiment is 992 for PBFDAF and 1927 for PBFDAF–CG (80 times higher computational cost).

Figure 7 shows the result of PBFDAF–CG with MLS source (identical settings) and Figure 8 the iterations versus time. Notice that more uniform MSE convergence and best misalignment. The computational cost decrease while time the increases. A better performance is possible increasing the SNR and diminishing the MSE level threshold.

REFERENCES

Aguado, A., Martínez, M., 2003. *Identificación y Control Adaptativo*, Prentice Hall.

Allen, J.B., Berkley, D.A., 1979. Image method for efficiently simulating small-room acoustics. In *J.A.S.A.*, 65:943-950.

Bendel, Y., Burshtein, D., 2001. Delayless Frequency Domain Acoustic Echo Cancelation. In *IEEE Transactions on Speech and Audio Processing*. 9(5):589-587.

Benesty, J., Huang, Y. (Eds.), 2003. *Adaptive Signal Processing: Applications to Real-World Problems*, Springer.

Boray, G., Srinath, M.D., 1992. Conjugate Gradient Techniques for Adaptive Filtering. In *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Application*. 39(1):1-10.

Luenberger, D.G., 1984. *Introduction to Linear and Nonlinear Programming*, MA: Addison-Wesley, Reading, Mass.

Shink, J., 1992. Frequency-Domain and Multirate Adaptive Filtering. In *IEEE Signal Processing Magazine*. 9(1):15-37.

Páez Borrallo, J., García Otero, M., 1992. On the implementation of a partitioned block frequency-domain adaptive filter (PBFDAF) for long acoustic echo cancellation. In *Signal Processing*. 27:301-315.

APPENDIX

The "conjugacy" relation $\mathbf{v}_i^H \mathbf{R} \mathbf{v}_j = 0, \forall i \neq j$ means that two vectors, $\mathbf{v}_i$ and $\mathbf{v}_j$, are orthogonal with respect to any symmetric positive matrix $\mathbf{R}$. This can be looked upon as a generalization of the orthogonality, for which $\mathbf{R}$ is the unity matrix. The best way to visualize the working of conjugate directions is by comparing the space we are working in with a "stretched" space.

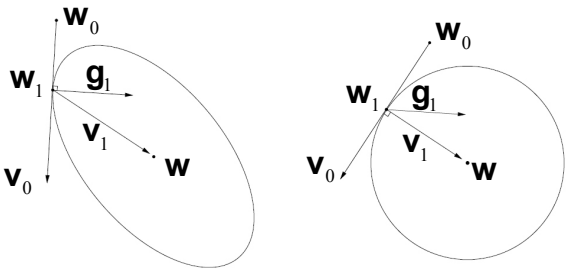


Figure 9: Optimality of CG method.

The SD methods are slow due to the successive gradient orthogonality that results of minimize the recursive updating equation (8) respect to $\mu[m]$. The movement toward a minimum has the zigzag form. The left part in Figure 9 shows the quadratic function contours in a real space (for $\mathbf{r} \neq \mathbf{0}$ in (2) are elliptical). Any pair of vectors that appear perpendicular in this space would be orthogonal. The right part shows the same drawing in a space that is stretched along the eigenvector axes so that the elliptical contours from the left part become circular. Any pair of vectors that appear to be perpendicular in this space is in fact $\mathbf{R}$-orthogonal. The search for a minimum of the quadratic function starts at $\mathbf{w}_0$, and takes a step in the direction $\mathbf{v}_0$ and stops at the point $\mathbf{w}_1$. This is a minimum point along that direction, determined in the same way for SD method. While the SD method would search in the direction $\mathbf{g}_1$, the CG method would chose $\mathbf{v}_1$. In this stretched space, the direction $\mathbf{v}_0$ appears to be a tangent to the now circular contours at the point $\mathbf{w}_1$. Since the next search direction $\mathbf{v}_1$ is constrained to be $\mathbf{R}$-orthogonal to the previous, they will appear perpendicular in this modified space. Hence, $\mathbf{v}_1$ will take us directly to the minimum point of the quadratic function (2nd order in the example).

MECHANICAL SYSTEM MODELLING OF ROBOT DYNAMICS USING A MASS/PULLEY MODEL

L. J. Stocco and M. J. Yedlin
The Department of Electrical and Computer Engineering, The University of British Columbia
2332 Main Mall, Vancouver, BC, Canada, V6T 1Z4
leos@ece.ubc.ca, matty@ece.ubc.ca

Keywords: Mass matrix, inertia matrix, MP model, pulley, differential transmission, mechanical system representation, robot dynamics, impedance, equivalent electric circuit.

Abstract: The well-known electro-mechanical analogy that equates current, voltage, resistance, inductance and capacitance to force, velocity, damping, spring constant and mass has a shortcoming in that mass can only be used to simulate a capacitor which has one terminal connected to ground. A new model that was previously proposed by the authors that combines a mass with a pulley (MP) is shown to simulate a capacitor in the general case. This new MP model is used to model the off-diagonal elements of a mass matrix so that devices whose effective mass is coupled between more than one actuator can be represented by a mechanical system diagram that is topographically parallel to its equivalent electric circuit model. Specific examples of this technique are presented to demonstrate how a mechanical model can be derived for both a serial and a parallel robot with both two and three degrees of freedom. The technique, however, is extensible to any number of degrees of freedom.

1 INTRODUCTION

The concept of impedance and its generalization reactance, has been used to define equivalent circuits of mechanical and electro-mechanical systems since the development of the Maxwell model of solids. The idea that driving point impedances could be decomposed into terms that parallel electrical elements was initiated by (Foster, 1924) who showed that the frequency response of any system is determined by the poles and zeros of its transfer function. The conditions for network synthesis are described by (Brune, 1931) and later applied by (Paynter 1961) who introduced bond graphs to distinguish and represent effort and flow variables in a graphical setting. Examples of electro-mechanical system simulations are numerous and include magnetic circuits (Hamill, 1993), mechatronics and electromechanical transducers (Tilmans, 1996), (van Amerongen & Breedveld, 2003), (Sass et al., 2004).

Mechanical block diagrams are routinely used to model robot dynamics although some (Eppinger & Seering, 1992) limit them to a single axis while others (Yamakita et al., 1992) rely entirely on equivalent electric circuits to avoid the inherent difficulties of creating mechanical models of multi-axis devices, transmission systems or other systems with coupled dynamics.

Section 2 of this paper describes the conventional electro-mechanical analogy and points out a limitation of the mass model. It goes on to describe a new mass/pulley (MP) model which overcomes the inherent deficiency in the conventional mass model. In Section 3, it is shown how the new MP model can be used to model the dynamics of devices which have coupled effective masses. Examples are provided which include both 2-DOF and 3-DOF serial and parallel manipulators. Lastly, concluding remarks are made in Section 4.

2 ELECTRO-MECHANICAL ANALOGIES

The ability to define an electro-mechanical equivalent circuit stems from the parallelism in the differential equations that describe electrical and mechanical systems, each of which involve an across variable, a through variable and an impedance or admittance variable. In electrical circuits, voltage $E(s)$ is the across variable and current $I(s)$ is the through variable. In mechanical systems, velocity $V(s)$ is the across variable and force $F(s)$ is the through variable (i.e. flow variable

(Fairlie-Clarke, 1999)). This results in a correspondence between resistance R and damping B, inductance L and spring constant K, and capacitance C and mass M shown in (1-3). An alternate approach treats force as the across variable and velocity as the through variable but that approach is not used here. By (1-3), the electromechanical equivalents shown in Figure 1 can be substituted for one another to model a mechanical system as an electrical circuit and vise versa.

$$E(s) = I(s)R = I(s)\frac{1}{G} \qquad V(s) = F(s)\frac{1}{B} \qquad (1)$$

$$E(s) = I(s)sL \qquad V(s) = F(s)\frac{s}{K} \qquad (2)$$

$$E(s) = I(s)\frac{1}{sC} \qquad V(s) = F(s)\frac{1}{sM} \qquad (3)$$

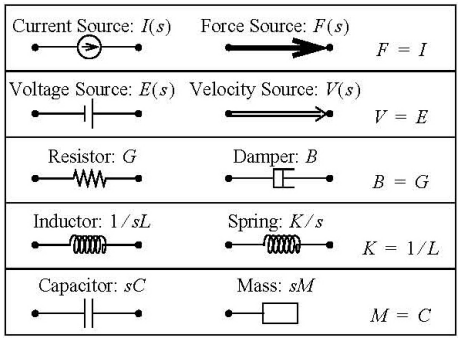


Figure 1: Admittance of electro-mechanical equivalents.

2.1 Classical Mass Model Limitation

Each of the components in Figure 1 has two terminals except for the mass which has only one. This is due to the fact that the dynamic equation of a mass (3) does not accommodate an arbitrary reference. Acceleration is always taken with respect to the global reference, or ground. Consider the two systems in Figure 2 which are well known to be analogous.

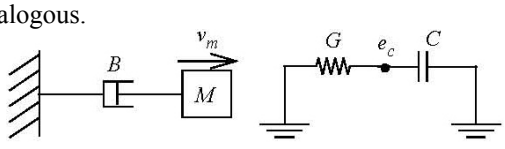


Figure 2: LC circuit and mechanical equivalent.

In Figure 2, the voltage across the capacitor e_c corresponds to the velocity of the mass v. Both of these are relative measurements that only correspond to one another because both are taken with respect to ground. Consider, on the other hand, the circuit in Figure 3 which contains a capacitor with one terminal open circuited.

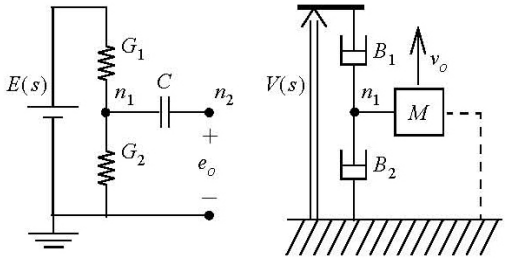


Figure 3: RC circuit and mechanical equivalent.

In Figure 3, the capacitor carries no current and therefore, has no effect on the output voltage e_o. In other words, the voltages at n_1 and n_2 are equal so the capacitor behaves like a short circuit. In the mechanical "equivalent", it is not possible to connect a non-zero mass M to node n_1 without affecting the output velocity v_o. This is due to the implicit ground reference of the mass (shown by a dotted line) which prevents it from ever behaving like a mechanical short circuit. Note that this same limitation does not apply to the spring or damper since they both act as a mechanical short circuit (infinitely stiff connection) if one terminal is left unconnected, just like their electrical counterparts, the inductor and resistor.

2.2 The Mass/Pulley (MP) Model

Because of the above limitation, there are mechanical systems which can not be modelled using a mechanical system diagram. Elaborate transmission systems such as robotic manipulators may contain mass elements that are only present when relative motion occurs between individual motion stages. Currently, systems such as these can only be modelled using electric circuits since capacitors can be used to model this type of behaviour but masses cannot.

It would be useful to have a mechanical model which simulates the behaviour of a capacitor without an implicit ground connection so that any mechanism (or electric circuit) could be modelled by a mechanical system diagram. This new model should have two symmetric terminals (i.e. flipping the device over should not affect its response), obey Ohm's Law, and be able to accommodate non-zero velocities at both terminals simultaneously. A model proposed by the authors (Stocco & Yedlin, 2006) combines a mass with the pulley-based differential

transmission shown in Figure 4. The pulley system obeys the differential position / velocity relationship shown in (4,5).

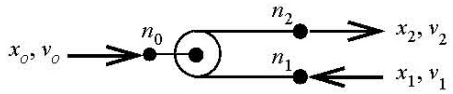


Figure 4: Pulley based differential transmission.

$$\Delta x_O = \frac{1}{2}(\Delta x_2 - \Delta x_1) \quad (4)$$

$$v_O = \frac{1}{2}(v_2 - v_1) \quad (5)$$

Note from (5) that although the pulley provides the desired differential velocity input, it also introduces an undesired 2:1 reduction ratio. However, setting v_1 to 0 (i.e. connecting n_1 to ground) results in (6). Therefore, a similar pulley system with one input tied to ground could be used to scale up velocity by an equivalent ratio.

$$v_2 = 2v_O \quad (6)$$

The double pulley system shown in Figure 5 is a differential transmission with a unity gear ratio. The primary pulley provides the differential input while the secondary pulley cancels the reduction ratio to achieve unity gain. A mass connected to the secondary pulley is accelerated by a rate equal to the difference between the acceleration of the two inputs, n_1 and n_2. This system simulates the behaviour of a capacitor that may or may not be connected to ground (Figure 5). Voltage E_1 corresponds to velocity V_1, voltage E_2 corresponds to velocity V_2, current I corresponds to tension F and capacitance C corresponds to mass M as shown by (7,8). Note that the free-body diagram of the centre pulley shows that the tension F in the primary cable is equal to the tension F in the secondary cable. The system must be balanced because any net force on the massless centre pulley would result in infinite acceleration of the pulley and therefore, the mass as well.

$$E_2(s) - E_1(s) = I(s)\frac{1}{sC} \quad (7)$$

$$V_2(s) - V_1(s) = F(s)\frac{1}{sM} \quad (8)$$

The MP model uses ideal cables with zero mass and infinite length and stiffness. The ideal cables travel through the system of massless, frictionless pulleys without any loss of energy. The MP model operates in zero gravity so the mass is only accelerated as a result of cable tension and/or compression. Unlike practical cables, the ideal cables never become slack. When an attractive force is applied between n_1 and n_2, $F<0$ and the mass is accelerated downward. A block diagram of the MP model is presented in Figure 6 where P has the same value as M in Figure 5. Note that, unlike a pure mass, the MP model has two terminals, n_1 and n_2 which correspond to the two ends of the primary cable.

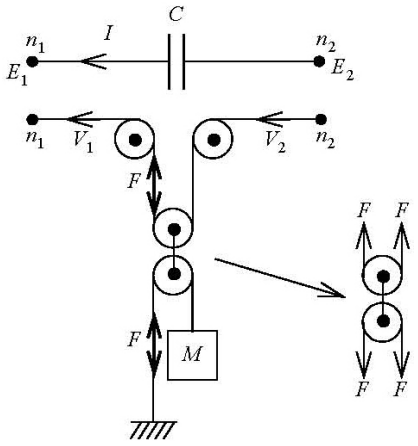


Figure 5: Mass / pulley equivalent of a capacitor.

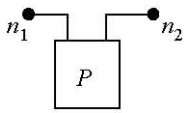


Figure 6: Block diagram of MP model.

Consider Figure 7 which is the mechanical system from Figure 3 with the mass replaced by an MP model. With terminal n_2 left unconnected, the primary cable of the MP model travels freely through the primary pulley without accelerating the mass or consuming energy. The MP model behaves like a mechanical short circuit, just like the capacitor in Figure 3. Also note the topological similarity between the electrical circuit in Figure 3 and its true mechanical equivalent in Figure 7. This is a direct result of the topological consistency between the capacitor and the MP model, both of which have two symmetric terminals. As pointed out in (Stocco & Yedlin, 2006), this consistency allows one to analyze mechanical systems using electric circuit analysis techniques once all masses have been replaced by MP models.

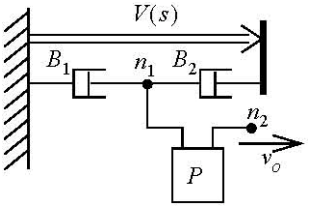


Figure 7: Mechanical equivalent using MP model.

3 ROBOT MASS MATRIX

Consider the simplified dynamics of a 2-DOF robot (9) where M is the mass matrix, B is the damping matrix, F is a vector of joint forces/torques (10), R is a vector of joint rates r_1 and r_2 (10), and s is the Laplace operator. Spring constants, gravitational and coriolis effects are assumed to be negligible for the purpose of this example. If the damping in the system is dominated by the actuator damping coefficients, B is a diagonal matrix (10). M, on the other hand, represents the effective mass perceived by each joint and is not diagonal or otherwise easily simplified in general.

$$F = BR + MsR \tag{9}$$

$$\begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} b_1 & 0 \\ 0 & b_2 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} + Ms \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} \tag{10}$$

For simple kinematic arrangements such as the redundant actuators shown in Figure 8 which only have a single axis of motion, M is shown in (11). The system responses are modeled by the mechanical system diagram shown in Figure 9 and the dynamic equation shown in (10). Using the electromechanical transformation described in Section 2, this system can also be represented by the electrical circuit analogy shown in Figure 9.

$$M = \begin{bmatrix} m_1 & m_2 \\ m_2 & m_2 \end{bmatrix} \tag{11}$$

Performing nodal analysis on the circuit in Figure 9 results in (12) by inspection. Note however, that (12) contains the term i_1-i_2 as well as v_2 which corresponds to the end-point velocity in the mechanical system or, in other words, the sum of the joint rates r_1+r_2. To obtain a correspondence between electrical and mechanical component values, the dynamic equation (10) is rearranged in (13) where the associated damping B' and mass M' matrices are shown in (14,15). From (14), the resistor admittances g_1 and g_2 and capacitor values c_1 and c_2 correspond to the equivalent damping and mass values b'_1, b'_2, m'_1 and m'_2 (16) respectively.

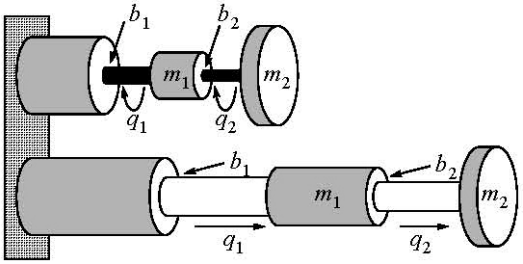


Figure 8: Redundant rotary & prismatic actuators.

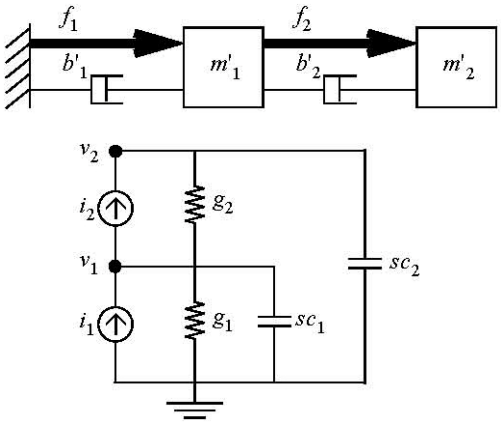


Figure 9: System models of redundant actuators.

$$\begin{bmatrix} i_1 - i_2 \\ i_2 \end{bmatrix} = \begin{bmatrix} g_1 + g_2 & -g_2 \\ -g_2 & g_2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + \begin{bmatrix} c_1 & 0 \\ 0 & c_2 \end{bmatrix} s \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{12}$$

$$\begin{bmatrix} f_1 - f_2 \\ f_2 \end{bmatrix} = B' \begin{bmatrix} r_1 \\ r_1 + r_2 \end{bmatrix} + M's \begin{bmatrix} r_1 \\ r_1 + r_2 \end{bmatrix} \tag{13}$$

$$B' = \begin{bmatrix} b'_1 + b'_2 & -b'_2 \\ -b'_2 & b'_2 \end{bmatrix} = \begin{bmatrix} b_1 + b_2 & -b_2 \\ -b_2 & b_2 \end{bmatrix} \tag{14}$$

$$M' = \begin{bmatrix} m'_1 & 0 \\ 0 & m'_2 \end{bmatrix} = \begin{bmatrix} m_1 + m_2 & 0 \\ 0 & m_2 \end{bmatrix} \tag{15}$$

$$\begin{bmatrix} b'_1 \\ b'_2 \\ m'_1 \\ m'_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ m_1 + m_2 \\ m_2 \end{bmatrix} \tag{16}$$

In this simple example, masses are sufficient to model the system behaviour but only because the device has a single degree of freedom so M' is diagonal and there is no cross-coupling between actuators. In general, however, effective mass is not always decoupled and the off-diagonal elements of M' can be expected to be non-zero. When M' is not diagonal, conventional single-terminal masses are unable to model the entire effective mass of the system. They can not model the off-diagonal terms that describe inertial effects resulting from relative motion of the actuators.

3.1 Serial 2-DOF Robot

Consider the 2-DOF serial robot shown in Figure 10. The mass matrix for this mechanism is approximated in (Craig, 1989) by two point masses d_1 and d_2 placed at the distal actuator and end-effector as indicated below. The resulting mass matrix (17) has the terms shown in (18-20) where q_1 and q_2 are the joint angles and l_1 and l_2 are the link lengths. Just as in the previous example, actuator damping coefficients b_1 and b_2 are taken to dominate the total system damping.

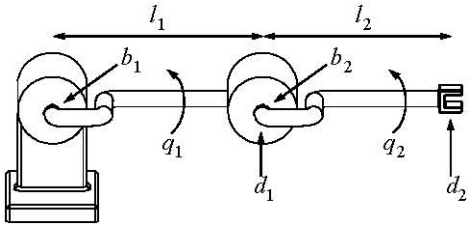


Figure 10: 2-DOF serial robot.

$$M(q) = \begin{bmatrix} m_1(q) & m_3(q) \\ m_3(q) & m_2(q) \end{bmatrix} \quad (17)$$

$$m_1 = l_2^2 d_2 + 2 l_1 l_2 d_2 \cos(q_2) + l_1^2 (d_1 + d_2) \quad (18)$$

$$m_2 = l_2^2 d_2 \quad (19)$$

$$m_3 = l_2^2 d_2 + l_1 l_2 d_2 \cos(q_2) \quad (20)$$

The equivalent circuit model of this system is shown in Figure 11. It is similar to Figure 9 except that the capacitor values are configuration dependent and a third capacitor c_{12} is included to model the coupled mass terms that are present. Performing nodal analysis results in (21) and the corresponding M' matrix in (22) which can be rearranged to solve for the mechanical model parameters in terms of the physical mass values in (23). B' is the same diagonal matrix as in (14).

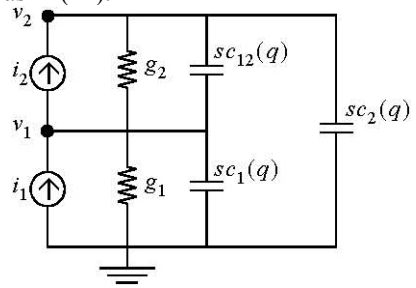


Figure 11: Electrical model of 2-DOF serial robot.

$$\begin{bmatrix} i_1 - i_2 \\ i_2 \end{bmatrix} = \begin{bmatrix} g_1 + g_2 & -g_2 \\ -g_2 & g_2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + \begin{bmatrix} c_1 + c_{12} & -c_{12} \\ -c_{12} & c_2 + c_{12} \end{bmatrix} s \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (21)$$

$$M'(q) = \begin{bmatrix} m'_1 + p'_{12} & -p'_{12} \\ -p'_{12} & m'_2 + p'_{12} \end{bmatrix} = \begin{bmatrix} m_1 + m_2 & m_3 - m_2 \\ m_3 - m_2 & m_2 \end{bmatrix} \quad (22)$$

$$\begin{bmatrix} m'_1 \\ m'_2 \\ p'_{12} \end{bmatrix} = \begin{bmatrix} m_1 + m_3 \\ m_3 \\ m_2 - m_3 \end{bmatrix} \quad (23)$$

Note from (22) that M' is diagonal (i.e. $p'_{12}=0$) when $p'_{12}=0$. From (19,20), this is merely the special case when $q_2 = \pm\pi/2$. Therefore, it is not possible to model this system using only masses due to their implicit ground reference, as described in Section 2.1. The off-diagonal terms can, however, be modelled using the MP model proposed in Section 2.2. It results in a mechanical system model that is topologically identical to the equivalent circuit in Figure 11 where each grounded capacitor (c_1,c_2) is replaced by a regular mass and each ungrounded capacitor (c_{12}) is replaced by an MP model since the MP model is able to accommodate a non-zero reference acceleration. The resulting mechanical system is shown in Figure 12.

Although p'_{12} has a negative value when $-\pi/2 < q_2 < \pi/2$, the net mass perceived by each actuator is always positive because M is positive definite. When p'_{12} is negative, it simply means that the motion of actuator 1 reduces the net mass perceived by actuator 2, but the net mass perceived by actuator 2 is always greater than zero.

3.2 Parallel 2-DOF Robot

The same technique can be applied to parallel manipulators such as the 2-DOF 5-bar linkage used

by (Hayward et al., 1994). In the case of parallel manipulators, each actuator is referenced to ground but there remains a coupling between the effective mass perceived by each actuator which, like a serial manipulator, is configuration dependent. This coupling is modelled by c_{12} and p'_{12} in the equivalent electrical and mechanical models shown in Figure 13. Typically, parallel manipulators also have coupled damping terms due to their passive joints which would be modelled by a conductance g_{12} added between nodes 1 and 2 (i.e. in parallel with c_{12}). However, for the sake of simplicity, the damping of the passive joints are neglected here.

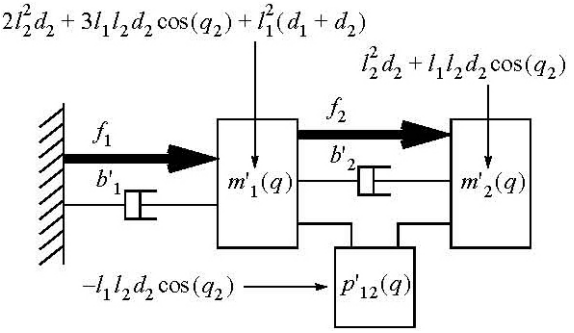


Figure 12: Mechanical model of a 2-DOF serial robot.

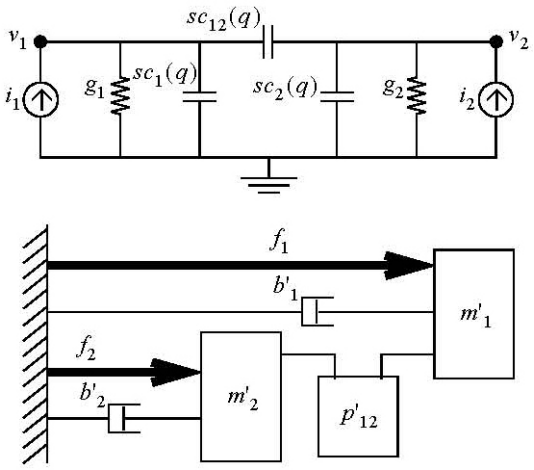


Figure 13: Model of a 2-DOF parallel robot.

Performing nodal analysis on the circuit in Figure 13 results in (24) by inspection. For a parallel robot, currents and voltages correspond directly to joint forces and joint rates so $B'= B$ and $M'= M$. For a mass matrix of the form shown in (17), the elements of the M' matrix, and therefore the parameter values associated with the masses and MP models of Figure 13, are shown in (26).

$$\begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} g_1 & 0 \\ 0 & g_2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + \begin{bmatrix} c_1 + c_{12} & -c_{12} \\ -c_{12} & c_2 + c_{12} \end{bmatrix} s \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (24)$$

$$\begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} b'_1 & 0 \\ 0 & b'_2 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} + \begin{bmatrix} m'_1 + p'_{12} & -p'_{12} \\ -p'_{12} & m'_2 + p'_{12} \end{bmatrix} s \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} \quad (25)$$

$$\begin{bmatrix} m'_1 \\ m'_2 \\ p'_{12} \end{bmatrix} = \begin{bmatrix} m_1 + m_3 \\ m_2 + m_3 \\ -m_3 \end{bmatrix} \quad (26)$$

3.3 Multiple DOF Robots

This technique is easily extended to devices with any number n of degrees of freedom. With serial manipulators, the compliance and damping is often mainly in the actuators and the damping B and spring K matrices are diagonal (27,28). With parallel manipulators, the B and K matrices typically contain off-diagonal terms but they are easily modelled using conventional techniques since springs and dampers are 2-terminal devices which can be placed at any two nodes in a system diagram.

$$B = diag\left(\begin{bmatrix} b_1 & b_2 & \dots & b_n \end{bmatrix}\right) \quad (27)$$

$$K = diag\left(\begin{bmatrix} 1/k_1 & 1/k_2 & \dots & 1/k_n \end{bmatrix}\right) \quad (28)$$

To account for inertial cross-coupling, the model must contain a capacitor and/or MP model between every pair of actuators. For example, the electric circuit model and corresponding mechanical system model of a serial 3-DOF manipulator are shown in Figure 14. The capacitance C matrix resulting from the nodal analysis (29) of the circuit in Figure 14 is shown in (30).

$$\begin{bmatrix} i_1 - i_2 \\ i_2 - i_3 \\ i_3 \end{bmatrix} = G(q) \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} + C(q)s \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad (29)$$

$$C(q) = \begin{bmatrix} c_1 + c_{12} + c_{13} & -c_{12} & -c_{13} \\ -c_{12} & c_2 + c_{12} + c_{23} & -c_{23} \\ -c_{13} & -c_{23} & c_3 + c_{23} + c_{13} \end{bmatrix} \quad (30)$$

Just as in the previous examples, the 3x3 mass matrix M' (32) is rearranged into the form shown in (31) to parallel the current/voltage relationship of (29). For the mass matrix M of the form shown in (33), the entries of the M' matrix are solved for in (34).

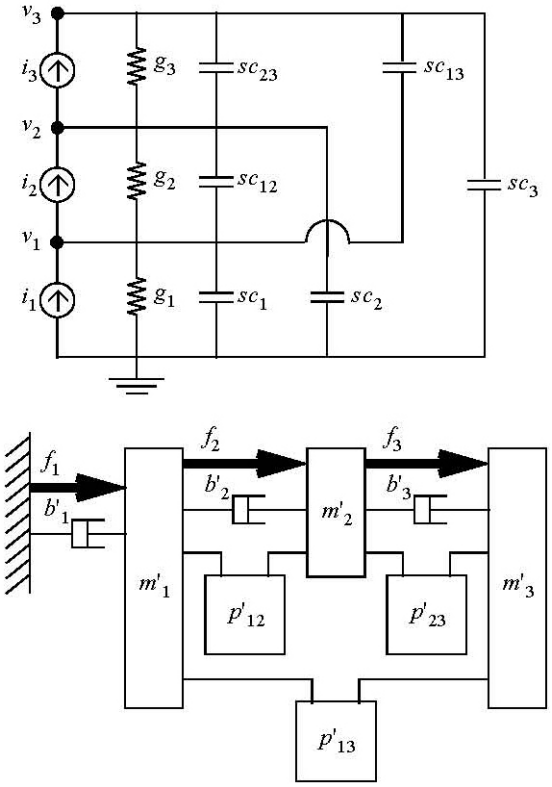


Figure 14: Model of a 3-DOF serial robot.

Similarly, for a parallel 3-DOF robot, the electric circuit model and corresponding mechanical system model are shown in Figure 15. For a mass matrix of the form shown in (33), the elements of M' are shown in (35).

$$\begin{bmatrix} f_1 - f_2 \\ f_2 - f_3 \\ f_3 \end{bmatrix} = B' \begin{bmatrix} r_1 \\ r_1 + r_2 \\ r_1 + r_2 + r_3 \end{bmatrix} + M's \begin{bmatrix} r_1 \\ r_1 + r_2 \\ r_1 + r_2 + r_3 \end{bmatrix} \quad (31)$$

$$M'(q) = \begin{bmatrix} m'_1 + p'_{12} + p'_{13} & -p'_{12} & -p'_{13} \\ -p'_{12} & m'_2 + p'_{12} + p'_{23} & -p'_{23} \\ -p'_{13} & -p'_{23} & m'_3 + p'_{13} + p'_{23} \end{bmatrix} \quad (32)$$

$$M(q) = \begin{bmatrix} m_1(q) & m_4(q) & m_5(q) \\ m_4(q) & m_2(q) & m_6(q) \\ m_5(q) & m_6(q) & m_3(q) \end{bmatrix} \quad (33)$$

$$\begin{bmatrix} m'_1 \\ m'_2 \\ m'_3 \\ p'_{12} \\ p'_{23} \\ p'_{13} \end{bmatrix} = \begin{bmatrix} m_1 - m_4 \\ m_4 - m_5 \\ m_5 \\ m_2 + m_5 - m_4 - m_6 \\ m_3 - m_6 \\ m_6 - m_5 \end{bmatrix} \quad (34)$$

$$\begin{bmatrix} m'_1 \\ m'_2 \\ m'_3 \\ p'_{12} \\ p'_{23} \\ p'_{13} \end{bmatrix} = \begin{bmatrix} m_1 + m_4 + m_6 \\ m_2 + m_4 + m_6 \\ m_3 + m_5 + m_6 \\ -m_4 \\ -m_5 \\ -m_6 \end{bmatrix} \quad (35)$$

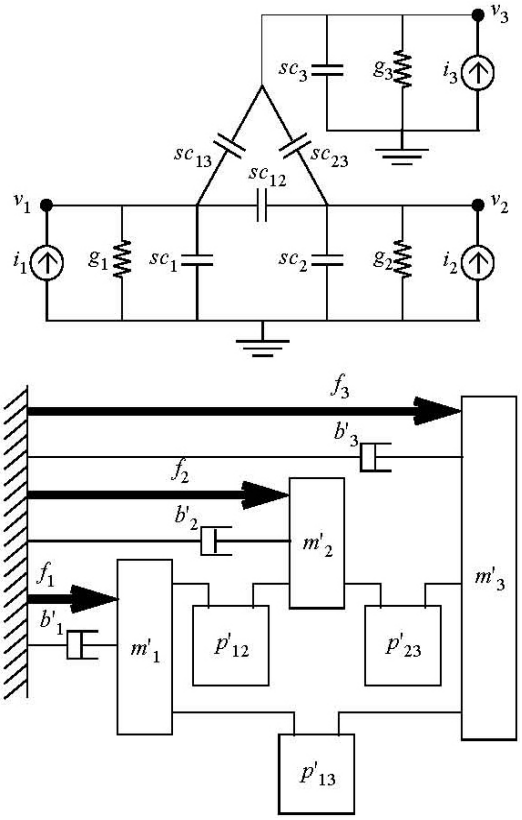


Figure 15: Model of a 3-DOF parallel robot.

4 CONCLUSION

It is argued that a plain mass is not a complete and general model of a capacitor since a mass only has

one terminal whereas a capacitor has two. The response of a mass corresponds to its acceleration with respect to ground and, therefore, can only be used to simulate a capacitor which has one terminal connected to ground. It cannot be used to simulate a capacitor which has a non-zero reference voltage. A new model described here that consists of a mass and a pulley correctly simulates the response of a capacitor in the general case.

It is shown that the MP model can be used to model systems with cross-coupled effective masses which are otherwise, impossible to model with pure masses alone. This includes both serial and parallel manipulators with any number of degrees of freedom. The mechanical system model that is obtained fully describes the dynamic response of the system and is topologically identical to its electric circuit equivalent. As shown in (Stocco & Yedlin, 2006), this makes it possible to apply electric circuit analysis techniques to mechanical systems, directly.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge Tim Salcudean for his valuable comments during the preparation of this manuscript.

REFERENCES

Brune, O., 1931. "Synthesis of a finite two-terminal network whose driving-point impedance is a prescribed function of frequency". *J. Math. Physics*. vol. 10, pp. 191-236.

Craig, J.J., 2005. "Introduction to Robotics Mechanics and Control". 3rd ed., *Pearson Prentice Hall*.

Eppinger, S., Seering, W., 1992. "Three Dynamic Problems in Robot Force Control". *IEEE Trans. Robotics & Auto.*, V. 8, No. 6, pp. 751-758.

Fairlie-Clarke, A.C., 1999. "Force as a Flow Variable". *Proc. Instn. Mech. Engrs.*, V. 213, Part I, pp. 77-81.

Foster, R. M., 1924. "A reactance theorem". *Bell System Tech. J.*, vol. 3, pp. 259-267.

Hamill, D.C., 1993. "Lumped Equivalent Circuits of Magnetic Components: The Gyrator-Capacitor Approach". *IEEE Transactions on Power Electronics*, vol. 8, pp. 97.

Hayward, V., Choksi, J., Lanvin, G., Ramstein, C., 1994. "Design and Multi-Objective Optimization of a Linkage for a Haptic Interface". *Proc. of ARK '94, 4th Int. Workshop on Advances in Robot Kinematics* (Ljubliana, Slovenia), pp. 352-359.

Paynter, H.M., 1961. *Analysis and Design of Engineering Systems*. MIT Press.

Sass, L., McPhee, J., Schmitke, C., Fisette, P., Grenier, D., 2004. "A Comparison of Different Methods for Modelling Electromechanical Multibody Systems". *Multibody System Dynamics*, vol. 12, pp. 209-250.

Stocco, L., Yedlin, M., Sept. 2006. "Closing the Loop on the Electro-Mechanical System Analogy". *Submitted to: IEEE J. Circuits & Systems*.

Tilmans, H.A.C., 1996. "Equivalent circuit representation of electromechanical transducers: I. Lumped-parameter systems". *J. Micromech. Microeng*, vol. 6, pp. 157-176.

van Amerongen, J., Breedveld, P., 2003. "Modelling of physical systems for the design and control of mechatronic systems". *Annual Reviews in Control*, vol. 27, pp. 87-117.

Yamakita, M., Shibasato, H., Furuta, K., 1992. "Tele-Virtual Reality of Dynamic Mechanical Model". *Proc. IEEE/RSJ Int. Conf. Intelligent Robots & Systems*, (Raleigh, NC), pp. 1103,-1110.

STUDY OF A CONTROLED COMPLEX MECHANICAL SYSTEM IN ANTI VIBRATORY DOMAIN

Application to a Hard Landing of an Aircraft

Cédric Lopez, François Malburet
Laboratoire des Sciences de l'Information et des Systèmes, équipe Ingéniérie Mécanique des Systèmes, ENSAM
2 cours des Arts et Métiers, 13617 Aix en Provence,France
cedric.lopez@aix.ensam.fr, francois.malburet@aix.ensam.fr

André Barraco
Laboratoire de Mécanique des Systèmes et des Procédés, ENSAM
151 Boulevard de l'Hôpital, 75013 Paris, France
andre.barraco@paris.ensam.fr

Keywords: Control, excitation, high speed shock, mechanical coupling, minimization, modeling, oscillations, PID, sliding mode, test bench.

Abstract: This paper studies problematic of a mechanical system composed of different parts mechanically coupled and submitted to a high speed shock.
After a shock, different parts of the system oscillate. If one of them is excited at a particular frequency, such as its proper frequency, important oscillations appear and can lead to the deterioration of the system by introducing important stresses. In this paper, we propose an analysis in order to understand this kind of problem and what we can do to avoid it. Firstly we discuss problematic and we expose the studied system. In a second time, we present model which allows us to understand the phenomenon by carrying out numerical simulations. Then we complete a comparative analysis of different methods of control. Prospects and problematic of real controlled device are studied. Finally experimental set up is described.

1 INTRODUCTION

The topic of this paper takes place in the problematic of the struggle against vibrations. More particularly in the minimisation of induced vibrations by a high speed shock in a complex mechanical system.

Vibrations and their effects are very problematic phenomenoms for all mechanical systems. Although there are a lot of applications, the overall of anti vibratory devices aim the increase of the service life of machines and structures but also the increase of the comfort of passengers in means of transportation.

In fact several complex systems are submitted to external and internal excitations. There are external excitations, like earthquakes or wind for buildings and structures for example and road disturbances (pothole for example) for vehicles. Internal excitations are issued from mechanical pieces in movement or out of balance for mechanical system. Here we study vibrations induced by external excitation and more especially these ones induced by shock.

Aeronautics is a domain where it is important to study the behaviour of an excited system. In fact progress in the domain of materials leads frames of aircrafts to be lighter. These ones easily bend under an excitation. During taxiing, the fuselage is submitted to excitations which lead to uncomfortable situation for passengers and stressful vibrations for the frame (Kruger, 2000). Moreover aircrafts are particularly constrained during a landing and especially a hard landing which is equivalent to a high speed shock. In fact because of the mechanical coupling existing between the fuselage and the landing gear, the frame of the aircraft bends and important deformations, resulting of a particular excitation of the frame, can lead to the deterioration of the aircraft. Reinforcement of the fuselage can be made. But this passive solution

makes the aircraft heavier.

So in order to insure comfort of passengers and to evict vibrations in fuselage, Ghiringhelli proposes to control the landing gears (Ghiringhelli, 2000). In this study, he only takes into account the cabin of an aircraft. Here we take into account the tail beam, a particular critic component that can easily bends under a high speed shock and whose oscillations lead to important stress in the area of the joint between the cabin and the tail beam. This phenomenon is particularly enhanced on helicopter.

Thus in order to analyse problematic and to understand the phenomenon, we study the behaviour of a mechanical system composed of different parts mechanically coupled and submitted to a high speed shock.

In order to reproduce a high speed shock, we study the free fall and the impact on the ground of the system. The behaviour of the upper part of the system is particularly studied because it represents for example the tail beam of an aircraft and so we want to understand and to avoid its oscillations.

Thus we firstly present modelization of studied system in order to carry out numerical simulations. Then we complete an analysis of different methods of control. A prospect of real device is introduced. Finally experimental set-up is exposed.

2 MODELING

2.1 Description

In a first time, in order to simplify the study only the main movement of bounce is taken into account. The studied system is composed of a system which is equivalent to a quarter part of a vehicle with another sprung mass located on the upper mass of the quarter part of a vehicle.

The quarter part of a vehicle is composed of a wheel, an unsprung mass (mns) and a sprung mass (ms) linked by a suspension (cf. Figure 1 and Figure 5). The subsystem located on the sprung mass of the quarter part of the vehicle, is composed of a mass (mq), a spring and a damper. Its damping rate is about 3%, which corresponds to a structural damping.

We have a free fall of the system; so the speed of the shock is proportional to the height of the fall. Here we study a shock with a speed of 3 m/s. The height of the fall is 0.4 m. In all following simulations, initial conditions on positions of different masses making up the system, allow us to adjust the speed of the shock.

Two approaches have been studied. An analytical approach and a multi body approach have been presented in a previous paper. Multi body approach corresponds to a non linear model based on experimental characterizations of some different constitutive parts of the system such as the tire and the hydraulic shock absorber. After study, the non linear model can be linearized. In this paper, we only present and study the linear analytical model. Moreover this one has been cross checked with experimental tests made on the drop test bench, described in the following of this paper.

The studied system is also described by the following figure:

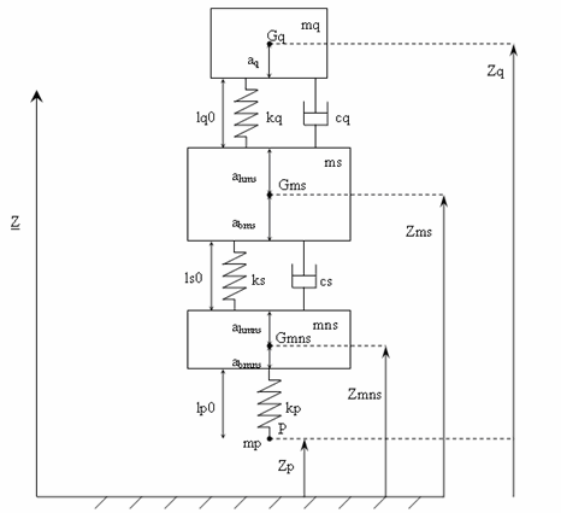


Figure 1: Model and definition of parameters.

We consider four degrees of freedom (d.o.f), which are:

-Zq, absolute displacement of the centre of mass mq.

-Zms, absolute displacement of the centre of mass ms.

-Zmns, absolute displacement of the centre of mass mns.

-Zp, absolute displacement of the point P.

REMARK.– setting conditions on the absolute displacement of the point P, which corresponds to the bottom point of the tire, allow us to differentiate the phase of fall and the phase of contact with the ground during simulations.

In fact we have following conditions:

$Zp>0$, phase of fall.

$Zp\leq 0$, phase of evolution of the system on ground.

Notations:

-mq, mass of the upper system.

-Gq, centre of mass mq.

-ms, sprung mass.
-Gms, centre of mass ms.
-mns, unsprung mass.
-Gmns, centre of mass mns.
-kq, stiffness of the upper system.
-lq0, length of the unloaded spring kq.
-cq, damping coefficient of the upper system.
-cs, damping coefficient of the suspension.
-ks, stiffness of the suspension.
-ls0, length of the unloaded spring ks.
-kp, stiffness of the tire.
-lp0, length of the unloaded tire.
-P, point of contact of the tire.
-a_i, distance between a centre of mass and the point of application of a spring. The index i corresponds to the different notations used in Figure 1.

The behaviour of the system is described by the following equations:

$$\begin{aligned} mq\cdot\ddot{Z}q = &-mq\cdot g - kq\cdot(Zq - Zms) - cq\cdot(\dot{Z}q - \dot{Z}ms) \\ &- kq\cdot(-a_q - a_{hms} - lq0) \end{aligned} \quad (1)$$

$$\begin{aligned} ms\cdot\ddot{Z}ms = &-ms\cdot g + kq\cdot(Zq - Zms) + cq\cdot(\dot{Z}q - \dot{Z}ms) \\ &+ kq\cdot(-a_q - a_{hms} - lq0) - ks\cdot(Zms - Zmns) \\ &- ks\cdot(-a_{bms} - a_{hmns} - ls0) - cs\cdot(\dot{Z}ms - \dot{Z}mns) \end{aligned} \quad (2)$$

$$\begin{aligned} mns\cdot\ddot{Z}mns = &-mns\cdot g + ks\cdot(-a_{bms} - a_{hmns} - ls0) \\ &+ ks\cdot(Zms - Zmns) + cs\cdot(\dot{Z}ms - \dot{Z}mns) \\ &- kp\cdot(Zmns - Zp - a_{bmns} - lp0) \end{aligned} \quad (3)$$

$$mp\cdot\ddot{Z}p = -mp\cdot g + kp\cdot(Zmns - Zp) + kp\cdot(-a_{bmns} - lp0) \quad (4)$$

The mass mp is set to zero. When Zp>0, the system is falling, the tire represented by the spring with stiffness kp doesn't apply any force on the mass mns.

2.2 Simulations and Analysis

We study vibrations induced by a high speed shock. In this study, free fall of system is considered. Thus the speed of the shock is determined by the height of the fall ie initial positions of different masses. Here we analyse a shock with a speed of 3 m/s (a 0.4 m high fall). Moreover we set the following condition; no bounce of the system can occur. This is a condition of stability for an aircraft during landing or a condition of safety for a car riding on a chaotic road.

The upper system is composed of the mass mq, the spring kq and the damper cq. It has a low proper frequency about 7 Hz.

The damping coefficient (cs) of the suspension is different between the phase of compression and the phase of extension. This difference makes the suspension softer and guaranties no bounce.

After several simulations, we chose a damping rate of 60% for compression and 90% for extension. The damping rate, noted λ, is calculated as following:

$$\lambda = \frac{cs}{2\cdot\sqrt{ks\cdot ms}} \quad (5)$$

The stiffness of the spring kp modelling the tire is set to 250000 N/m. This is an average value of the used tire on the test bench.

We study the excitation force transmitted to the sprung mass (ms). We obtain the following result of simulation:

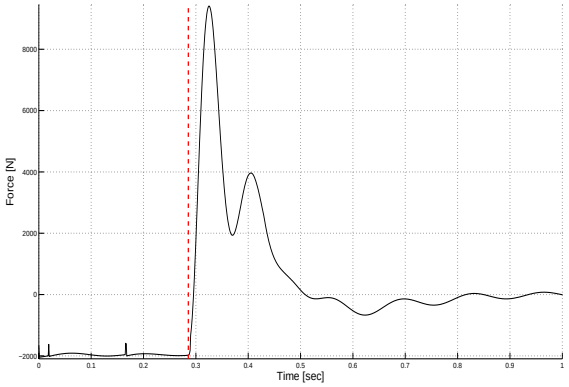


Figure 2: Excitation force on ms.

The impact occurs at the time 0.28 sec (outlined on the graph by the red vertical dashed lined). As soon as the impact occurs, we notice the presence of a double bump. The first peak depends on characteristics of the suspension (stiffness, damping rate). The second peak depends on the stiffness of the tire. The stiffer these elements are, the higher the peaks are. The duration of the double bump is equal to 0.13 sec.

Because the coupling between the mass ms and the mass mq, the double bump excites the upper system in a frequency band near its proper frequency; leading to important displacements of the mass mq.

We can conclude that the duration and the particular shape of the excitation transmitted by the suspension and resulting of the high speed shock are responsible for important displacements of the mass mq.

Thus in order to prevent important oscillations of the upper system, we have to control the transmitted excitation. This one is transmitted by the suspension. To control the dynamic behaviour of the suspension allows us to minimize oscillations of the mass mq and also to minimize the force on the upper system.

In the following, different methods of control of the dynamic behaviour are designed and a comparative analysis is presented. Then problematic and prospects of real device are exposed.

3 CONTROLED SYSTEM

3.1 Problematic

The previous work shows that the particular excitation transmitted by the suspension to the mass ms, leads to important oscillations of the mass mq.

Several studies propose different controled suspensions in order to minimize the acceleration of the mass ms (Giua et al., 2004; Guglielmino and Edge, 2004; Kim et al., 2003). The aim of all these studies is to minimize the acceleration of the mass ms in order to insure the comfort of passengers (Yagiz, 2004). Our aim is to minimize acceleration of the mass mq. In fact, according to the coupling between the sprung mass (ms) and the upper mass (mq), we will control the transmitted force on ms in order to minimize acceleration of the upper mass (mq).

In fact we can't add a control force on the upper system; that would mean a collocated actuator on the tail beam on a real aircraft. This is more difficult and less practicable than control the landing gear.

3.2 Comparative Analysis of Different Methods of Control

Here we compare different methods of control. First we study two classical methods of PID with feedback on ms measure of acceleration and then on mq measure of acceleration in order to respectively minimize acceleration on ms and on mq.

Then we design sliding mode controller with state feedback on ms using the existing coupling between the sprung mass (ms) and the upper mass (mq) in order to minimize the acceleration of mq.

We want to control the excitation force transmitted by the suspension to the sprung mass (ms). We introduce a control force, noted u, in the equations defining the system. This force is added on the sprung mass in parallel with passive force of damping and stiffness. According to equations (2) and (3) previously exposed we obtain:

$$\begin{aligned} ms\cdot\ddot{Z}ms = &-ms\cdot g + kq\cdot(Zq - Zms) + cq\cdot(\dot{Z}q - \dot{Z}ms) \\ &+ kq\cdot(-a_q - a_{hms} - lq0) - ks\cdot(Zms - Zmns) \\ &- ks\cdot(-a_{bms} - a_{hmns} - ls0) - cs\cdot(\dot{Z}ms - \dot{Z}mns) \\ &+ u \end{aligned} \quad (6)$$

$$\begin{aligned} mns\cdot\ddot{Z}mns = &-mns\cdot g + ks\cdot(-a_{bms} - a_{hmns} - ls0) \\ &+ ks\cdot(Zms - Zmns) + cs\cdot(\dot{Z}ms - \dot{Z}mns) \\ &- kp\cdot(Zmns - Zp - a_{bmns} - lp0) - u \end{aligned} \quad (7)$$

3.2.1 Design of PID Controller

Considering the Laplace domain, the transfer function used for the PID controller is the following:

$$H(p) = \frac{U(p)}{\varepsilon(p)} = K_p\cdot\left(1 + \frac{1}{T_i\cdot p} + \frac{T_d\cdot p}{a\cdot T_d\cdot p + 1}\right) \quad (8)$$

Where K_p, T_d, T_i and a are tuning parameters determined from simulations. $\varepsilon(p)$ is the offset between the set point and the measure of the considered parameter.

We study two approaches. First, we minimize the acceleration of the sprung mass (ms). On a second time, we minimize the acceleration of the upper mass (mq). In fact, we firstly minimize the acceleration of the sprung mass (ms) because according to mechanical coupling between the two masses, we want to analyse the behaviour of the upper mass (mq) using a PID controller in order to minimize the acceleration of the sprung mass (ms). Then we use the same PID controller with minimization of the upper mass (mq), always exerting the control force u on the sprung mass.
Results of the simulations of these two controled systems are presented and discussed in the following of this paper (cf. part 3.2.3).

3.2.2 Design of Sliding Mode Controller

Always using the mechanical coupling between the sprung mass (ms) and the upper mass (mq), we control the behaviour of the sprung mass (ms) using a sliding mode controller in order to minimize the acceleration of the upper mass (mq).

In this part we develop the design of the sliding mode controller which we will implement in the following. We have the following state vector:

$$\underline{x} = \begin{bmatrix} Zq \\ Zms \\ Zmns \\ \dot{Z}q \\ \dot{Z}ms \\ \dot{Z}mns \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \quad (9)$$

In order to design the sliding mode controller, we explain the system model as an affine system of the form:

$$\dot{\underline{x}} = \underline{f}(\underline{x}) + \underline{g} \cdot u \quad (10)$$

Using this form, we can write:

$$\begin{aligned} \dot{x}_1 &= x_4 \\ \dot{x}_4 &= f_4(x_j) \end{aligned} \quad (11)$$

$$\begin{aligned} \dot{x}_2 &= x_5 \\ \dot{x}_5 &= f_5(x_j) + g_5 \cdot u \end{aligned} \quad (12)$$

$$\begin{aligned} \dot{x}_3 &= x_6 \\ \dot{x}_6 &= f_6(x_j) + g_6 \cdot u \end{aligned} \quad (13)$$

Where j=1…6. Moreover we have:

$$f_4(x_j) = \frac{1}{mq} \cdot \begin{pmatrix} kq \cdot (x_2 - x_1) + cq \cdot (x_5 - x_4) \\ -mq \cdot g \end{pmatrix} \quad (14)$$

$$f_5(x_j) = \frac{1}{ms} \cdot \begin{pmatrix} kq \cdot x_1 + (-kq - ks) \cdot x_2 + ks \cdot x_3 \\ +cq \cdot x_4 + (-cq - cs) \cdot x_5 + cs \cdot x_6 \\ -ms \cdot g \end{pmatrix} \quad (15)$$

$$f_6(x_j) = \frac{1}{mns} \cdot \begin{pmatrix} ks \cdot x_2 + (-ks - kp) \cdot x_3 \\ +cs \cdot x_5 + (-cs) \cdot x_6 - mns \cdot g \end{pmatrix} \quad (16)$$

We consider the desired state x_2^d. The error between the actual and the desired state can be written as:

$$e = x_2 - x_2^d \quad (17)$$

Here we consider the switching surface s defined for second order system by:

$$s = \dot{e} + \lambda \cdot e \quad (18)$$

λ sets the dynamic in the sliding phase (s=0). The control force u must be chosen so that trajectory of the state approaches the switching surface and then stay on it for all future time; guarantying stability and convergence to desired state. It is compound of a sum of two terms as following:

$$u = u_{eq} + u^* \quad (19)$$

The first term called equivalent control, is defined according to parameters of the nominal system. It is expressed as:

$$u_{eq} = g_5^{-1} \cdot \left(\ddot{x}_2^d - \lambda \cdot \dot{e} - f_5(x_j) \right) \quad (20)$$

The second term is defined in order to tackle uncertainties and to introduce reaching law. It is defined by:

$$u^* = g_5^{-1} \cdot (-k \cdot s) \quad (21)$$

The parameter k is chosen by the designer in order to define a reaching rate.

Thus we obtain the following law of control:

$$u = g_5^{-1} \cdot \left(\ddot{x}_2^d - \lambda \cdot \dot{e} - k \cdot s \right) \quad (22)$$

Results of the simulations of this controled system are discussed in the following part.

3.2.3 Analysis of Simulations Results

Simulations of the previous designed controllers lead to following results:

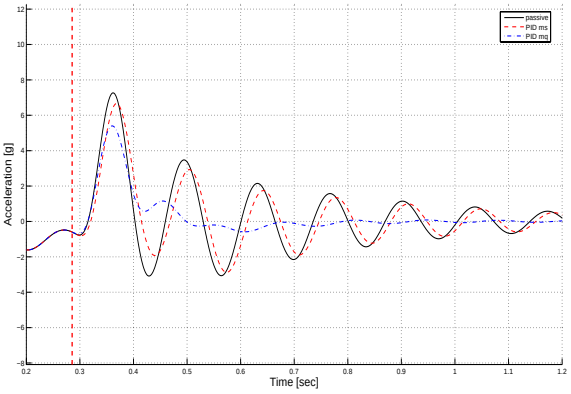


Figure 3: Acceleration of the mass (mq) - comparison between passive and PID controllers.

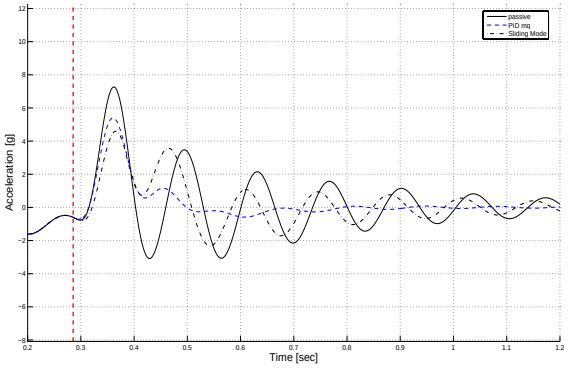


Figure 4: Acceleration of the mass (mq) - comparison between passive, PID and Sliding mode controllers.

The critical point occurs at the landing during the first compression of the landing gear. Thus, we want to minimize the amplitude of the first peak on the acceleration.

On Figure 3, we compare the passive system with PID controllers minimizing the acceleration of ms and mq (cf. part 3.2.1). The minimization of acceleration of ms using PID is not effective on the minimization of the acceleration of mq. Nevertheless minimization of acceleration of mq using the same PID is effective. In fact we have respectively a gain of 9% and 25% in comparison with the passive system.

On Figure 4, we compare the passive system with PID and sliding mode controllers. Here minimization of ms using sliding mode controller in order to minimize the acceleration of mq is very effective. We notice a gain of 35% in comparison with the passive system on the first peak of acceleration of the upper mass (mq).

Moreover in order to guaranty the stability of the system and optimize the behaviour in minimization of the acceleration of mq, the sliding mode controller is operative only at the impact of the system on the ground and during a defined time corresponding to the proper period of the upper system. This characteristic allows the maximum minimization of the first peak of the acceleration of the upper system (mq).

Thus using mechanical coupling, in order to minimize the acceleration of the upper mass (mq), the sliding mode controller is the most effective.

3.3 Prospects of Real Device

On the real device, we can't add an actuator in parallel of the passive landing gear.

In order to guaranty the maximum of stability and to follow the control force u which will lead to an optimal transmitted force, we keep a passive hydraulic shock absorber that will dissipate the majority of the shock energy and in parallel of the passive shock absorber we add a controled throttling device that will dissipate the rest of the energy.

This device is a semi active device where only the damping coefficient of suspension will be modified. Such a device doesn't need a lot of energy and moreover in case of failure of the controller, the stability of the system is insured.

4 EXPERIMENTAL SET UP

We build a drop test bench in order to test free falls of the system. The drop test bench is composed of a static part and a mobile part. Two columns and a base make up the static part. The mobile part is composed of the quarter part of a vehicle (wheel, suspension, sprung mass (ms) and unsprung mass (mns)) and the upper system (mass mq, springs).

The stiffness of the suspension is insured by two parallel linear springs. Damping is insured by a hydraulic shock absorber. Four tuning parameters on it, allow us to modify its characteristic damping curve, in order to differentiate the damping rate in domains of low and high speeds for phases of compression and extension.

The used wheel is a wheel of an industrial vehicle. This one has been selected because its capacity to support heavy loads. A ball-bearing runner insures the guide of the mobile and leads the shock to be purely vertical.

Here frequential similitudes between a real aircraft and each subsystem of the drop test bench have been made. Drop test bench is represented on the following figure:

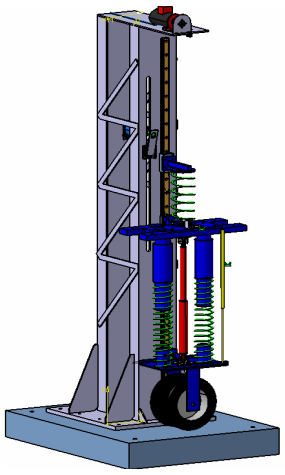


Figure 5: Numerical mock-up of drop test bench.

Accelerometers on each mass insure the knowledge of accelerations. Speeds and displacements are determined by numerical integrations. A force transducer between the shock absorber and the sprung mass (ms) measures the force transmitted by the damper. A linear inductive displacement transducer gives the stroke of the suspension. Thus redundancy of data on stroke of the suspension is insured.

Several tests in different configurations have been realized and have allowed us to cross check the previous exposed model. Cross checking leads us to have an accurate numerical model that allows us to develop the controled device. In following of the study, this one will be test on the drop test bench.

5 CONCLUSION

In this paper, a study of the induced vibrations by a high speed shock on a complex mechanical system has been presented. Different anti vibratory methods of control have been designed from a cross checked numerical model which has been previously exposed. Cross checking results from an experimental study that has been realized on a drop test bench. Using mechanical coupling, a sliding mode controller has proved its efficacy in order to minimize the acceleration of an upper system located on an equivalent quarter part of vehicle system submitted to high speed shock.

Nevertheless this control force must be reachable by a dynamical tuning of the damping coefficient of the hydraulic shock absorber.

Thus in prospect, a semi active device has been designed and will have to be tested.

REFERENCES

Ghiringhelli, G.L. 2000. ''Testing of semi active landing gear control for a general aviation aircraft''. *Journal of aircraft vol. 37, No 4*. (July-August).

Giua, A.; M. Melas and C. Seatzu. 2004. ''Design of a control law for a semi active suspension system using a solenoid valve damper''. *Proceeding 2004 IEEE Conference on Control Applications*, Taipei, Taiwan. (Sept.).

Guglielmino, E. and K.A. Edge. 2004. ''A controlled friction damper for vehicle applications''. *Control Engineering Practice 12*, pp 431-443.

Kim, W.S.; W.S. Lee and J.H. Kim. 2003. ''Control of an Active Vehicle Suspension Using Electromagnetic Motor''. *ICCAS2003*, Gyeongju, Korea. (Oct. 22-25).

Kruger, W. 2000. ''*Integrated Design Process for the Development of Semi-Active Landing Gears for Transport Aircraft*''. Thesis Institut für Flugmechanik und Flugregelung der Universität Stuttgart. 122p.

Yagiz, N. 2004. ''Comparison and Evaluation of Different Control Strategies on a Full Vehicle Model with Passenger Seat using Sliding Modes''. *International Journal of Vehicle Design, vol. 34, No 2*, pp 168-182.

TIME-FREQUENCY REPRESENTATION OF INSTANTANEOUS FREQUENCY USING A KALMAN FILTER

Jindřich Liška and Eduard Janeček
Department of Cybernetics, University of West Bohemia in Pilsen, Univerzitní 8, Plzeň, Czech Republic
jinliska@kky.zcu.cz, janecek@kky.zcu.cz

Keywords: Instantaneous frequency, Kalman filter, time-frequency analysis, state estimation, Hilbert transform.

Abstract: In this paper, a new method for obtaining a time-frequency representation of instantaneous frequency is introduced. A Kalman filter serves for dissociation of signal into modes with well defined instantaneous frequency. A second order resonator model is used as a model of signal components – 'monocomponent functions'. Simultaneously, the Kalman filter estimates the time-varying signal components in a complex form. The initial parameters for Kalman filter are obtained from the estimation of the spectral density through the Burg's algorithm by fitting an auto-regressive prediction model to the signal. To illustrate the performance of the proposed method, experimental results show the contribution of this method to improve the time-frequency resolution.

1 INTRODUCTION

Data analysis is a necessary part in pure research and in practical applications. The problem of estimating of a signal is of great interest in many areas of engineering, such as energy processing, speech recognition, vibration analysis and time series modeling. To analyze a non-stationary data, previous methods repeatedly apply block data processing such as the short-time Fourier transform, with the assumption, that the frequency characteristics are time-invariant (or that the process is stationary) for the duration of the time block. The resolution of such methods is limited by the Heisenberg-Gabor uncertainty principle.

In this work a different approach is proposed, in which a Kalman filter is used to decompose the time-varying signal into analytic components. As is well known, the Kalman-filter can estimate the state vectors of time-varying systems with knowledge of the stochastic characteristics of the measurement noise. The estimated components are then used for computation of instantaneous amplitude and frequency.

The rest of the paper is organized as follows. In Section 2, a summary of the common non-stationary data processing methods is presented. In Section 3, we mention the instantaneous frequency phenomenon. In Section 4, the use of Kalman filter to obtain complex signal component estimation is described. In Section 5, the results from experiments and from real application are discussed. Conclusions are drawn in Section 6.

2 NON-STATIONARY DATA PROCESSING METHODS

The spectrogram is the most basic method, which is a limited time window-width Fourier spectral analysis. Since it relies on the traditional Fourier transform, one has to assume the data to be piecewise stationary. There are also practical difficulties in applying the method: in order to localize an event in time, the window width must be narrow, but, on the other hand, the frequency resolution requires longer time series (uncertainty principle).

The wavelet approach is essentially a Fourier spectral analysis with an adjustable window. For specific applications, the basic wavelet function can be modified according to special needs, but the form has to be given before the analysis. In most common applications, the Morlet wavelet is defined as Gaussian enveloped sine and cosine wave groups with 5.5 waves. It is very useful in analysing data with gradual frequency changes. Difficulty of the wavelet analysis is among others its non-adaptive nature. Once the basic wavelet is selected then is used to analyse all the data.

The Wigner-Ville distribution is sometimes also referred to as the Heisenberg wavelet. By definition it is the Fourier transform of the central covariance function.

Above mentioned methods were used in Section 5 to compare their results with the output of the method based on Kalman estimation.

3 INSTANTANEOUS FREQUENCY AND THE COMPLEX SIGNAL

Instantaneous frequency, $\omega(t)$, is often defined as derivation of phase

$$\omega(t) = \frac{d\varphi(t)}{dt} = 2\pi f(t) \tag{1}$$

One of the ways how the unknown phase can be obtained is to introduce a complex signal $z(t)$ which corresponds to the real signal. As mentioned in (Hahn, 1996) or in (Huang, 1998), the Hilbert transform can be the elegant solution of this problem.

The Hilbert transform, $v(t)$, of a real signal $u(t)$ of the continuous variable t is

$$v(t) = \frac{1}{\pi} P\int_{-\infty}^{\infty} \frac{u(\eta)}{\eta - t} d\eta \tag{2}$$

where P indicates the Cauchy Principle Value integral. The complex signal $z(t)$

$$z(t) = u(t) + j \cdot v(t) = a(t)e^{j\varphi(t)} \tag{3}$$

whose imaginary part is the Hilbert transform $v(t)$ of the real part $u(t)$ is then called the analytical signal and its spectrum is composed only of the positive frequencies of the real signal $u(t)$.

From the complex signal, an instantaneous frequency and amplitude can be obtained for every value of t. Following (Hahn, 1996) the instantaneous amplitude is defined as

$$a(t) = \sqrt{u(t)^2 + v(t)^2} \tag{4}$$

and the instantaneous phase can be defined as

$$\varphi(t) = \arctan\frac{v(t)}{u(t)} \tag{5}$$

The instantaneous frequency then simplifies to

$$\omega(t) = \frac{d}{dt}(\arctan\frac{v(t)}{u(t)}) = \frac{u(t)\dot{v}(t) - v(t)\dot{u}(t)}{u(t)^2 + v(t)^2} \tag{6}$$

Even with the Hilbert transform, there is still considerable controversy in defining the instantaneous frequency as in (Boashash, 1992a). Applying the Hilbert transform directly to a multicomponent signal provides values of $a(t)$ and $\omega(t)$ which are unusable for describing the signal. The idea of instantaneous frequency and amplitude does not make sense when a signal consists of multiple components at different frequencies. This leads Cohen in (Cohen, 1995)to introduce term 'monocomponent function' where at any given time, there is only one frequency value. Huang (Huang, 1998) introduced a so called Empirical Mode Decomposition method to decompose the signal into monocomponent functions (Intrinsic Mode Functions).

4 USE OF KALMAN FILTER TO OBTAIN THE SIGNAL COMPONENTS

In this paper, an adaptive Kalman filter based approach is used to decompose the analyzed signal into monocomponent functions. As mentioned above, it is required that the estimated components are complex functions because of efficient computation of the instantaneous frequency. The analyzed signal is modeled as a sum of resonators in this study.

4.1 Complex Signal Component Model

The second-order model ($n = 2$) of auto-regressive (AR), linear time-invariant (LTI) system is considered as a resonator. Its external description in continuous domain is defined by the following differential equation

$$y(t) = a \cdot \sin(\omega \cdot t) \tag{7}$$

$$\dot{y}(t) = a \cdot \omega \cdot \cos(\omega \cdot t) \tag{8}$$

$$\ddot{y}(t) = -a \cdot \omega^2 \sin(\omega \cdot t) = -\omega^2 \cdot y(t) \tag{9}$$

where a is the amplitude and ω is the natural frequency of the resonator. Let the measured system is described by its state equations:

$$\dot{x}(t) = A \cdot x(t) \tag{10}$$

$$y(t) = C \cdot x(t) \tag{11}$$

where $x(t)$ denotes the vector of system internal states ($u(t)$ and $v(t)$) at time t, $y(t)$ is the output signal, A is the state matrix and C is the output matrix. Hence it follows that the internal model representation of the resonator with suitable selected state variables ($u(t) = \sin(\omega \cdot t)$ and $v(t) = \cos(\omega \cdot t)$) is then

$$\begin{bmatrix} \dot{u}(t) \\ \dot{v}(t) \end{bmatrix} = \begin{bmatrix} 0 & -\omega \\ \omega & 0 \end{bmatrix} \cdot \begin{bmatrix} u(t) \\ v(t) \end{bmatrix} \tag{12}$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} u(t) \\ v(t) \end{bmatrix}. \tag{13}$$

The state equation (12) shows the state matrix of the continuous model as a 2D rotation matrix whose eigenvalues are pure imaginary numbers. The trajectory in state space of such a system is a circle.
There is need to discretize the continuous state space model for a digital computation needs. This can be done by solving the state differential equation (14) and substitution of the time t with sampling step h (see Fairman, 1998)

$$x(t) = e^{At} x(0). \tag{14}$$

The discretized state model ($\Delta t = h$) with state noise $\xi(k)$ and output noise $\eta(k)$ is then

$$\begin{bmatrix} u(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} \cos(h \cdot \omega) & -\sin(h \cdot \omega) \\ \sin(h \cdot \omega) & \cos(h \cdot \omega) \end{bmatrix} \cdot \cdot \begin{bmatrix} u(k) \\ v(k) \end{bmatrix} + \Gamma \cdot \xi(k) \tag{15}$$

$$y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} u(k) \\ v(k) \end{bmatrix} + \Delta \cdot \eta(k) \tag{16}$$

The variables $\xi(k)$ and $\eta(k)$ are white noise vectors with identity covariance matrices. The specific features of the noises are characterized by the covariance matrices Γ and Δ.
This resonator model forms together with Kalman filtering approach an estimator of complex signal. The estimation of the first model state is a real part (sine function) and the estimation of the second model state is an imaginary part (cosine function) of the complex signal.

4.2 Discrete Kalman Filter

A discrete-time Kalman filter realizes a statistical estimation of the internal states of noisy linear system and it is able to reject uncorrelated measurement noise – a property shared by all Kalman filters. Let's assume a system with more components. Then the state matrix consists of following blocks:

$$A_i = \begin{bmatrix} \cos(h \cdot \omega_i) & \sin(h \cdot \omega_i) \\ -\sin(h \cdot \omega_i) & \cos(h \cdot \omega_i) \end{bmatrix} \tag{17}$$

and the state noise matrix blocks may be defined as a derivative of the state matrix blocks:

$$\Gamma_i = \frac{\partial A_i}{\partial (h \cdot \omega_i)} = \begin{bmatrix} -\sin(h \cdot \omega_i) & \cos(h \cdot \omega_i) \\ -\cos(h \cdot \omega_i) & -\sin(h \cdot \omega_i) \end{bmatrix} \tag{18}$$

The derivation of state matrix blocks as an estimation of the state noise matrix was selected experimentally, because the derivation produces blocks also in the state noise matrix and the components relate to each other in the same manner as in the state matrix.
The state-variable representation of the whole system, which is characterized by the sum of resonators, is given by the following matrices:

$$A = \underbrace{\begin{bmatrix} A_1 & 0 & 0 & \dots & 0 \\ 0 & A_2 & 0 & \dots & 0 \\ 0 & 0 & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & 0 \\ 0 & 0 & \dots & 0 & A_n \end{bmatrix}}_{2n \times 2n};$$

$$C = \underbrace{[1 \quad 0 \quad 1 \quad 0 \quad \dots \quad 1 \quad 0]}_{1 \times 2n};$$

$$\Gamma = \underbrace{\begin{bmatrix} \Gamma_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & \Gamma_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & \ddots & & \vdots & \vdots \\ \vdots & \vdots & & \ddots & 0 & 0 \\ 0 & 0 & \dots & 0 & \Gamma_n & 0 \end{bmatrix}}_{2n \times (2n+1)};$$

$$\Delta = \underbrace{[0 \quad \dots \quad 0 \quad \delta]}_{1 \times (2n+1)}. \tag{19}$$

Commonly, the Kalman estimation includes two steps – prediction and correction phase. Let's assume that the state estimate μ_0 is known with an error variance P_0. An a priori value of the state at instant $k+1$ can be obtained as

$$\mu_{k+1} = A \cdot \mu_k \tag{20}$$

The measured value $y(k)$ is then used to update the state at instant k. The additive correction of the a priori estimated state at $k+1$ is according to (Vaseghi, 1987) proportional to the difference between the a priori output at instant k defined as $C \cdot \mu_k$ and the measured $y(k)$:

$$\mu_{k+1} = A \cdot \mu_k + K_k \cdot (y_k - (C \cdot \mu_k)) \tag{21}$$

where K_k is the Kalman gain which guarantees the minimal variance of the error $x_k - \mu_k$.

Also, at each step the variance $P(k+1)$ of the error of μ_{k+1} is calculated (see (Vaseghi, 1987)):

$$P_{k+1} = AP_kA^T + \Gamma\Gamma^T - K_k \cdot (CP_kA^T + \Delta\Gamma^T) \tag{22}$$

It is used for calculation of Kalman gain in the next step of the recursive calculation (correction phase):

$$K_k = (AP_kC^T + \Gamma\Delta^T) \cdot (CP_kC^T + \Delta\Delta^T)^{-1} \tag{23}$$

4.3 Estimation of Initial Parameters

The initial parameters for Kalman filter are obtained from the estimation of the spectral density by fitting an AR prediction model to the signal. The used estimation algorithm is known as Burg's method (Marple, 1987), which fits an AR linear prediction filter model of a specified order to the input signal by minimizing the arithmetic mean of the forward and backward prediction errors. The spectral density is then computed from the frequency response of the prediction filter. The AR filter parameters are constrained to satisfy the Levinson-Durbin recursion.

The initial Kalman filter parameters (frequencies of the resonators) are then obtained as local maxima of estimated spectral density which are greater than a predefined level. These values indicate significant frequencies in spectral density and determine the order of the model (see Section 3.2).

5 RESULTS

Within this work three test signals are analyzed. The first test signal s_1 (t) contains three harmonic components. $1kHz$ sampling rate was used and the signal was 1 second long ($N = 1000$ points). Signal was formed by sinus functions with oscillation frequencies f_1=10Hz, f_2=30Hz and f_3=50Hz. The amplitude was for all three components set to 10, but the second component was zero for the first 0.5 seconds. The output noise with mean $m = 0$ and variance $\sigma = 1$ was added to the simulation signal.

The initial parameters for Kalman filter were obtained through Burg's AR linear prediction filter of order 10 and the level for local maxima was determined as $\max > 1$. Under these conditions the initial frequencies ($n=3$) for Kalman estimator were obtained from Burg's spectral density. The initial conditions of Kalman estimator were set up in the following way: μ_0=[1…1], P_0=10^6.I, $\delta = 1$, where dim(μ_0)=$1 \times n$ and dim(P_0)= $2n \times 2n$.

To take a look at the convergence of the estimate, the comparison of the Hilbert approach and Kalman filter is considered. In figure 1 the complex signal of second component of the simulated signal is displayed. The results were obtained through Hilbert transform and through Kalman estimation. The disadvantage of the Hilbert transform is that it requires the pre-processing of the signal through some signal decomposition method. To decompose the signal into its components, the above introduced algorithm uses the model of sum of resonators and simultaneously the Kalman estimator is used to estimate the time progression of these components.

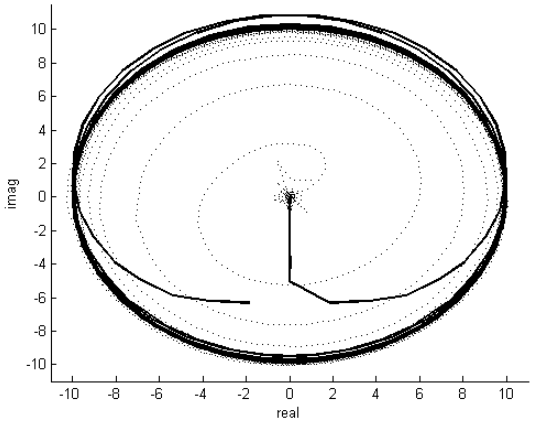


Figure 1: Second component of test signal s_1 (t) - complex signal obtained through Hilbert transform (solid line) and through Kalman estimation (dotted line).

In figure 2, there is shown the instantaneous frequency of all components of test signal s_1 (t).

The additive noise in simulation signal is the cause of the instantaneous frequency oscillating.

The algorithm based on Kalman estimation is also illustrated on another two non-stationary test signals s_2 (t) and s_3 (t). The initial conditions of Kalman estimator were set up as mentioned above and initial frequencies of the model were obtained through Burg's AR linear prediction filter of order 25 as maxima in estimated power spectrum (*n=10*).

The test signal s_2 (t) consists of two components in time-frequency domain - stationary harmonic signal with constant frequency and concave parabolic chirp signal. Both components exist in time between t = 100 and t = 900. The results of the Kalman estimation is compared with the methods mentioned in Section 2 and the results are shown in Figure 4. The output of Kalman estimation in time-frequency domain has relatively better time-frequency resolution in both components than the other methods.

The test signal s_3 (t) consists of four harmonic components and the accuracy of the method to identify the frequency and also the time of the origin and end of the components is tested. The signal begins again in time t = 100 and ends in t = 900. The frequency changes in t = 300 and t = 600. There are two components simultaneously in time between t = 300 and t = 600. The ability of methods to distinguish between these two frequencies is visible in Figure 5. The smoothed pseudo Wigner-Ville distribution and Kalman filter have better time-frequency resolution compared to short-time Fourier transform and to Morlet wavelet.

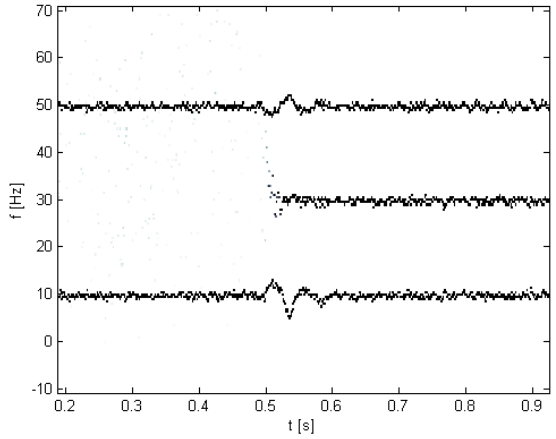


Figure 2: Estimation of instantaneous frequency of test signal components.

The last example is the transform of the acoustic signal from the real equipment where the instantaneous event took place. The signal was measured with 80 kHz sampling rate. For comparison, in figure 3, the time-frequency-amplitude responses of the short-time Fourier transform (STFT) and of the Kalman estimator approach are compared. The black column at first 4 milliseconds in the left spectrogram is the adaptation phase of Kalman filter. This example was obtained with following initial conditions: Burg's filter of order 400 was used to identify the power spectral density and all local maxima (*n=148*), which satisfy the inequality $\max > 10^{-7}$ were appointed as monitored frequencies. All resonance frequencies (5, 6, 14 and 27kHz) in the STFT spectrogram are also presented in the left one (Kalman). An event which occurs at time 0.022 seconds is displayed also in both spectrograms (see the frequency band 2 - 15 kHz). It is visible that the Kalman version of spectrogram offers a better resolution in time and frequency than the spectrogram obtained through STFT.

6 CONCLUSIONS

The new method for obtaining the time-frequency representation of instantaneous frequency has been introduced in this work. The procedure is based on the Kalman estimation and shares its advantages regarding the suppression of measurement noise. In this method the Kalman filter serves for dissociation of signal into modes with well defined instantaneous frequency. Simultaneously the time progression of signal components is estimated. This procedure utilizes the adaptive feature of the Kalman filter.
In cases where the short–time Fourier transform cannot offer sufficient resolution in frequency-time domain, there can be taken advantage of this method despite of higher computational severity. In vibrodiagnostic methods, where frequency-time information is used for localizing of non-stationary events, the sharpness of the introduced method can be helpful for the improvement of the event localization.

ACKNOWLEDGEMENTS

This work was supported by AREVA NP GmbH, Department SD-G in Erlangen (Germany) and from the specific research of Department of Cybernetics at the University of West Bohemia in Pilsen.

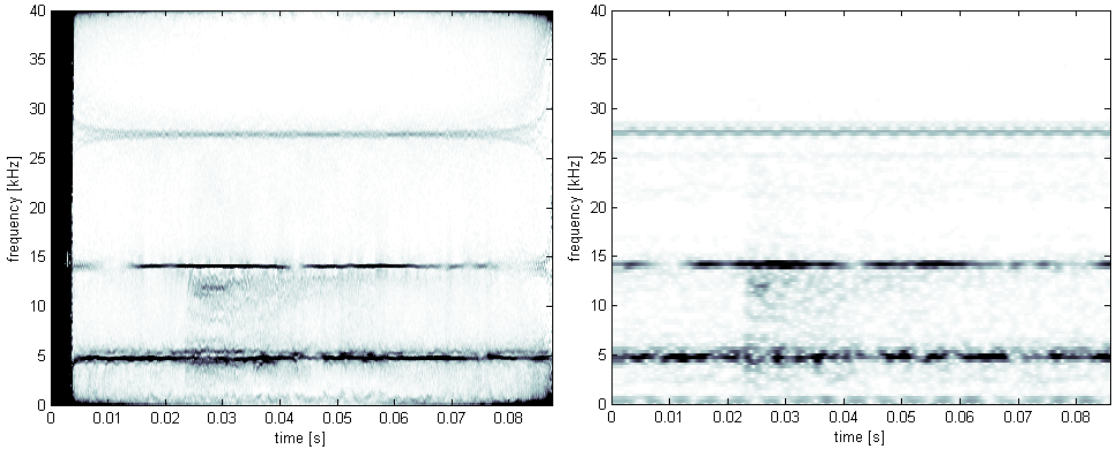


Figure 3: Spectrogram using Kalman estimation (left) and using short-time Fourier transform (right).

REFERENCES

Boashash, B., 1992a. Estimating and Interpreting the Instantaneous Frequency of a Signal – Part 1: Fundamentals. In *Proceedings of the IEEE*, vol. 80, no. 4, pp. 520-538.

Boashash, B., 1992b. Estimating and Interpreting the Instantaneous Frequency of a Signal – Part 2:Algorithms and Applications. In *Proceedings of the IEEE*, vol. 80, no. 4, pp. 540-568.

Cohen, L., 1995. *Time-Frequency Analysis*, Prentice Hall PTR. New Jersey.

Hahn, S.L., 1996. *Hilbert Transforms in Signal Processing*, Artech House. Boston.

Huang, N.E., et al., 1998. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. In *Proc.R.Soc.Lond. A,* vol. 454, no. 1971, pp. 903-995.

Huang, N. E., 2003. A confidence limit for the empirical mode decomposition and the Hilbert spectral analysis. In *Proc. Roy. Soc. Lond*., 459, pp. 2317-2345, 2003

Rilling, G., Flandrin, P., Goncalves, P., 2003. On Empirical Mode Decomposition and its Algorithms. In *IEEE-EURASHIP workshop on nonlinear signal an image processing NSIP-03*

Maragos, P., Kaiser, J.F., Quantieri, T.F., 1993. On Amplitude and Frequency Demodulation Using Energy Operators. In *IEEE Trans. on Signal Processing,* vol. 41, no. 4, pp. 1532-1550.

Marple, S.L., 1987. *Digital Spectral Analysis*, Prentice Hall. New Jersey.

Vaseghi, S.V., 2000. *Advanced Digital Signal Processing and Noise Reduction*, John Wiley & Sons. New Jersey.

Fairman, F.W., 1998. *Linear Control Theory: The State Space Approach*, John Wiley& Sons. Toronto

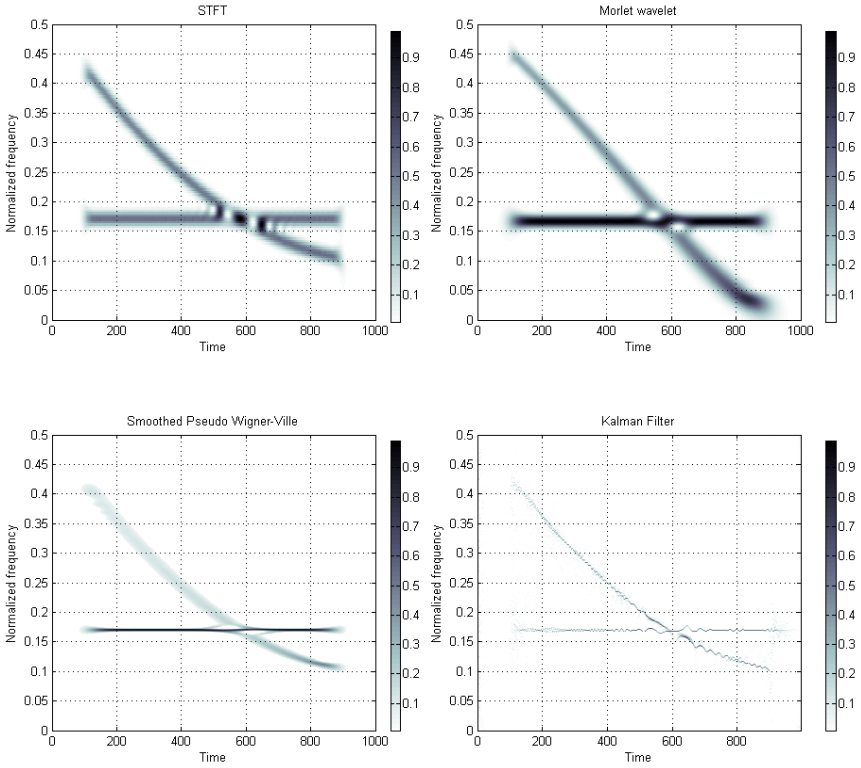


Figure 4: Time-frequency analysis results of the test signal $s_2(t)$.

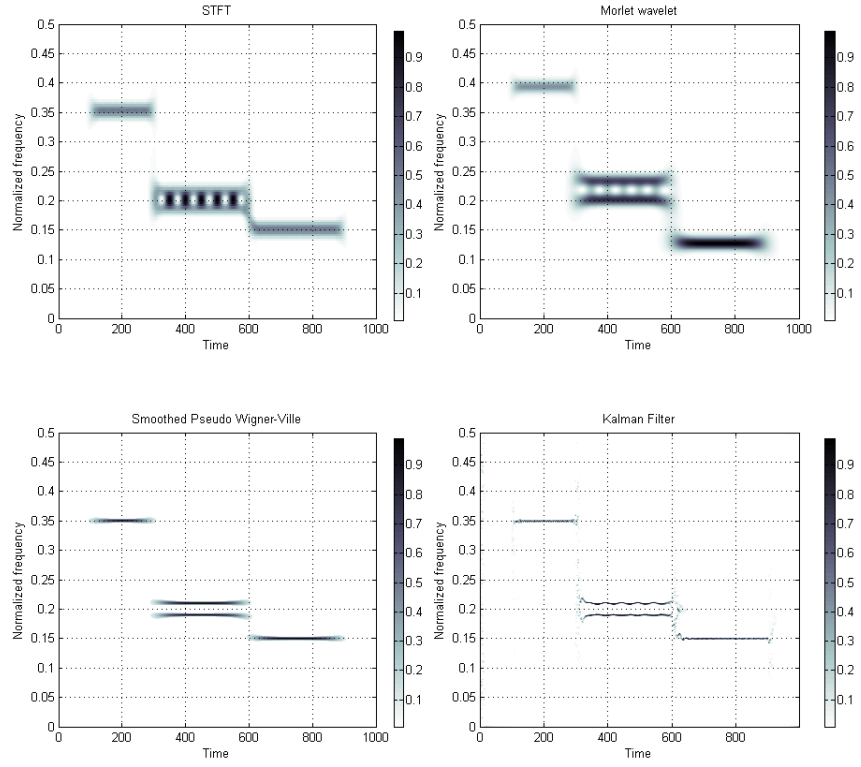


Figure 5: Time-frequency analysis results of the test signal $s_3(t)$.

AN INVESTIGATION OF EXTENDED KALMAN FILTERING IN THE ERRORS-IN-VARIABLES FRAMEWORK

A Joint State and Parameter Estimation Approach

Jens G. Linden, Benoit Vinsonneau and Keith J. Burnham
Control Theory and Applications Centre, Coventry University, Priory Street, Coventry, U.K.
j.linden@coventry.ac.uk, b.vinsonneau@coventry.ac.uk, k.burnham@coventry.ac.uk

Keywords: Errors-in-variables filtering, Kalman filtering, Parameter estimation.

Abstract: The paper addresses the problem of errors-in-variables filtering, i.e. the optimal estimation of inputs and outputs from noisy observations. While the optimal solution is known for linear time-varying systems of known parameterisation, this paper considers a suboptimal approach where only an approximated set of parameters is available. The proposed filter is derived by the means of joint state and parameter estimation using the extended Kalman filter approach which, in turn, leads to a coupled state-parameter estimation procedure. However, the resulting parameter estimates appear to be biased in the presence of input noise. The novel filter is compared with a previously proposed suboptimal filter.

1 INTRODUCTION

Kalman filtering (Anderson and Moore, 1979) deals with the optimal estimation of states and outputs in the presence of process and output noise. If an errors-in-variables (EIV) framework is adopted, i.e. the inputs are also affected by measurement noise, Kalman filtering cannot directly be applied (Guidorzi et al., 2003). The EIV filtering problem, which deals with the optimal estimation of noise free input and output signals, has been solved in (Guidorzi et al., 2003) and (Markovsky and De Moor, 2005). A unified framework for both, Kalman filtering and EIV filtering has been presented in (Diversi et al., 2005), where the EIV filtering problem is solved by the means of a standard Kalman filter (Kf) applied to a reformulated model. An EIV extended Kalman filter (EIVeKf), which is able to accommodate for model mismatch, in the case where the true system generating the data is unknown, has been presented in (Vinsonneau et al., 2005).

In this paper, the theory of extended Kalman filtering for joint state and parameter estimation (Ljung, 1979) is applied to the reformulated EIV model used in (Diversi et al., 2005). This leads to an algorithm which is shown to be similar to the EIVeKf. The differences and similarities between both approaches are discussed. Essentially, the filters calculate an estimate of the parameters of an assumed model and use this linear time-varying (LTV) model for filtering. It is revealed that these estimates are biased in the presence of input noise.

Section 2 reviews the extended Kalman filter for joint state and parameter estimation, while the existing EIV filtering techniques are summarised in Section 3. The modified algorithm for joint state and parameter estimation in the case of EIV, which is considered to be novel, is presented in Section 4, and an illustrative simulation example is given in Section 5. In Section 6, both EIV extended Kalman filters are compared and the results obtained from simulation are critically appraised . Finally, concluding remarks are given in Section 7.

2 EKF FOR JOINT STATE AND PARAMETER ESTIMATION

Assuming the data is generated by a linear time-invariant (LTI) discrete-time state-space system, its corresponding model may be given by

$$x_{k+1} = A(\theta)x_k + B(\theta)u_k + v_k \quad (1)$$

$$y_k = C(\theta)x_k + D(\theta)u_k + e_k \quad (2)$$

where x_k denotes the state, u_k the input, y_k the output, v_k process noise, e_k measurement noise and the model matrices $A(\theta)$, $B(\theta)$, $C(\theta)$ and $D(\theta)$ of appropriate dimension are characterised by the parameter vector θ. The noise sequences $\{v_k\}$ and $\{e_k\}$ are assumed to be independent with zero mean and covariance matrices

$$\Sigma_v = E\left[v_k v_{k-\tau}^T\right]\delta(\tau) \quad (3)$$

$$\Sigma_e = E\left[e_k e_{k-\tau}^T\right]\delta(\tau) \quad (4)$$

$$\Sigma_{ve} = E\left[v_k e_{k-\tau}^T\right]\delta(\tau) \quad (5)$$

where $\delta(\tau)$ denotes the Kronecker delta function. Based on an extended Kalman filter (eKf) (Anderson and Moore, 1979) an adaptive estimator for the model parameters can be derived by extending the state with the time dependent parameter vector θ_k, which leads to the following nonlinear state equation

$$\begin{bmatrix} x_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} A(\theta)x_k + B(\theta)u_k \\ \theta_k \end{bmatrix} + \begin{bmatrix} v_k \\ d_k \end{bmatrix} \quad (6)$$

The noise term d_k with covariance matrix

$$\Sigma_d = E\left[d_k d_{k-\tau}^T\right]\delta(\tau) \quad (7)$$

allows for variations in the system parameters and is usually set to zero if time-invariance is assumed.

Defining for convenience

$$A_k = A(\hat{\theta}_k) \qquad B_k = B(\hat{\theta}_k)$$
$$C_k = C(\hat{\theta}_k) \qquad D_k = D(\hat{\theta}_k) \quad (8)$$

the eKf for joint state and parameter estimation (jeKf) is given by (Ljung, 1979)

$$\hat{x}_{k+1} = A_k\hat{x}_k + B_k u_k + K_k\left[y_k - C_k\hat{x}_k - D_k u_k\right] \quad (9)$$

$$\hat{\theta}_{k+1} = \hat{\theta}_k + L_k\left[y_k - C_k\hat{x}_k - D_k u_k\right] \quad (10)$$

where

$$K_k = [A_k P_{1_k} C_k^T + F_k P_{2_k}^T C_k^T + A_k P_{2_k} H_k^T + F_k P_{3_k} H_k^T + \Sigma_{ve}]S_k^{-1} \quad (11)$$

$$S_k = C_k P_{1_k} C_k^T + C_k P_{2_k} E_k^T + H_k P_{2_k}^T C_k^T + H_k P_{3_k} H_k^T + \Sigma_e \quad (12)$$

$$L_k = \left[P_{2_k}^T C_k^T + P_{3_k} H_k^T\right] S_k^{-1} \quad (13)$$

$$P_{1_{k+1}} = A_k P_{1_k} A_k^T + A_k P_{2_k} F_k^T + F_k P_{2_k}^T A_K^T + F_k P_{3_k} F_k^T - K_k S_k K_k^T + \Sigma_v \quad (14)$$

$$P_{2_{k+1}} = A_k P_{2_k} + F_k P_{3_k} - K_k S_k L_k^T \quad (15)$$

$$P_{3_{k+1}} = P_{3_k} - L_k S_k L_k^T + \Sigma_d \quad (16)$$

with the Jacobians being defined by

$$F_k = \frac{\partial}{\partial\theta}\left(A(\theta)\hat{x}_k + B(\theta)u_k\right)\Big|_{\theta=\hat{\theta}_k} \quad (17)$$

$$H_k = \frac{\partial}{\partial\theta}\left(C(\theta)\hat{x}_k + D(\theta)u_k\right)\Big|_{\theta=\hat{\theta}_k} \quad (18)$$

It is shown in (Ljung, 1979) that the above recursive parameter estimator can be interpreted as an attempt to minimise the expected value of squared residuals associated with a constant model θ, i.e. minimising the cost function

$$V(\theta) = E\,|\bar{\varepsilon}_k(\theta)|^2 \quad (19)$$

where $\bar{\varepsilon}_k(\theta)$ is the innovation. Hence, this estimator is closely related to a recursive prediction error method (Ljung, 1999). A convergence analysis of this parameter estimator for linear systems is also carried out in (Ljung, 1979) and it is shown that it can produce biased estimates or even diverge. However, the above procedure can be modified to become a stochastic descent-algorithm which is globally convergent by including an approximation of

$$\left[\frac{\partial}{\partial\theta}\bar{K}(\theta)\right]\bar{\varepsilon}_k \quad (20)$$

into the Jacobian F_k (referred to as the coupling term (Ljung, 1979)), where $\bar{K}(\theta)$ is the steady-state Kalman gain. One way to ensure this is to assume an innovation model structure

$$x_{k+1} = A(\theta)x_k + B(\theta)u_k + K(\theta)\varepsilon_k \quad (21)$$

$$y_k = C(\theta)x_k + D(\theta)u_k + \varepsilon_k \quad (22)$$

rather than (1)-(2) and include all elements of the Kalman gain K into the parameter vector θ. Parametrising K and Σ_ε explicitly leads to a modified algorithm given by

$$\hat{x}_{k+1} = A_k\hat{x}_k + B_k u_k + K_k\varepsilon_k \quad (23)$$

$$\hat{\theta}_{k+1} = \hat{\theta}_k + L_k\varepsilon_k \quad (24)$$

where

$$\varepsilon_k = y_k - C_k\hat{x}_k - D_k u_k \quad (25)$$

$$L_k = \left[P_{2_k}^T C_k^T + P_{3_k} H_k^T\right]\Sigma_{\varepsilon_k}^{-1} \quad (26)$$

$$+ F_k P_{3_k} F_k^T - K_k\Sigma_{\varepsilon_k}K_k^T + \Sigma_v \quad (27)$$

$$P_{2_{k+1}} = A_k P_{2_k} + F_k P_{3_k} - K_k\Sigma_{\varepsilon_k}L_k^T \quad (28)$$

$$P_{3_{k+1}} = P_{3_k} - L_k\Sigma_{\varepsilon_k}L_k^T + \Sigma_d \quad (29)$$

$$\Sigma_{\varepsilon_k} = \Sigma_{\varepsilon_{k-1}} + \frac{1}{k}\left(\varepsilon_k\varepsilon_k^T - \Sigma_{\varepsilon_{k-1}}\right) \quad (30)$$

and

$$F_k = \frac{\partial}{\partial\theta}\left(A(\theta)\hat{x}_k + B(\theta)u_k + K(\theta)\varepsilon_k\right)\Big|_{\theta=\hat{\theta}_k} \quad (31)$$

$$K_k = K(\hat{\theta}_k) \quad (32)$$

Moreover, a projection facility has to be utilised to ensure that $\hat{\theta}_k$ lies in the compact subset

$$D_s = \{\theta | A(\theta) - K(\theta)C(\theta) \textit{ is exponentially stable.}\} \quad (33)$$

In practice, a step-size reduction might also be necessary to achieve convergence.

3 KALMAN AND EIV FILTERING

Whereas traditional Kalman filtering (Anderson and Moore, 1979) addresses the problem of estimating the optimal states and outputs in the case of process and output noise, EIV filtering deals with the optimal estimation of inputs and outputs, where both quantities are considered to be observations, which are affected by additive noise.

The EIV filtering problem for the LTI case has been solved in (Guidorzi et al., 2003), where the optimal input and output estimates are determined based on the state-space model

$$\xi_{k+1} = \mathcal{A}\xi_k + \mathcal{B}\begin{bmatrix} y_k^T & u_k^T \end{bmatrix}^T \tag{34}$$

$$\gamma_k = \mathcal{C}\xi_k + \mathcal{D}\begin{bmatrix} y_k^T & u_k^T \end{bmatrix}^T \tag{35}$$

where ξ_k denotes the state, $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\mathcal{D}$ the model matrices and γ_k the residuals. A different approach has been presented in (Markovsky and De Moor, 2005), where the EIV state-space representation is reformulated, such that the new state-space model depends on the measured quantities u_k and y_k with redefined process and measurement noise terms. Subsequently, a modified Kalman filter can be applied to obtain an estimate of the system states, which, in turn, allows the estimation of the true input and output signals. Utilising the latter reformulation of the EIV state-space system, a unified context for both traditional Kalman filtering and EIV filtering has been proposed (Diversi et al., 2005) and this is outlined in the following Subsection.

3.1 Unified Framework for Kalman and EIV Filtering

Consider the discrete-time LTI EIV state-space model given by

$$x_{k+1} = Ax_k + Bu_{0_k} + w_k \tag{36}$$

$$y_{0_k} = Cx_k + Du_{0_k} \tag{37}$$

$$u_k = u_{0_k} + \tilde{u}_k \tag{38}$$

$$y_k = y_{0_k} + \tilde{y}_k \tag{39}$$

where x_k denotes the state, u_{0_k} and y_{0_k} the unknown inputs and outputs, u_k and y_k the noisy measurements. The noise terms $\tilde{u}_k$, $\tilde{y}_k$ and w_k denote input, output and process noise, respectively, which are assumed to be of zero mean and with covariance matrices

$$E\left[w_k w_{k-\tau}^T\right] = \Sigma_w \delta(\tau) \tag{40}$$

$$E\left[\tilde{u}_k \tilde{u}_{k-\tau}^T\right] = \Sigma_{\tilde{u}} \delta(\tau) \tag{41}$$

$$E\left[\tilde{y}_k \tilde{y}_{k-\tau}^T\right] = \Sigma_{\tilde{y}} \delta(\tau) \tag{42}$$

$$E\left[\tilde{u}_k \tilde{y}_{k-\tau}^T\right] = \Sigma_{\tilde{u}\tilde{y}} \delta(\tau) \tag{43}$$

$$E\left[w_k \tilde{u}_{k-\tau}^T\right] = 0 \tag{44}$$

$$E\left[w_k \tilde{y}_{k-\tau}^T\right] = 0 \tag{45}$$

The model equations (36)-(39) can be rewritten as

$$x_{k+1} = Ax_k + Bu_k + v_k \tag{46}$$

$$z_k = Cx_k + e_k \tag{47}$$

where z_k, v_k and e_k are the redefined measurements, process noise and measurement noise, respectively, which are given by

$$z_k = y_k - Du_k \tag{48}$$

and

$$v_k = w_k - B\tilde{u}_k \tag{49}$$

$$e_k = \tilde{y}_k - D\tilde{u}_k \tag{50}$$

The covariance matrices are readily obtained via

$$\Sigma_v = \Sigma_w + B\Sigma_{\tilde{u}} B^T \tag{51}$$

$$\Sigma_e = \Sigma_{\tilde{y}} - \Sigma_{\tilde{u}\tilde{y}}^T D^T - D\Sigma_{\tilde{u}\tilde{y}} + D\Sigma_{\tilde{u}} D^T \tag{52}$$

$$\Sigma_{ve} = B\left[\Sigma_{\tilde{u}} D^T - \Sigma_{\tilde{u}\tilde{y}}\right] \tag{53}$$

A standard Kalman filter is then utilised to determine the optimal state estimate

$$\hat{x}_{k+1|k} = Ax_{k|k-1} + Bu_k + K_k \varepsilon_k \tag{54}$$

$$K_k = \left[AP_{k|k-1}C^T + \Sigma_{ve}\right]\Sigma_\varepsilon^{-1} \tag{55}$$

$$P_{k+1|k} = AP_{k|k-1}A^T + \Sigma_v - \left[AP_{k|k-1}C^T + \Sigma_{ve}\right] \times \Sigma_\varepsilon^{1-}\left[AP_{k|k-1}C^T + \Sigma_{ve}\right]^T \tag{56}$$

where

$$\varepsilon_k = z_k - C\hat{x}_{k|k-1} = C\left(x_k - \hat{x}_{k|k-1}\right) + e_k \tag{57}$$

$$\Sigma_\varepsilon = E\left[\varepsilon_k \varepsilon_k^T\right] = CP_{k|k-1}C^T + \Sigma_e \tag{58}$$

are the innovations and its corresponding covariance matrix. The filtered inputs and outputs are then given by (Diversi et al., 2005)

$$\hat{u}_{0_k} = u_k - E\left[\tilde{u}_k | z_k\right] = u_k - \left[\Sigma_{\tilde{u}\tilde{y}} - \Sigma_{\tilde{u}} D^T\right]\Sigma_\varepsilon^{-1}\varepsilon_k \tag{59}$$

$$\hat{y}_{0_k} = y_k - E\left[\tilde{y}_k | z_k\right] = y_k - \left[\Sigma_{\tilde{y}} - \Sigma_{\tilde{u}\tilde{y}}^T D^T\right]\Sigma_\varepsilon^{-1}\varepsilon_k \tag{60}$$

Hence, a traditional Kalman filter can be utilised to achieve both, the optimal estimation of states and input/output sequences.

3.2 Extended EIV Kalman Filtering

A drawback of the linear filter described in Subsection 3.1, is that it relies on exact information of the noise characteristics and an exact model of the process generating the data. In an attempt to compensate for the latter requirement, an extended EIV Kalman filter (EIVeKf), based on the EIV Kalman filter given in (Guidorzi et al., 2003), has been proposed in (Vinsonneau et al., 2005). Instead of an exact process representation, the EIVeKf requires only an approximate parametrisation of a linear default model, characterised by θ_d, to achieve acceptable results. Under certain conditions, use of the EIVeKf can lead to a superior filter performance with respect to the linear counterpart in cases where the system parametrisation is only approximately known. Moreover, the EIVeKf is also able to accommodate, to a certain degree, the case of LTV systems.

The idea of the EIVeKf is very similar to the jeKf approach; the state vector is augmented with the compensating parameters θ_c such that the new state vector becomes

$$\begin{bmatrix} \xi_k \\ \theta_{c_k} \end{bmatrix} \tag{61}$$

where ξ_k is the original state vector in (34) for the calculation of the residual sequence $\{\gamma_k\}$. The resulting system equations are thus nonlinear and the EIVKf filter can be modified using first order Taylor approximations for the predicting step, which results to the EIVeKf equations.

4 EIV EXTENDED KALMAN FILTER FOR JOINT PARAMETER ESTIMATION

Since the EIV filtering problem can be solved by the means of a standard Kalman filter, as outlined in Section 3.1, one could apply well known modifications of traditional Kalman filtering techniques to estimate u_{0_k} and y_{0_k}. The approach proposed here is to apply the idea of joint state and parameter estimation, as summarised in Section 2, to EIV systems. This is expected lead to a similar filter as the one presented in (Vinsonneau et al., 2005) with the difference that the estimate $\hat{\theta}_k$ is not only used for the prediction step, but in the overall filter equations. In addition, the changes in the parameters can be tracked by the means of Σ_d defined in (7).

4.1 Algorithm

Assuming the data is generated by a Eiv system of structure (36)-(39) and the assumed model structure is given by

$$x_{k+1} = Ax_k + Bu_k + v_k \tag{62}$$

$$y_k = Cx_k + Du_k + e_k \tag{63}$$

with v_k and e_k as defined in (49)-(53). Then the EIV extended Kalman filter for joint parameter estimation (EIVjeKf) is readily given by (23)-(30) together with the estimated inputs and outputs as defined in (59) and (60), whereas A, B, C, D are replaced by A_k, B_k, C_k, D_k, respectively.

However, it it found in simulations, that this form of the EIVjeKf can suffer from outliers in terms of overall EIV filter performance as illustrated in Section 5. Therefore, a slight different formulation will be preferred in the subsequent: while the Kalman gain is still to be estimated, in order to assure the existence of the terms $\left[\frac{\partial}{\partial\theta}K(\theta)\right]\big|_{\theta=\hat{\theta}_k}\bar{\varepsilon}_k$ within F_k, these estimates are not further utilised in the algorithm but rather K_k as given by (11)-(16). For clarification, the algorithm is summarised as follows.

Algorithm 4.1 *Assuming an EIV system of the form* (36)-(39) *and the model structure of* (62)-(63) *with noise characteristics* (49)-(53)*, the EIVjeKf is given by*

1. *Augment the state vector x_k with the model parameters θ_k and Kalman gain K_k*
2. *Determine the time-varying model matrices A_k, B_k, C_k and D_k as given in* (8)
3. *Compute the innovation given by* (25) *and its covariance matrix as defined in* (30)
4. *Determine the Jacobians F_k and H_k as given in* (31) *and* (18)*, respectively*
5. *Determine the reformulated covariance matrices* (51)-(53) *with B and D replaced by B_k and D_k*
6. *Compute $\hat{x}_{k+1}$ and $\hat{\theta}_{k+1}$ using* (9)-(16)
7. *Determine $\hat{u}_{0_k}$ and $\hat{u}_{0_k}$ given by* (59) *and* (60)*, where D is replaced with D_k*
8. *Increment k and continue with step 2*

Remark 4.1 *As outlined in Section 2, the parameter estimator resulting from the EIVjeKf can be interpreted as an recursive prediction error method with the correction inspired by the eKf algorithm. However, it is known, that the application of standard prediction error methods to EIV systems does not yield consistent estimates as demonstrated in (Söderström, 1981). Therefore, the estimated $\hat{\theta}_m$ is expected to be biased with respect to the true parametrisation.*

5 SIMULATION

Consider the single-input single-output LTV state-space system given by (36)-(39) with

$$A = \begin{bmatrix} 0 & 0.1 \\ -0.2 & 0.3 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
$$C = \begin{bmatrix} 0.9 & b_k - 1.35 \end{bmatrix} \qquad D = -4.5 \qquad (64)$$

and

$$\Sigma_{\tilde{u}} = 0.2 \quad \Sigma_{\tilde{y}} = 5 \quad \Sigma_{\tilde{u}\tilde{y}} = 0.8 \quad \Sigma_w = 0 \qquad (65)$$

where the time-varying parameter b_k with mean value $E[b_k] = 4.7$ slowly varies as illustrated in Figure 1. The input u_{0_k} is a zero mean white noise process with unity variance, the signal-to-noise ratios (SNR) are given by

$$\mathrm{SNR}_u = 10\log\left(\frac{\mathrm{var}(u_0)}{\mathrm{var}(\tilde{u})}\right) = 16.0 \qquad (66)$$

$$\mathrm{SNR}_y = 10\log\left(\frac{\mathrm{var}(y_0)}{\mathrm{var}(\tilde{y})}\right) = 19.7 \qquad (67)$$

whereas u_0, $\tilde{u}$, y_0, $\tilde{y}$ without index denote the sequences to the corresponding signals, i.e. $\tilde{u} = \{\tilde{u}_k\}_{k=1}^N$ and so forth. The number of samples is set to $N = 5000$. While the covariance matrices (65) are assumed to be known, the system parametrisation is approximated by the default parameter

$$\theta_d = \begin{bmatrix} -0.5 & 0.3 & -3.1 & 4.1 \end{bmatrix}^T \qquad (68)$$

while θ_k is given by

$$\theta_k = \begin{bmatrix} -0.3 & 0.2 & -4.5 & b_k \end{bmatrix}^T \qquad (69)$$

In order to model the variation in the system parameters, the covariance matrix (7) corresponding to θ_k is chosen to be

$$\Sigma_d = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 1} \\ 0_{1\times 3} & 1\cdot 10^{-3} \end{bmatrix} \qquad (70)$$

where $0_{m\times n}$ denotes the $m \times n$ zero matrix. The performance index of interest is the EIV filter performance, i.e 'how much' noise can be removed from the noisy observations u_k and y_k. This can be quantified by

$$P_u = 100\frac{||u_0 - u||_2 - ||u_0 - \hat{u}_0||_2}{||u_0 - u||_2} \qquad (71)$$

$$P_y = 100\frac{||y_0 - y||_2 - ||y_0 - \hat{y}_0||_2}{||y_0 - y||_2} \qquad (72)$$

giving a relative measure for the removed noise in percentage, i.e. a value of 100 indicates a perfect filtering (estimate and true signal are identical), while a value of 0 corresponds to no filtering performance (estimate and noisy signal are identical). All simulations are verified by the means of 100 Monte-Carlo runs.

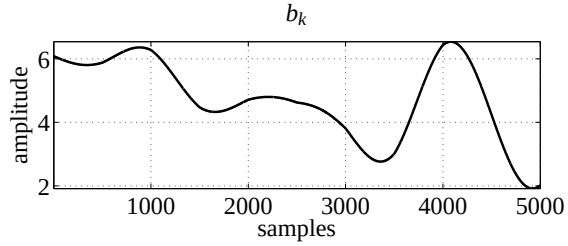


Figure 1: Time-varying parameter b_k.

5.1 Results

The three filters are applied to the above simulation, that is, the Kf presented in Section 3.1, the EIVeKf of (Vinsonneau et al., 2005) and the new approach, the EIVjeKf discussed in the previous Section. The mean and variances of P_u and P_y for the different Monte-Carlo runs are summarised in Table 1. It is

Table 1: Filter performance for the different filters.

	Kf	EIVeKf	EIVjeKf
$E[P_u]$	25.2	33.7	42.9
$E[P_y]$	26.1	38.5	50.0
$\mathrm{var}(P_u)$	0.9	1.4	2.1
$\mathrm{var}(P_y)$	0.8	0.8	2.5

observed that the Kf exhibits the worst EIV filter performance by removing only around 25% and 26% of input and output noise, respectively, while the best performance is achieved applying the EIVjeKf, which removes on average approximately 43% and 50% of the noise contamination. In contrast, the variances of the performance indices with respect to the Monte-Carlo simulation are smaller in the case of the Kf. The results of the EIVeKf lie in between of the other two filters for both, mean and variance of the filter performance.

6 DISCUSSION

Since only a default time-invariant model (and not the true system parametrisation) is available to the Kf, a negative impact on the filter performance is not surprising. If the true LTV system parametrisation, and hence, the true time-varying covariance matrices (51)-(53) are known, the Kf would be optimal and may well outperform the eKf approaches considered

here. In fact, the only reason that the nonlinear approaches yield superior performance is that they attempt to compensate for the parameter-mismatch by estimating the the model parameters, which are then used for filtering (at least in-part).

The different performance results of the EIVeKf and the EIVjeKf can be explained by regarding the estimate of b_k, which is shown in Figure 2. Since the EIVjeKf models the variations of b_k by means of Σ_d, it is able, to a certain degree, to track b_k, while the EIVeKf uses no adaptivity to estimate θ_k. However, it would be a straightforward step to include Σ_d into the EIVeKf algorithm. In such a case, both estimates become nearly identical and the corresponding filter performance ($E[P_u] = 41.8$ and $E[P_y] = 48.6$) is very similar to the results of the EIVjeKf (cf. Table 1). The remaining differences may be explained by the fact that while the prediction phase of the EIVeKf utilises the estimates of θ_k, the default parameter set θ_d is still used for the correction.

Another point to be observed in Figure 2 is that the estimate for b_k produced by the EIVjeKf is biased. This was expected, as mentioned in Remark 4.1, since the parameter estimator resulting from the eKf approach is not adjusted for the EIV case. Hence, the EIV filter performance of the nonlinear approaches can deteriorate if the bias is large with respect to the model mismatch characterised by θ_d. In such a situation, the EIVeKf is expected to perform better than the EIVjeKf, since the latter relies completely on $\hat{\theta}_k$. This can be verified by modifying the above simulation and increasing the input noise to $\Sigma_{\tilde{u}} = 1$, which corresponds to $\mathrm{SNR}_u = -0.1$. The filter performance[1] is given in Table 2 and the time-varying parameter b_k and its estimates are shown in Figure 3. It can be observed, that the estimate produced by the EIVjeKf becomes more biased as $\Sigma_{\tilde{u}}$ increases resulting in a decreased filter performance, while the estimate given by the EIVeKf is less biased, hence, by opposition, yielding a better filter performance.

[1]Note, that in this case, Σ_d is incorporated into the EIVeKf algorithm.

Table 2: Filter performance for the different filters for the case $\Sigma_{\tilde{u}} = 1$.

	Kf	EIVeKf	EIVjeKf
$E[P_u]$	39.5	36.1	16.7
$E[P_y]$	18.4	24.9	20.0
$\mathrm{var}(P_u)$	0.8	0.7	3.7
$\mathrm{var}(P_y)$	0.8	0.8	5.2

7 CONCLUSIONS

The solution of the EIV filtering problem as a special case of traditional Kalman filtering in extended noise environments (Diversi et al., 2005) has been reviewed. Since the optimal estimation of noise-free inputs and outputs can be achieved by applying the well known Kalman filter to a reformulated model, a joint state and parameter estimation procedure via extended Kalman filtering (Ljung, 1979) is investigated for the EIV case. The resulting algorithm is very similar to the approach presented in (Vinsonneau et al., 2005). In fact, these nonlinear EIV filter approaches attempt to estimate the model parameters by means of a recursive prediction error method. In turn, this means that these estimates are generally biased in the presence of input noise and this may be considered as the main limitation of these approaches. The difference between both nonlinear filters is that the EIVeKf in (Vinsonneau et al., 2005) uses the estimated model parameters only for the prediction phase of the filter, and whilst more investigation may be necessary, it appears to lead to more robustness if the SNR of the input is low.

Some potentially interesting further work would aim to investigate other suboptimal filters, again with

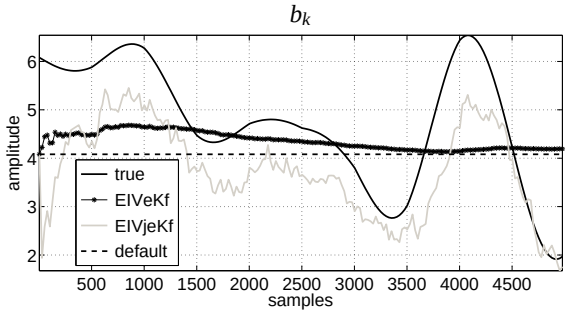


Figure 2: Time-varying parameter b_k, its estimates and the default value.

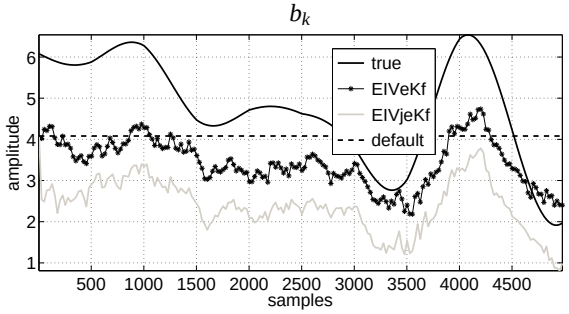


Figure 3: Time-varying parameter b_k, its estimates and the default value for the case $\Sigma_{\tilde{u}} = 1$.

coupled state and parameter estimation, but where the parameter set is obtained via a recursive EIV identification technique.

REFERENCES

Anderson, B. D. O. and Moore, J. B. (1979). *Optimal Filtering*. Prentice-Hall, Englewood Cliffs, New Jersey.

Diversi, R., Guidorzi, R., and Soverini, U. (2005). Kalman filtering in extended noise environments. *IEEE Trans. Autom. Contr.*, 50:1396–1402.

Guidorzi, R., Diversi, R., and Soverini, U. (2003). Optimal errors-in-variables filtering. *Automatica*, 39:281–289.

Ljung, L. (1979). Asymptotic behavior of the extended Kalman filter as a parameter estimator for linear systems. *IEEE Trans. on Automatic Control*, 24(1):36–50.

Ljung, L. (1999). *System Identification - Theory for the user*. PTR Prentice Hall Infromation and System Sciences Series. Prentice Hall, 2nd edition.

Markovsky, I. and De Moor, B. (2005). Linear dynamic filtering with noisy input and output. *Automatica*, 41:167–171.

Söderström, T. (1981). Identification of stochastic linear systems in presence of input noise. *Automatica*, 17:713–725.

Vinsonneau, B., Goodall, D. P., and Burnham, K. J. (2005). Errors-in-variables extended Kalman filter. In *Proc. IAR & ACD Conf.*, pages 217–222, Mulhouse, France.

A STATE ESTIMATOR FOR NONLINEAR STOCHASTIC SYSTEMS BASED ON DIRAC MIXTURE APPROXIMATIONS

Oliver C. Schrempf and Uwe D. Hanebeck
Intelligent Sensor-Actuator-Systems Laboratory, Universität Karlsruhe (TH), Germany
schrempf@ieee.org, uwe.hanebeck@ieee.org

Keywords: Nonlinear Dynamic Systems, Stochastic Filter, Dirac Mixture.

Abstract: This paper presents a filter approach for estimating the state of nonlinear dynamic systems based on recursive approximation of posterior densities by means of Dirac mixture functions. The filter consists of a prediction step and a filter step. The approximation approach is based on a systematic minimization of a distance measure and is hence optimal and deterministic. In contrast to non-deterministic methods we are able to determine the optimal number of components in the Dirac mixture. A further benefit of the proposed approach is the consideration of measurements during the approximation process in order to avoid parameter degradation.

NOTATION

k	time index
x_k	state variable
y_k	measurement variable
$\hat{y}_k$	actual measurement at time k
$\hat{x}_k$	point estimate at time k
$\tilde{f}^x(x)$	probability density function of $\boldsymbol{x}$
$f^x(x)$	approximation of $\tilde{f}^x(x)$
$f^p(x_{k+1})$	predicted density function
$f^e(x_k)$	filtered density function
$f^L(x_k, \hat{y}_k)$	Likelihood functon
$\delta(x)$	Dirac Delta function
$H(x)$	Heaviside step function
G	distance measure
$\underline{\eta}$	parameter vector
γ	progression parameter
$\mathcal{N}(.,m,\sigma)$	Gaussian density with mean m and standard deviation σ

1 INTRODUCTION

In this paper, we present a novel stochastic filter for nonlinear dynamic systems suffering from system as well as measurement noise. The uncertainty in the filter's estimate caused by the noise is described by means of probability density functions. The problem that arises with the application of stochastic filters to nonlinear systems is that the complexity of the density representation increases and the exact densities cannot be calculated directly in general. Common solutions to this problem in order to build practical estimators can be devided into two classes. The approaches of the first class approximate or modify the system and measurement functions and apply a filter. The idea of the second class is to approximate the resulting density functions themselves in order to calculate the filter steps in closed-form .

A common representative of the first class is the extended Kalman filter (EKF). It is based on linearization of the system and measurement functions and applying a standard Kalman filter to this modified system. This approach is applicable to systems with only negligible nonlinearities and additive noise, but fails in more general cases.

Another approach is to approximate the system together with its noise as a probabilistic model (Huber and Hanebeck, 2007). The application of adequate representations of the model like Gaussian mixtures with axis-aligned components (Huber et al., 2006), allows for efficient implementation of the filter steps.

Filters approximating the density functions instead of the system function can be divided into two

main approaches found in the literature: i) sample-based representations and ii) analytic density representations.

Sample-based filters like the popular particle filter (Doucet et al., 2000)(Doucet et al., 2001) apply Monte Carlo methods for obtaining a sample representation. Since these sample are usually produced by a random number generator, the resulting estimate is not deterministic. Furthermore, Markov Chain Monte Carlo Methods (MCMC) are iterative algorithms that are unsuited for recursive estimation, hence, importance sampling like in (Geweke, 1989) is often applied. The problem of sample degradation is usually tackled by bootstrap methods (Gordon, 1993).

Other methods describe the probability density functions by means of their moments. A popular approach is the so called Unscented Kalman filter (UKF) (Julier and Uhlmann, 1997) that uses the first moment and the second central moment for representing the densities. This allows for an efficient calculation of the update but fails in representing highly complex densities arising from nonlinear systems. Furthermore, the assumption of jointly Gaussian distributed states and measurements is made, which is not valid in general.

An approach that represents the densities by means of Gaussian mixture densities is the so called Gaussian sum filter (Alspach and Sorenson, 1972). The Gaussian mixture representation allows for approximating arbitrary density functions, but finding the appropriate parameters is a tough problem. A more recent approach is the Progressive Bayes filter (Hanebeck et al., 2003) which uses a distance measure for approximating the true densities. The key idea in this approach is to transform the approximation problem into an optimization problem. This is a major motivation for the approximation applied in the approach presented here.

The filter method we propose here follows the idea of approximating the density functions instead of the system itself, but the approximation is performed in a systematic manner. The general idea is to approximate the continuous density function by means of a Dirac mixture function that minimizes a certain distance measure to the true density. The approximation process itself is described in (Schrempf et al., 2006a)(Schrempf et al., 2006b) and will therefore only be discussed briefly in this paper. We will focus here on the complete filter consisting of approximation, prediction (Schrempf and Hanebeck, 2007) and filter step.

Since we make use of a distance measure, we are able to quantify the quality of our approximation. Furthermore, it is possible to find an optimal number of components required for sufficient estimates. Following this idea we will extend our optimization method to a full estimation cycle by considering the measurement as well.

The paper is organized as follows: We will give a problem formulation in Section 2 followed by an overview of the complete filter in Section 3. The building blocks of the filter are described in Section 4 whereas Section 5 presents further optimization methods. Experimental results comparing the proposed filter to state-of-the-art filters are given in Section 6 followed by conclusions in Section 7.

2 PROBLEM FORMULATION

We consider discrete-time nonlinear dynamic systems according to

$$x_{k+1} = a_k(x_k, u_k, w_k)$$

producing measurements according to the nonlinear function

$$y_k = h_k(x_k, v_k) \quad .$$

The state of the system that is not directly observable in general is represented by x_k. u_k is a known input, and y_k is an observable output of the system. $a_k(\cdot)$ is a time-varying nonlinear mapping describing the system's dynamic behavior. w_k represents both endogenous and exogenous noise sources acting upon the system and is described by means of a density function $f_k^w(w_k)$. $h_k(\cdot)$ maps the system state to an output value which suffers from noise v_k which is modeled by means of a density function $f_k^v(v_k)$.

Starting with an initial state x_0, our goal is to keep track of the system's state over time while maintaining a full continuous stochastic representation of the uncertainty involved caused by the system and measurement noise.

This corresponds to sequentially calculating the state densities $f_k^x(x_k), k = 1,\ldots,N$, by means of a prediction and a filter step where the system and measurement functions are applied.

Exact computation of these densities, however, is not feasible, as the complexity of the density increases in every step. In addition, the resulting densities cannot be calculated in an analytic form in general.

The aim of this work is to provide a density representation that approximates the true density, in order to allow for closed-form calculation of the prediction step while maintaining a predefined quality of the approximation with respect to a given distance measure.

For reasons of brevity, we omit the input u_k. We further focus on additive noise, which results in the system equation

$$x_{k+1} = g_k(x_k) + w_k$$

and a measurement equation

$$y_k = h_k(x_k) + v_k \ .$$

In addition, the time index k is omitted in some cases without notice.

3 FILTER OUTLINE

In this section, we will give a brief overview of the recursive filtering scheme depicted as a block diagram in Figure 1. The left part of the figure shows the nonlinear system suffering from additive noise as described in Sec. 2. The right part shows the estimator. The input of the estimator is a measurement $\hat{y}_k$ coming from the system. The output of the estimator is a probability density function $f^e(x_k)$ from which a point estimate $\hat{x}_k$ can be derived. The estimator itself works recursively as can be seen from the loop in the diagram. Each recursion consists of a prediction step, an approximation step, and a filter step.

The prediction step receives a density $f^e(x_k)$ from the previous filter step. This density is an approximation represented by means of a Dirac mixture allowing for an analytically exact solution of the Bayesian prediction integral with respect to this approximation. The prediction yields a continuous mixture density representation (e.g. a Gaussian mixture) $\tilde{f}^p(x_{k+1})$. Details are given in Sec. 4.2.

The continuous mixture density $\tilde{f}^p(x_{k+1})$ resulting from the prediction step serves as input to the approximation step. The density is systematically approximated by means of a Dirac mixture $f^p(x_{k+1})$ minimizing a distance measure $G\left(\tilde{f}^p(x_{k+1}), f^p(x_{k+1})\right)$ as described in Sec. 4.1.

The approximated density $f^p(x_{k+1})$ is then fed to the filter step, where it gets fused with the likelihood function $f^L(x,\hat{y})$. This step is described in detail in Sec. 4.3.

4 FILTER COMPONENTS

4.1 Density Approximation

We will now introduce Dirac mixture functions and explain how they can be interpreted as parametric density functions. Subsequently, we briefly describe the systematic approximation scheme.

4.1.1 Dirac Mixture Density Representation

Dirac mixtures are a sum of weighted Dirac delta functions according to

$$f(x,\underline{\eta}) = \sum_{i=1}^{L} w_i \delta(x - x_i) \ , \qquad (1)$$

where

$$\underline{\eta} = [x_1, x_2, \ldots, x_L, w_1, w_2, \ldots, w_L]^T$$

is a parameter vector consisting of locations $x_i, i = 1,\ldots,L$ and weighting coefficients $w_i, i = 1,\ldots,L$. The Dirac delta function is an impulse representation with the properties

$$\delta(x) = \begin{cases} 0, & x \neq 0 \\ \text{not defined}, & x = 0 \end{cases}$$

and

$$\int_{\mathbf{R}} \delta(x)\, dx = 1 \ .$$

This results in the fundamental property

$$\int_{-\infty}^{\infty} f(x)\delta(x - x_i)\, dx = f(x_i) \ .$$

A mixture of Dirac delta functions as given in (1) can be used for representing arbitrary density functions if the following requirements are considered. Since the properties of a density function $f(x)$ demand that $f(x) \geq 0$ and $\int_{\mathbf{R}} f(x)\, dx = 1$, we have

$$w_i \geq 0, i = 1,\ldots,L$$

and

$$\sum_{i=1}^{L} w_i = 1 \ .$$

Hence, we require $2L$ parameters with $2L-1$ degrees of freedom.

A simplified density representation is given by equally weighted Dirac mixtures, as

$$f(x,\underline{\eta}) = \frac{1}{L} \sum_{i=1}^{L} \delta(x - x_i) \ ,$$

where only L parameters and L degrees of freedom are used. This results in a simpler, less memory consuming representation with less approximation capabilities.

Dirac mixtures are a generic density representation useful for approximating complicated densities arising in estimators for nonlinear dynamic systems.

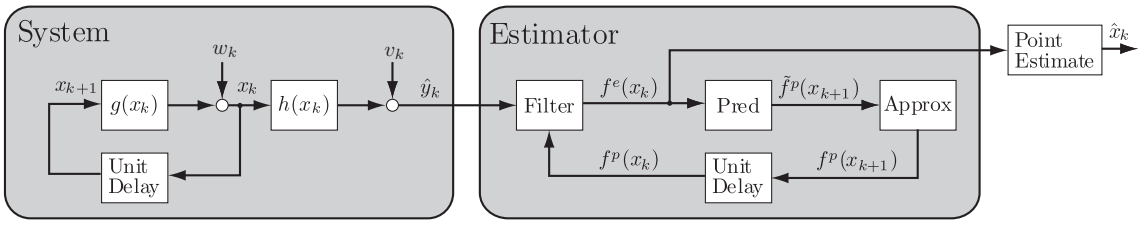


Figure 1: A block diagram of the recursive estimator. The estimator consists of a filter step, a prediction step and an approximation step.

4.1.2 Approximation Approach

A systematic approximation of continuous density by means of another density requires a distance measure between the two densities

$$G\left(\tilde{f}^p(x_{k+1}), f^p(x_{k+1}, \underline{\eta})\right) \quad ,$$

where $\tilde{f}^p(\cdot)$ is an arbitrary continuous density function and $f^p(\cdot, \underline{\eta})$ is a Dirac mixture density. The approximation problem can then be reformulated as an optimization problem by finding a parameter vector $\underline{\eta}$ that minimizes this distance measure.

Popular distance measures for comparing continuous densities, measures are the Kullback–Leibler divergence (Kullback and Leibler, 1951) or the integral quadratic measure. For comparing a continuous density to a Dirac mixture, however, they are not very useful, since the Dirac mixture is zero between the Dirac pulses. Hence, instead of comparing the densities directly, the corresponding (cumulative) distribution functions are employed for that purpose. For the rest of this subsection we will omit the time index k and the p index in order to keep the formulae comprehensible.

The distribution function corresponding to the true density $\tilde{f}(x)$ is given by

$$\tilde{F}(x) = \int_{-\infty}^{x} \tilde{f}(t)\, dt \quad .$$

The distribution function corresponding to the Dirac mixture approximation can be written as

$$F(x, \underline{\eta}) = \int_{-\infty}^{x} f(t, \underline{\eta})\, dt = \sum_{i=1}^{L} w_i H(x - x_i) \quad , \qquad (2)$$

where $H(.)$ denotes the Heaviside function defined as

$$H(x) = \begin{cases} 0, & x < 0 \\ \frac{1}{2}, & x = 0 \\ 1, & x > 0 \end{cases} \quad .$$

A suitable distance measure is given by the weighted Cramér–von Mises distance (Boos, 1981)

$$G(\underline{\eta}) = \int_{-\infty}^{\infty} r(x) \left(\tilde{F}(x) - F(x, \underline{\eta})\right)^2 dx \quad , \qquad (3)$$

where $r(x)$ is a nonnegative weighting function. $r(x)$ will later in the filter step be selected in such a way that only those portions of the predicted probability density function having support by the likelihood, are approximated with high accuracy. This avoids to put much approximation effort into irrelevant regions of the state space.

The goal is now to find a parameter vector $\underline{\eta}$ that minimizes (3) according to $\underline{\eta} = \arg\min_{\underline{\eta}} G(\underline{\eta})$. Unfortunately, it is not possible to solve this optimization problem directly. Hence, we apply a progressive method introduced in (Schrempf et al., 2006b). For this approach, we introduce a so called progression parameter γ into $\tilde{F}(x)$ that goes from $0 \ldots 1$. The purpose of this parameter is to find a very simple and exact approximation of $\tilde{F}(x, \gamma)$ for $\gamma = 0$. Further we must guarantee that $\tilde{F}(x, \gamma = 1) = \tilde{F}(x)$. By varying γ from 0 to 1 we track the parameter vector $\underline{\eta}$ that minimizes the distance measure.

In order to find the minimum of the distance measure, we have to find the root of the partial derivative with respect to $\underline{\eta}$ according to

$$\frac{\partial G(\underline{\eta}, \gamma)}{\partial \underline{\eta}} = \begin{bmatrix} \frac{\partial G(\underline{\eta}, \gamma)}{\partial \underline{x}} \\ \frac{\partial G(\underline{\eta}, \gamma)}{\partial \underline{w}} \end{bmatrix} \stackrel{!}{=} \underline{0} \quad . \qquad (4)$$

Together with (2) and (3) this results in the system of equations

$$\begin{aligned} \tilde{F}(x_i, \gamma) &= \sum_{j=1}^{L} w_j H(x_i - x_j) \quad , \\ \int_{x_i}^{\infty} r(x) \tilde{F}(x, \gamma)\, dx &= \sum_{j=1}^{L} w_j \int_{x_i}^{\infty} r(x) H(x - x_j)\, dx \quad , \end{aligned}$$

where $i = 1, \ldots, L$.

In order to track the minimum of the distance measure we have to take the derivative of (4) with respect to γ.

This results in a system of ordinary first order differential equations that can be written in a vector–matrix–form as

$$\underline{b} = \mathbf{P}\, \dot{\underline{\eta}} \quad , \qquad (5)$$

where

$$\underline{b} = \begin{bmatrix} \frac{\partial \tilde{F}(x_1,\gamma)}{\partial \gamma} \\ \vdots \\ \frac{\partial \tilde{F}(x_L,\gamma)}{\partial \gamma} \\ \int_{x_0}^{\infty} \frac{\partial \tilde{F}(x,\gamma)}{\partial \gamma}\, dx \\ \int_{x_1}^{\infty} \frac{\partial \tilde{F}(x,\gamma)}{\partial \gamma}\, dx \\ \vdots \\ \int_{x_L}^{\infty} \frac{\partial \tilde{F}(x,\gamma)}{\partial \gamma}\, dx \end{bmatrix}$$

and

$$\dot{\underline{\eta}} = \frac{\partial \underline{\eta}}{\partial \gamma} = \left[\dot{x}_1, \ldots, \dot{x}_L, \dot{w}_0, \dot{w}_1, \ldots, \dot{w}_L\right]^T \quad .$$

$\dot{\underline{\eta}}$ denotes the derivative of $\underline{\eta}$ with respect to γ.

The **P** matrix as well as the derivations are given in (Schrempf et al., 2006b). The approximation of $\tilde{f}(x)$ now boils down to solving (5).

4.2 Prediction Step

We now explain the Bayesian prediction step and show how the approximation introduced in the last subsection can be used for closed-form calculations.

Calculating the state densities $f^p(x_{k+1})$, $k = 1, \ldots, N$, is performed by evaluation the Bayesian forward step, which is given by

$$f^p(x_{k+1}) = \int_{-\infty}^{\infty} f(x_{k+1}|x_k) f^e(x_k)\, dx_k \quad , \qquad (6)$$

where the transition density $f(x_{k+1}|x_k)$ of the considered nonlinear system with additive noise is given by

$$f(x_{k+1}|x_k) = f^w(x_{k+1} - g(x_k)) \quad ,$$

where $f^w(\cdot)$ is the density of the system noise (e.g. Gaussian).

In general, the integral involved in (6) cannot be solved analytically for arbitrary prior densities $f^e(x_k)$ For a given input point $\bar{x}_k$, however, represented by the Dirac delta function $f^e(x_k) = \delta(x_k - \bar{x}_k)$, (6) can be solved in closed form according to

$$f_p(x_{k+1}) = f^w(x_{k+1} - g(\bar{x}_k)) \quad .$$

In the case of zero mean Gaussian system noise with

$$f^w(w) = \mathcal{N}(w, 0, \sigma^w) \quad ,$$

this yields

$$f_p(x_{k+1}) = \mathcal{N}(x_{k+1}, g(\bar{x}_k), \sigma^w) \quad ,$$

which is a Gaussian Density with a standard deviation σ^w.

For a given Dirac mixture prior $f^e(x_k)$ according to (1) given by

$$f^e(x_k) = \sum_{i=1}^{L} w_k^{(i)} \delta(x_k - x_k^{(i)}) \quad , \qquad (7)$$

the posterior according to (6) is a Gaussian mixture given by

$$f^p(x_{k+1}) = \sum_{i=1}^{L} w_k^{(i)} \mathcal{N}\left(x_{k+1}, g(x_k^{(i)}), \sigma^w\right) \quad ,$$

which is a closed-form solution.

Please note, that similar result can be derived for non-additive and non-Gaussian noise.

4.3 Filter Step

The filter step consists of fusing the predicted density $f^p(x_k)$ and the likelihood function $f^L(x_k, \hat{y}_k)$ governed by the measurement $\hat{y}_k$ according to

$$f^e(x_k) = c \cdot f^p(x_k) \cdot f^L(x_k, \hat{y}_k) \quad , \qquad (8)$$

where c is a normalizing constant. The likelihood function is given by

$$f^L(x_k, \hat{y}_k) = f(\hat{y}_k|x_k) \quad .$$

For a nonlinear system with additive noise, the conditional density for the measurement $f(y_k|x_k)$ is given by

$$f(y_k|x_k) = f^v(y_k - h(x_k)) \quad ,$$

where $f^v(\cdot)$ is the density of the measurement noise and $h(x_k)$ is the nonlinear measurement function. In the case of zero-mean Gaussian measurement noise the likelihood function can be written as

$$f^L(x_k, \hat{y}_k) = \mathcal{N}(\hat{y}_k, h(x_k), \sigma^v) \quad .$$

We would like to emphasize, that in the general nonlinear case this likelihood function is no proper density function. Furthermore, a parametric representation of this function is not available in general. This is the reason, why the update equation (8) cannot be solved analytically, even if the prediction is given in a parametric representation.

Our solution to this problem is driven by the same observation made for solving the prediction step in Sec. 4.2. The likelihood can be evaluated at certain points $\bar{x}_k$, which yields constant values.

In order to calculate the product of a likelihood and a prediction, where the latter is already given as a Dirac mixture, it comes quite naturally to use the $x_k^{(i)}$ points of the Diracs to evaluate the likelihood. The obtained values of $f^L(\cdot)$ can then be used to reweight the predicted density according to

$$f^e(x_k) = \sum_{i=1}^{L} \bar{w}_k^{(i)} \delta(x_k - x_k^{(i)})$$

with

$$\bar{w}_k^{(i)} = c \cdot w_k^{(i)} \cdot f^v(\hat{y}_k - h(x_k^{(i)})) \ ,$$

where $w_k^{(i)}$ is the i'th weight and $x_k^{(i)}$ is the i'th position of the approximated prediction $f^p(x_k)$. The normalization constant can be calculated as

$$c = \left(\sum_{i=1}^{L} w_k^{(i)} \cdot f^v(\hat{y}_k - h(x_k^{(i)})) \right)^{-1} \ .$$

Naive approximation of the predicted density in a fixed interval may lead to many small weights, since not all regions of the state space supported by the prediction are as well supported by the likelihood. This phenomenon can be described as parameter degradation. To circumvent this problem, we make use of the weighting function $r(x)$ in (3). Details on this approach are presented in the next section.

5 OPTIMAL NUMBER OF PARAMETERS

In this section, we describe how to tackle the problem of parameter degradation that is inherent to all filter approaches considering only discrete points of the density. We further describe a method for finding an optimal number of components for the approximation taking into account the prediction and filter steps as well.

To fight the problem of parameter degradation described in the previous section we make use of the fact, that although the likelihood function is not a density it decreases to zero for $x \to \pm\infty$ in many cases. An example for this are polynomial systems suffering from additive noise. Therefore, we can define an area of support for which the likelihood is higher than a certain value. This area of support is an interval and can be represented by the weighting function $r(x)$ in (3). It guarantees, that all components of the approximation are located in this interval and are therefor not reweighed to zero in the filter step. In other words, the complete mass of the approximation function accounts for the main area of interest.

In (Schrempf and Hanebeck, 2007), we introduced an algorithm for finding the optimal number of components required for the approximation with respect to the following prediction step. We will now extend this algorithm in order to account for the preceding filter step as well.

At the beginning of Algorithm 1 in line 6, an initial approximation with a very large number of components is generated and passed through the prediction step, resulting in a continuous density representation with parameter vector $\underline{\kappa}_t$. Due to the high number of components we can assume this density to be very close to the true density. An efficient procedure for approximating arbitrary mixture densities with Dirac mixtures comprising a large number of components is given in (Schrempf and Hanebeck, 2007).

Algorithm 1 Optimal number of components w.r.t. the filter step and posterior density.

```
 1: Select max. Error Threshold G_max
 2: Select initial number of Components L = L_0
 3: Select search step ΔL
 4: f^L(x) = likelihood(ŷ)
 5: r(x) = support(f^L(x))
 6: κ_t = predict(filter(approx(L_large,r(x)),ŷ))
 7: while G > Gmax do
 8:     κ = predict(filter(approx(L,r(x)),ŷ))
 9:     G = G(κ_t, κ)
10:     L = L + ΔL
11: end while
12: L_l = L − 2ΔL
13: L_u = L − ΔL
14: while L_u − L_l > 1 do
15:     L_t = L_l + ⌊(L_u − L_l)/2⌋
16:     κ = predict(filter(approx(L_l,r(x)),ŷ))
17:     G = G(κ_t, κ)
18:     if G > G_max then
19:         L_l = L_t
20:     else
21:         L_u = L_t
22:     end if
23: end while
```

In each search step of the algorithm, the distance measure of the approximated density at hand to the density defined by $\underline{\kappa}_t$ is calculated. In this way the smallest number of components for a prespecified error can be found.

6 EXPERIMENTAL RESULTS

In order to compare the performance of our filter to other state-of-the-art filters, we have simulated a nonlinear dynamic system according to the left part of Figure 1. We apply the filter to a strongly nonlinear cubic system and measurement function motivated by the cubic sensor problem introduced in (Bucy, 1969).

The simulated system function is

$$g(x_k) = 2x_k - 0.5x_k^3 + w$$

and the additive noise is Gaussian with $\sigma^w = 0.2$ standard deviation. The measurement function is

$$h(x_k) = x_k - 0.5x_k^3 + v$$

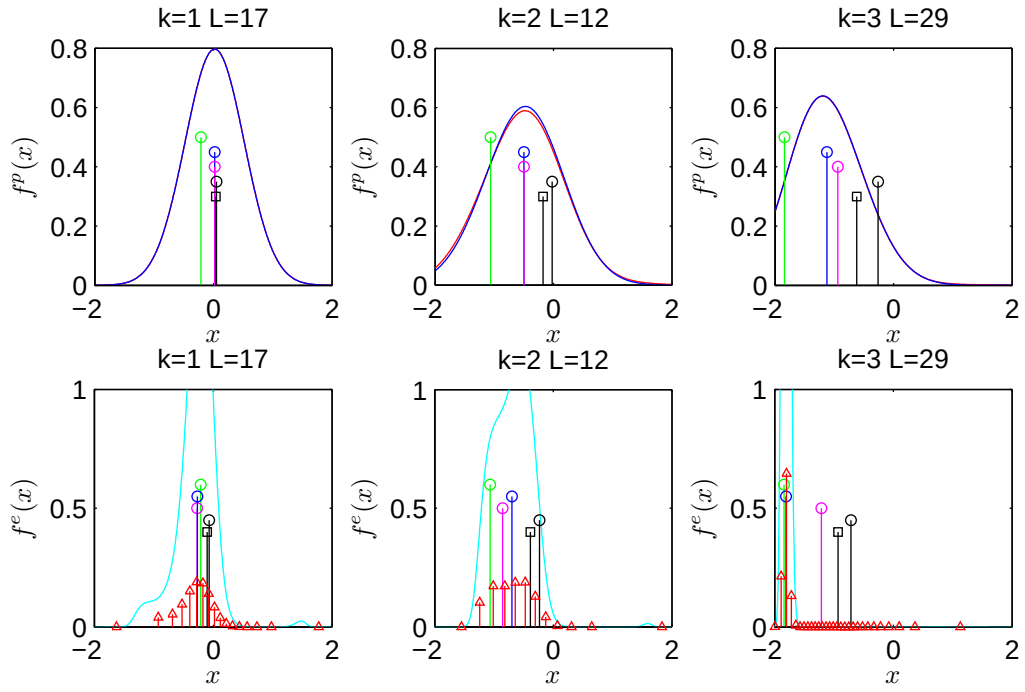


Figure 2: The recursive filter for $T = 3$ steps. k indicates the step number and L the number of components for the Dirac mixture. The upper row shows the prediction steps, the lower row shows the filter steps. **Upper row:** The blue is the continuous density predicted by the DM filter, the red line underneath is the true density. The green marker depicts the true (simulated) system state, the other markers depict the predicted point estimates of the following filters: blue=DM, pink=UKF, black_circle=PF20, black_square=PF1000. **Lower row:** The cyan line shows the likelihood. The colors of the point estimates are similar to the upper line.

with additive Gaussian noise and $\sigma^v = 0.5$.

The generated measurements are used as input to our filter as well as to an unscented Kalman filter and a particle filter. The particle filter is applied in a variant with 20 particles and a variant with 1000 particles in order to compare the performance.

In a first run of the experiment we show $T = 3$ steps in Figure 2. The upper row shows the prediction steps, the lower row shows the corresponding filter steps. The continuous prediction $f^p(x_{k+1})$ of the Dirac mixture (DM) filter is depicted by the dark blue line. The red line underneath shows the true prediction computed numerically as a reference. The cyan plot in the lower line shows the likelihood given by the current measurement and the red arrows depict the Dirac mixture after the filter step.

Both rows also show the point estimates of the various applied filters in the current step. The green marker indicates the true (simulated) state, whereas blue stands for the Dirac mixture point estimate. Pink is the UKF estimate an black are the particle filter estimates. The particle filter indicated by the circle uses 20 particles, the one indicated by the square uses 1000 particles.

We simulated the system a further 10 times for $T = 7s$ in order to calculate the root means square error e_{rms} of the 4 filters. The results are shown in Figure 3. The plot shows that the point estimates of the Dirac mixture filter are much closer to the true state than the point estimates of the other filters.

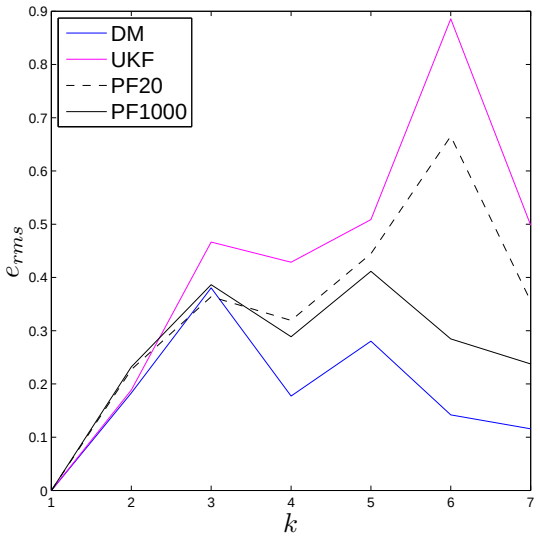


Figure 3: Root mean square error for 10 runs and $T = 7$ steps.

7 CONCLUSION

In this paper, we presented a complete Dirac mixture filter that is based on the approximation of the posterior density. The filter makes use of the properties of the Dirac mixture approximation wherever they are required, but does not deny the continuous character of the true density. This can especially be seen after each prediction step, where the full continuous density representation is used.

The new approach is natural, mathematically rigorous, and based on an efficient algorithms (Schrempf et al., 2006a)(Schrempf et al., 2006b) for the optimal approximation of arbitrary densities by Dirac mixtures with respect to a given distance.

Compared to a particle filter, the proposed method has several advantages. First, the Dirac components are systematically placed in order to minimize a given distance measure, which is selected in such a way that the future evolution of approximate densities is always close to the true density while also considering the actual measurements. As a result, very few samples are sufficient for achieving an excellent estimation quality. Second, the optimization does not only include the parameters of the Dirac mixture approximation, i.e., weights and locations, but also the number of components. As a result, the number of components is automatically adjusted according to the complexity of the underlying true distribution and the support area of a given likelihood. Third, as the approximation is fully deterministic, it guarantees repeatable results.

Compared to the Unscented Kalman Filter, the Dirac mixture filter has the advantage, that it is not restricted to first and second order moments. Hence, multi-modal densities, which cannot be described sufficiently by using only the first two moments, can be treated very efficiently. Such densties occur quite often in strongly nonlinear systems. Furthermore, no assumptions on the joint distribution of state and measurement have to be made.

REFERENCES

Alspach, D. L. and Sorenson, H. W. (1972). Nonlinear Bayesian Estimation Using Gaussian Sum Approximation. *IEEE Transactions on Automatic Control*, AC–17(4):439–448.

Boos, D. D. (1981). Minimum Distance Estimators for Location and Goodness of Fit. *Journal of the American Statistical association*, 76(375):663–670.

Bucy, R. S. (1969). Bayes Theorem and Digital Realizations for Non-Linear Filters. *Journal of Astronautical Sciences*, 17:80–94.

Doucet, A., Freitas, N. D., and Gordon, N. (2001). *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York.

Doucet, A., Godsill, S., and Andrieu, C. (2000). On Sequential Monte Carlo Sampling Methods for Bayesian Filtering. *Statistics and Computing*, 10(3):197–208.

Geweke, J. (1989). Bayesian Inference in Econometric Models using Monte Carlo Integration. *Econometrica*, 24:1317–1399.

Gordon, N. (1993). *Bayesian Methods for Tracking*. PhD thesis, University of London.

Hanebeck, U. D., Briechle, K., and Rauh, A. (2003). Progressive Bayes: A New Framework for Nonlinear State Estimation. In *Proceedings of SPIE*, volume 5099, pages 256–267, Orlando, Florida. AeroSense Symposium.

Huber, M., Brunn, D., and Hanebeck, U. D. (2006). Closed-Form Prediction of Nonlinear Dynamic Systems by Means of Gaussian Mixture Approximation of the Transition Density. In *International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2006), Heidelberg, Deutschland*, pages 98–103.

Huber, M. F. and Hanebeck, U. D. (2007). Hybrid Transition Density Approximation for Efficient Recursive Prediction of Nonlinear Dynamic Systems. In *International Conference on Information Processing in Sensor Networks (IPSN 2007), Cambridge, USA*.

Julier, S. and Uhlmann, J. (1997). A New Extension of the Kalman Filter to Nonlinear Systems. In *Proceedings of SPIE AeroSense, 11th International Symposium on Aerospace/Defense Sensing, Simulation, and Controls, Orlando, FL.*

Kullback, S. and Leibler, R. A. (1951). On Information and Sufficiency. *Annals of Mathematical Statistics*, 22(2):79–86.

Schrempf, O. C., Brunn, D., and Hanebeck, U. D. (2006a). Density Approximation Based on Dirac Mixtures with Regard to Nonlinear Estimation and Filtering. In *Proceedings of the 45th IEEE Conference on Decision and Control (CDC'06), San Diego, California, USA*.

Schrempf, O. C., Brunn, D., and Hanebeck, U. D. (2006b). Dirac Mixture Density Approximation Based on Minimization of the Weighted Cramér–von Mises Distance. In *Proceedings of the International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2006), Heidelberg, Germany*, pages 512–517.

Schrempf, O. C. and Hanebeck, U. D. (2007). Recursive Prediction of Stochastic Nonlinear Systems Based on Dirac Mixture Approximations. In *Proceedings of the American Control Conference (ACC '07), New York City, USA*.

A CLOSED-FORM MODEL PREDICTIVE CONTROL FRAMEWORK FOR NONLINEAR NOISE-CORRUPTED SYSTEMS

Florian Weissel, Marco F. Huber and Uwe D. Hanebeck
Intelligent Sensor-Actuator-Systems Laboratory, Universität Karlsruhe (TH), Germany
weissel@ieee.org, marco.huber@ieee.org, uwe.hanebeck@ieee.org

Keywords: Nonlinear Model Predictive Control; Stochastic Systems; Nonlinear Estimation.

Abstract: In this paper, a framework for Nonlinear Model Predictive Control (NMPC) that explicitly incorporates the noise influence on systems with continuous state spaces is introduced. By the incorporation of noise, which results from uncertainties during model identification and the measurement process, the quality of control can be significantly increased. Since NMPC requires the prediction of system states over a certain horizon, an efficient state prediction technique for nonlinear noise-affected systems is required. This is achieved by using transition densities approximated by axis-aligned Gaussian mixtures together with methods to reduce the computational burden. A versatile cost function representation also employing Gaussian mixtures provides an increased freedom of modeling. Combining the prediction technique with this value function representation allows closed-form calculation of the necessary optimization problems arising from NMPC. The capabilities of the framework and especially the benefits that can be gained by considering the noise in the controller are illustrated by the example of a mobile robot following a given path.

NOTATION

x	variable
$\boldsymbol{x}$	random variable
$\underline{\boldsymbol{x}}$	vector-valued random variable
$\tilde{f}^x(x)$	probability density function of $\boldsymbol{x}$
$f^x(x)$	approximate of $\tilde{f}^x(x)$
$\mathcal{N}(x-\mu;\sigma^2)$	Gaussian density with mean μ and standard deviation σ
$\mathrm{E}_{\boldsymbol{x}}\{\underline{\boldsymbol{x}}\}$	expected value of $\underline{\boldsymbol{x}}$
$J_k(\underline{\boldsymbol{x}}_k)$	value function
$V_k(\underline{\boldsymbol{x}}_k,\underline{u}_k)$	input dependent value function
$g_n(\underline{\boldsymbol{x}}_n,\underline{u}_n)$	cost function
k	time index
n	time index of prediction horizon

1 INTRODUCTION

Model Predictive Control (MPC), which is also referred to as Receding or Rolling Horizon Control, has become more and more important for control applications from various fields. This is due to the fact that not only the current system state, but also a model-based prediction of future system states over a finite N stage prediction horizon is considered in the control law. For this prediction horizon, an open-loop optimal control problem with a corresponding value function is solved. Then the resulting optimal control input is applied as a closed-loop control to the system.

The well understood and widely used MPC for linear system models (Qin and Badgewell, 1997) together with linear or quadratic cost functions is not always sufficient if it is necessary to achieve even higher quality control, e.g. in high precision robot control or in the process industry. Steadily growing requirements on the control quality can be met by incorporating nonlinear system models and cost functions in the control. The typically significant increase in computational demand arising from the nonlinearities has been mitigated in the last years by the steadily increasing available computation power for control processes (Findeisen and Allgöwer, 2002) and advances in the employed algorithms to solve the connected open-loop optimization (Ohtsuka, 2003).

Nevertheless, in most approaches, especially for the important case of continuous state spaces, the influence of noise on the system is not considered (Camacho and Bordons, 2004), which obviously leads to unsatisfactory solutions especially for highly nonlinear systems and cost functions. In (Deisenroth et al., 2006) an extension of the deterministic cost function by a term considering the noise is presented. In (Nikovski and Brand, 2003) an approach for infinite horizon optimal control is presented, where a continuous state space is discretized by means of a radial-basis-function network. This approach leads to a consideration of the noise influence, but as any discretization suffers from the curse of dimensionality.

In technical applications, like robotics or sensor-actuator-networks, discrete-time controllers for systems with continuous-valued state spaces, e.g. the posture of a robot, but a finite set of control inputs, e.g. turn left / right or move straight, are of special importance. Therefore, in this paper a framework for discrete-time NMPC for continuous state spaces and a finite set of control inputs is presented that is based on the efficient state prediction of nonlinear stochastic models. Since an exact density representation in closed form and with constant complexity is preferable, a prediction method is applied that is founded on the approximation of the involved system transition densities by axis-aligned Gaussian mixture densities (Huber et al., 2006). To lower the computational demands for approximating multi-dimensional transition densities, the so-called modularization for complexity reduction purposes is proposed. Thus, the Gaussian mixture representation of the predicted state can be evaluated efficiently with high approximation accuracy. As an additional part of this framework, an extremely flexible representation of the cost function, on which the optimization is based, is presented. Besides the commonly used quadratic deviation, a versatile Gaussian mixture representation of the cost function is introduced. This representation is very expressive due to the universal approximation property of Gaussian mixtures. Combining the efficient state prediction and the different cost function representations, an efficient integrated closed-form approach to NMPC for nonlinear noise affected systems with novel abilities is obtained.

The remainder of this paper is structured as follows: In the next section, the considered NMPC problem is described together with an example from the field of mobile robot control. In Section 3, the efficient closed-form prediction approach for nonlinear systems based on transition density approximation and complexity reduction is derived. Different techniques for modeling the cost function are introduced in Section 4. In Section 5, three different kinds of NMPC controllers are compared based on simulations employing the example system, which has been introduced in previous sections. Concluding remarks and perspectives on future work are given in Section 6.

2 PROBLEM FORMULATION

The considered discrete-time system is given by

$$\underline{\boldsymbol{x}}_{k+1} = \underline{a}(\underline{\boldsymbol{x}}_k, \underline{u}_k, \underline{\boldsymbol{w}}_k) \,, \tag{1}$$

where $\underline{\boldsymbol{x}}_k$ denotes the vector-valued random variable of the system state, $\underline{u}_k$ the applied control input, and $\underline{a}(\cdot)$ a nonlinear, time-invariant function. $\underline{\boldsymbol{w}}_k$ denotes the white stationary noise affecting the system additively *element-wise*, i.e., the elements of $\underline{\boldsymbol{w}}_k$ are processed in $\underline{a}(\cdot)$ just additively. For details see Section 3.3.

Example System

A mobile two-wheeled differential-drive robot is supposed to drive along a given trajectory, e.g. along a wall, with constant velocity. This robot can be modeled by the distance to the wall $\boldsymbol{x}_k$ and its orientation relative to the wall $\boldsymbol{\alpha}_k$, which leads to the discrete-time nonlinear system description

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + v \cdot T \cdot \sin(\boldsymbol{\alpha}_k) + \boldsymbol{w}_k^x \,, \\ \boldsymbol{\alpha}_{k+1} &= \boldsymbol{\alpha}_k + u_k + \boldsymbol{w}_k^\alpha \,, \end{aligned} \tag{2}$$

where $\underline{\boldsymbol{x}}_k = [\boldsymbol{x}_k, \boldsymbol{\alpha}_k]^\mathsf{T}$, v is a constant velocity, T the sampling interval and $\boldsymbol{w}_k^x$ and $\boldsymbol{w}_k^\alpha$ denote the noise influence on the system. The input u_k is a steering action, i.e., a change of direction of the robot. Furthermore, the robot is equipped with sensors to measure distance $\boldsymbol{y}_k^x$ and orientation $\boldsymbol{y}_k^\alpha$ with respect to the wall according to

$$\begin{aligned} \boldsymbol{y}_k^x &= \boldsymbol{x}_k + \boldsymbol{v}_k^x \,, \\ \boldsymbol{y}_k^\alpha &= \boldsymbol{\alpha}_k + \boldsymbol{v}_k^\alpha \,, \end{aligned} \tag{3}$$

where $\boldsymbol{v}_k^x$ and $\boldsymbol{v}_k^\alpha$ describe the measurement noise.

At any time step k, the system state is predicted over a finite N step prediction horizon. Then an open-loop optimal control problem is solved, i.e., the optimal input $\underline{u}_k^*$ is determined according to

$$\underline{u}_k^*(\underline{\boldsymbol{x}}_k) = \arg\min_{\underline{u}_k} V_k(\underline{\boldsymbol{x}}_k, \underline{u}_k)$$

with

$$V_k(\underline{\boldsymbol{x}}_k, \underline{u}_k) = \min_{\substack{\underline{u}_{k+1}, \dots, \\ \underline{u}_{k+N-1}}} \mathrm{E}_{\substack{\underline{x}_{k+1}, \dots, \\ \underline{x}_{k+N}}} \left\{ g_N(\underline{\boldsymbol{x}}_{k+N}) + \sum_{n=k}^{k+N-1} g_n(\underline{\boldsymbol{x}}_n, \underline{u}_n) \right\} \,, \tag{4}$$

where the optimality is defined by a cumulative value function $J_k(\underline{\boldsymbol{x}}_k)$

$$J_k(\underline{\boldsymbol{x}}_k) = \min_{\underline{u}_k} V_k(\underline{\boldsymbol{x}}_k, \underline{u}_k) \tag{5}$$

comprising the step costs $g_n(\underline{\boldsymbol{x}}_n, \underline{u}_n)$ depending on the predicted system states $\underline{\boldsymbol{x}}_n$ and the corresponding control inputs $\underline{u}_n$, as well as a terminal cost $g_N(\underline{\boldsymbol{x}}_{k+N})$. This optimal control input $\underline{u}_k^*$ is then applied to the system at time step k. In the next time step $k+1$ the whole procedure is repeated.

For most nonlinear systems, the analytical evaluation of (4) is not possible. One reason for this is the required prediction of system states for a noise-affected nonlinear system. The other one is the necessity to calculate expected values, which also cannot be performed in closed form. Therefore, in the next sections an integrated approach to overcome these two problems is presented.

3 STATE PREDICTION

Predicting the system state is an important part in NMPC for noise-affected systems. The probability density $\tilde{f}_{k+1}^{x}(\underline{x}_{k+1})$ of the system state $\underline{\boldsymbol{x}}_{k+1}$ for the next time step $k+1$ has to be computed utilizing the so-called Chapman-Kolmogorov equation (Schweppe, 1973)

$$\tilde{f}_{k+1}^{x}(\underline{x}_{k+1}) = \int_{\mathbb{R}^d} \tilde{f}_{\underline{u}_k}^{T}(\underline{x}_{k+1}|\underline{x}_k)\, \tilde{f}_k^{x}(\underline{x}_k)\mathrm{d}\underline{x}_k \,. \qquad (6)$$

The transition density $\tilde{f}_{\underline{u}_k}^{T}(\underline{x}_{k+1}|\underline{x}_k)$ depends on the system described by (1). For linear systems with Gaussian noise the Kalman filter (Kalman, 1960) provides an exact solution to (6), since this equation is reduced to the evaluation of an integral over a multiplication of two Gaussian densities, which is analytically solvable.

For nonlinear systems, an approximate description of the predicted density $\tilde{f}_{k+1}^{x}(\underline{x}_{k+1})$ is inevitable, since an exact closed-form representation is generally impossible to obtain. One very common approach in context of NMPC is linearizing the system and then applying the Kalman filter (Lee and Ricker, 1994). The resulting single Gaussian density is typically not sufficient for approximating $\tilde{f}_{k+1}^{x}(\underline{x}_{k+1})$. Hence, we propose representing all densities involved in (6) by means of Gaussian mixtures, which can be done efficiently due to their universal approximation property (Maz'ya and Schmidt, 1996).

To reduce the complexity of approximating all density functions corresponding to system (1) and to allow for an efficient state prediction, the *concept of modularization* is proposed, see Section 3.3. Here, (1) is decomposed into vector-valued subsystems. Approximations for these subsystems in turn can be reduced to the scalar case, as stated in Section 3.2. For that purpose, in the following section a short review on the closed-form prediction approach for scalar systems with additive noise is given. Employing this, modularization enables state prediction for system (1) based on Gaussian mixture approximations of the transition density functions corresponding to scalar systems.

3.1 Scalar Systems

For the scalar system equation

$$\boldsymbol{x}_{k+1} = a(\boldsymbol{x}_k, \underline{u}_k) + \boldsymbol{w}_k \;\;,$$

the approach proposed by (Huber et al., 2006) allows to perform a closed-form prediction resulting in an approximate Gaussian mixture representation $f_{k+1}^{x}(x_{k+1})$ of $\tilde{f}_{k+1}^{x}(x_{k+1})$,

$$f_{k+1}^{x}(x_{k+1}) = \sum_{i=1}^{L} \omega_i \cdot \mathcal{N}\left(x_{k+1} - \mu_i; \sigma_i^2\right) , \qquad (7)$$

where L is the number of Gaussian components, $\mathcal{N}\left(x_{k+1} - \mu_i; \sigma_i^2\right)$ is a Gaussian density with mean μ_i, standard deviation σ_i, and weighting coefficients ω_i with $\omega_i > 0$ as well as $\sum_{i=1}^{L} \omega_i = 1$. For obtaining this approximate representation of the true predicted density that provides high accuracy especially with respect to higher-order moments and a multimodal shape, the corresponding transition density $\tilde{f}_{\underline{u}_k}^{T}(x_{k+1}|x_k)$ from (6) is approximated off-line by the Gaussian mixture

$$\begin{aligned} &f_{\underline{u}_k}^{T}(x_{k+1}, x_k, \underline{\eta}) \\ &= \sum_{i=1}^{L} \omega_i \cdot \mathcal{N}\left(x_{k+1} - \mu_{i,1}; \sigma_{i,1}^2\right) \cdot \mathcal{N}\left(x_k - \mu_{i,2}; \sigma_{i,2}^2\right) \end{aligned}$$

with parameter vector

$$\underline{\eta} = \left[\underline{\eta}_1^{\mathrm{T}}, \ldots, \underline{\eta}_L^{\mathrm{T}}\right]^{\mathrm{T}} \;\;,$$

comprising L axis-aligned Gaussian components (short: axis-aligned Gaussian mixture), i.e., the covariance matrices of the Gaussian components are diagonal, with parameters

$$\underline{\eta}_i^{\mathrm{T}} = \left[\omega_i, \mu_{i,1}, \sigma_{i,1}, \mu_{i,2}, \sigma_{i,2}\right] \;\;.$$

The axis-aligned structure of the approximate transition density allows performing repeated prediction steps with constant complexity, i.e., a constant number L of mixture components for $f_{k+1}^{x}(x_{k+1})$.

This efficient prediction approach can be directly applied to vector-valued systems, like (1). However, off-line approximation of the multi-dimensional transition density corresponding to such a system is computationally demanding. Therefore, in the next two sections techniques to lower the computational burden are introduced.

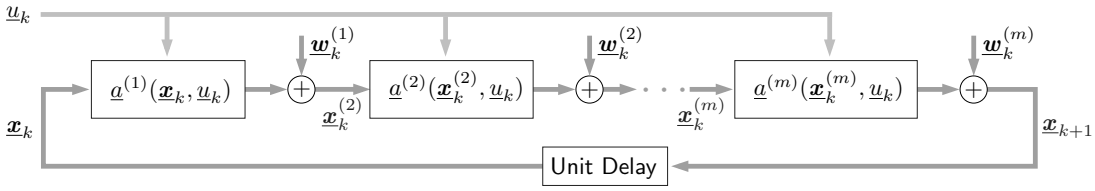


Figure 1: Modularization of the vector valued system $\underline{x}_{k+1.} = \underline{a}(\underline{x}_k, \underline{u}_k, \underline{w}_k)$.

3.2 Vector-Valued Systems

Now we consider the vector-valued system

$$\underline{x}_{k+1} = \underline{a}(\underline{x}_k, \underline{u}_k) + \underline{w}_k \ , \tag{8}$$

with $\underline{x}_{k+1} = [x_{k+1,1}, x_{k+1,2}, \ldots, x_{k+1,d}]^{\mathrm{T}} \in \mathbb{R}^d$ and noise $\underline{w}_k = [w_{k,1}, w_{k,2}, \ldots, w_{k,d}]^{\mathrm{T}} \in \mathbb{R}^d$. Assuming $\underline{w}_k$ to be white and stationary (but not necessarily Gaussian or zero-mean), with *mutually stochastically independent* elements $w_{k,j}$, approximating the corresponding transition density $\tilde{f}^T_{\underline{u}_k}(\underline{x}_{k+1}|\underline{x}_k) = \tilde{f}^w(\underline{x}_{k+1} - \underline{a}(\underline{x}_k, \underline{u}_k))$ can be reduced to the scalar system case.

Theorem 1 (Composed Transition Density)
The transition density $\tilde{f}^T_{\underline{u}_k}(\underline{x}_{k+1}|\underline{x}_k)$ of system (8) is composed of separate transition densities of the scalar systems $a_j(\cdot)$, $j = 1, 2, \ldots, d$, where $\underline{a}(\cdot) = [a_1(\cdot), a_2(\cdot), \ldots, a_d(\cdot)]^T$.

PROOF. Marginalizing $\tilde{f}^x_{k+1}(\underline{x}_{k+1})$ from the joint density function $\tilde{f}_k(\underline{x}_{k+1}, \underline{x}_k, \underline{w}_k)$ and separating the elements of $\underline{w}_k$ leads to

$$\begin{aligned}
&\tilde{f}^x_{k+1}(\underline{x}_{k+1}) \\
&= \int_{\mathbb{R}^{2d}} \delta(\underline{x}_{k+1} - \underline{a}(\underline{x}_k, \underline{u}_k) - \underline{w}_k)\, \tilde{f}^x_k(\underline{x}_k)\, \tilde{f}^w(\underline{w}_k) \mathrm{d}\underline{x}_k \mathrm{d}\underline{w}_k \\
&= \int_{\mathbb{R}^{2d}} \prod_{j=1}^{d} \delta(x_{k+1,j} - a_j(\underline{x}_k, \underline{u}_k) - w_{k,j}) \\
&\qquad \cdot \tilde{f}^x_k(\underline{x}_k) \prod_{j=1}^{d} \tilde{f}^{w_j}(w_{k,j})\, \mathrm{d}\underline{x}_k \mathrm{d}\underline{w}_k \\
&= \int_{\mathbb{R}^d} \Bigg(\prod_{j=1}^{d} \underbrace{\tilde{f}^{w_j}\left(x_{k+1,j} - a_j(\underline{x}_k, \underline{u}_k)\right)}_{\text{separate transition densities}} \Bigg) \tilde{f}^x_k(\underline{x}_k)\, \mathrm{d}\underline{x}_k \ .
\end{aligned}$$

As a result of the mutually stochastically independence of the elements in $\underline{w}_k$, the transition density of the vector-valued system (8) is separated into d transition densities of d scalar systems. Approximating these lower-dimensional transition densities is possible with decreased computational demand (Huber et al., 2006). □

The concept of modularization, introduced in the following section, benefits strongly from the result obtained in Theorem 1.

3.3 Concept of Modularization

For our proposed NMPC framework, we assume that the nonlinear system is corrupted by element-wise additive noise. Incorporating this specific noise structure, the previously stated closed-form prediction step can indirectly be utilized for system (1). Similar to Rao-Blackwellised particle filters (de Freitas, 2002), we can reduce the system in (1) to a set of less complex subsystems with a form according to (8),

$$\begin{aligned}
\underline{x}_{k+1} &= \underline{a}(\underline{x}_k, \underline{u}_k, \underline{w}_k) = \underline{a}^{(m)}(\underline{x}_k^{(m)}, \underline{u}_k) + \underline{w}_k^{(m)} \\
\underline{x}_k^{(m)} &= \underline{a}^{(m-1)}(\underline{x}_k^{(m-1)}, \underline{u}_k) + \underline{w}_k^{(m-1)} \\
&\ \ \vdots \\
\underline{x}_k^{(2)} &= \underline{a}^{(1)}(\underline{x}_k^{(1)}, \underline{u}_k) + \underline{w}_k^{(1)} \ .
\end{aligned}$$

We name this approach *modularization*, where the subsystems

$$\underline{x}_k^{(i+1)} = \underline{a}^{(i)}(\underline{x}_k^{(i)}, \underline{u}_k) + \underline{w}_k^{(i)}, \text{ for } i = 1, \ldots, m$$

correspond to transition densities, that can be approximated according to Section 3.1 and 3.2. Since these subsystems are less complex than the overall system (1), approximating transition densities is also less complex. Furthermore, a nested prediction can be performed to obtain the predicted density $f^x_{k+1}(\underline{x}_{k+1})$, see Fig. 1. Starting with $\underline{x}_k^{(1)} = \underline{x}_k$, each subsystem $\underline{a}^{(i)}(\cdot)$ receives an *auxiliary system state* $\underline{x}_k^{(i)}$ and generates an auxiliary predicted system state $\underline{x}_k^{(i+1)}$.

The noise $\underline{w}_k$ is separated into its subvectors $\underline{w}_k^{(i)}$ according to

$$\underline{w}_k = [\underline{w}_k^{(1)}, \underline{w}_k^{(2)}, \ldots, \underline{w}_k^{(m)}]^{\mathrm{T}} \ ,$$

in case that the single noise subvectors $\underline{w}_k^{(i)}$ are mutually independent.

Example System: Modularization
The system model (2) describing the mobile robot can be modularized into the subsystems

$$x_k^{(2)} = \sin(\alpha_k) + w_k^x$$

and

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + v \cdot T \cdot \boldsymbol{x}_k^{(2)} \ ,$$
$$\boldsymbol{\alpha}_{k+1} = \boldsymbol{\alpha}_k + u_k + \boldsymbol{w}_k^{\alpha} \ .$$

The auxiliary system state $\boldsymbol{x}_k^{(2)}$ is stochastically dependent on $\boldsymbol{\alpha}_k$. We omitted this dependence in further investigations of the example system for simplicity.

Please note that there are typically stochastic dependencies between several auxiliary system states. To consider this fact, the relevant auxiliary system states have to be augmented to conserve the dependencies. Thus, the dimensions of these auxiliary states need not all to be equal.

4 COST FUNCTIONS

In this section, two possibilities to model cost functions, the well known quadratic deviation and a novel approach employing Gaussian mixture cost functions, are presented. Exploiting the fact that the predicted state variables are, as explained in the previous section, described by Gaussian mixture densities, the necessary evaluation of the expected values in (4) can be calculated efficiently in closed-form for both options.

In the following, cumulative value functions according to (5) are considered, where $g_n(\underline{\boldsymbol{x}}_n, \underline{u}_n)$ denotes a step cost within the horizon and $g_N(\underline{\boldsymbol{x}}_n)$ a cost depending on the terminal state at the end of the horizon. The value function $J_k(\underline{\boldsymbol{x}}_k)$ is the minimal cost for the next N steps of the system, starting at state $\underline{\boldsymbol{x}}_k$ plus the terminal cost $g_N(\underline{\boldsymbol{x}}_n)$.

For simplicity, step costs that are additively decomposable according to

$$g_n(\underline{\boldsymbol{x}}_n, \underline{u}_n) = g_n^x(\underline{\boldsymbol{x}}_n) + g_n^u(\underline{u}_n)$$

are considered, although the proposed framework is not limited to this case.

4.1 Quadratic Cost

One of the most popular cost functions is the quadratic deviation from a target value $\check{\underline{x}}$ or $\check{\underline{u}}$ according to

$$g_n^x(\underline{\boldsymbol{x}}_n) = (\underline{\boldsymbol{x}}_n - \check{\underline{x}}_n)^{\mathrm{T}} (\underline{\boldsymbol{x}}_n - \check{\underline{x}}_n) \ .$$

As in our framework the probability density function of the state $\underline{\boldsymbol{x}}_n$ is given by an axis-aligned Gaussian mixture $f_n^x(\underline{x}_n)$ with L components, the calculation of $\mathrm{E}_{\underline{\boldsymbol{x}}_n}\{g_n^x(\underline{\boldsymbol{x}}_n)\}$, which is necessary to compute

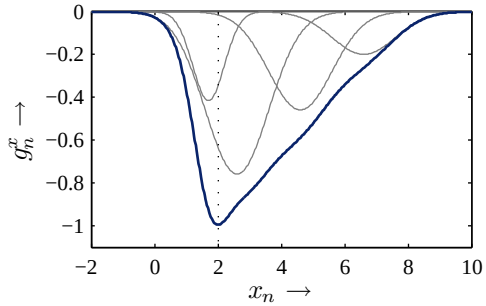


Figure 2: Asymmetric cost function consisting of four components (gray) with a minimum at $\check{x}_n = 2$.

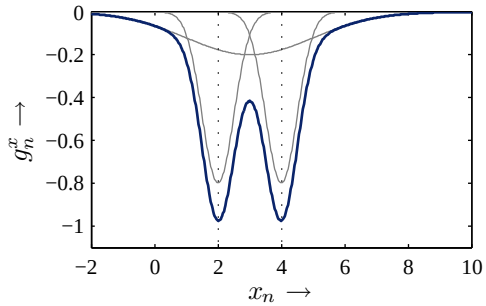


Figure 3: Multimodal cost function consisting of three components (gray) with minima at $\check{x}_n^a = 2$, $\check{x}_n^b = 4$.

(4), can be performed analytically as it can be interpreted as the sum over shifted and dimension-wise calculated second-order moments

$$\begin{aligned}
&\mathop{\mathrm{E}}_{\underline{\boldsymbol{x}}_n}\{g_n^x(\underline{\boldsymbol{x}}_n)\} \\
&= \mathop{\mathrm{E}}_{\underline{\boldsymbol{x}}_n}\{(\underline{\boldsymbol{x}}_n - \check{\underline{x}}_n)^{\mathrm{T}} (\underline{\boldsymbol{x}}_n - \check{\underline{x}}_n)\} \\
&= \mathrm{trace} \mathop{\mathrm{E}^2}_{\underline{\boldsymbol{x}}_n}\{(\underline{\boldsymbol{x}}_n - \check{\underline{x}}_n)\} \\
&= \mathrm{trace} \sum_{i=1}^{L} \omega_i \left((\underline{\mu}_i - \check{\underline{x}}_n)(\underline{\mu}_i - \check{\underline{x}}_n)^{\mathrm{T}} + \mathrm{diag}(\underline{\sigma}_i)^2 \right),
\end{aligned}$$

employing $\mathrm{E}_{\boldsymbol{x}}^2\{\boldsymbol{x}\} = \sum_{i=1}^{L} \omega_i (\mu_i^2 + \sigma_i^2)$.

Example System: Quadratic Cost
If the robot is intended to move parallel along the wall, the negative quadratic deviation of the angle $\boldsymbol{\alpha}_k$ with respect to the wall, i.e., $g_n^{\alpha}(\boldsymbol{\alpha}_n) = (\boldsymbol{\alpha}_n - \alpha_n^{Wall})^2$ is a suitable cost function.

4.2 Gaussian Mixture Cost

A very versatile description of the cost function can be realized if Gaussian mixtures are employed. In this case, arbitrary cost functions can be realized due to the Gaussian mixtures' universal approximation property (Maz'ya and Schmidt, 1996). Obviously, in this

case the Gaussian mixtures may have arbitrary parameters, e.g. negative weights ω.

Example System: Gaussian Mixture Cost Function
In case the robot is intended to move at a certain optimal distance to the wall (e.g. $\check{x}_n = 2$, with $x_n^{Wall} = 0$), where being closer to the wall is considered less desirable than being farther away, this can, e.g. be modeled with a cost function as depicted in Fig. 2. If two different distances are considered equally optimal, this can be modeled with a cost function as depicted in Fig. 3.

Please note that assigning rewards to a state is equivalent to assigning negative costs, which leads to the depicted negative cost functions in Fig. 2 and Fig. 3.

Here, the calculation of the expected value $\mathrm{E}_{\underline{\boldsymbol{x}}_n}\{g_n^x(\underline{\boldsymbol{x}}_n)\}$, which is necessary for the calculation of (4), can also be performed analytically

$$\begin{aligned}
&\mathop{\mathrm{E}}_{\underline{\boldsymbol{x}}_n}\{g_n^x(\underline{\boldsymbol{x}}_n)\} \\
&= \int_{\mathbb{R}^d} f_n^x(\underline{x}_n)\cdot g_n^x(\underline{x}_n)\,\mathrm{d}\underline{x}_n \\
&= \int_{\mathbb{R}^d} \sum_{i=1}^{L} \omega_i\, \mathcal{N}\left(\underline{x}_n - \underline{\mu}_i; \mathrm{diag}(\underline{\sigma}_i)^2\right) \\
&\qquad \cdot \sum_{j=1}^{M} \omega_j\, \mathcal{N}\left(\underline{x}_n - \underline{\mu}_j; \mathrm{diag}(\underline{\sigma}_j)^2\right) \mathrm{d}\underline{x}_n \\
&= \sum_{i=1}^{L}\sum_{j=1}^{M} \omega_{ij} \underbrace{\int_{\mathbb{R}^d} \mathcal{N}\left(\underline{x}_n - \underline{\mu}_{ij}; \mathrm{diag}(\underline{\sigma}_{ij})^2\right) \mathrm{d}\underline{x}_n}_{=1} ,
\end{aligned}$$

with

$$\omega_{ij} = \omega_i\omega_j \cdot \mathcal{N}\left(\underline{\mu}_i - \underline{\mu}_j; \mathrm{diag}(\underline{\sigma}_i)^2 + \mathrm{diag}(\underline{\sigma}_j)^2\right) ,$$

where $f_n^x(\underline{x}_n)$ denotes the L-component Gaussian mixture probability density function of the system state (7) and $g_n^x(\underline{\boldsymbol{x}}_n)$ the cost function, which is a Gaussian mixture with M components.

4.3 Input Dependent Part

The input dependent part of the cost function $g_n^u(\underline{u}_n)$ can either be modeled similar to the procedures described above or with a lookup-table since there is just a finite number of discrete $\underline{u}_n$.

Using the efficient state prediction presented in Section 3 together with the value function representations presented above, (4) can be solved analytically for a finite set of control inputs. Thus, an efficient closed-form solution for the optimal control problem within NMPC is available. Its capabilities will be illustrated by simulations in the next section.

5 SIMULATIONS

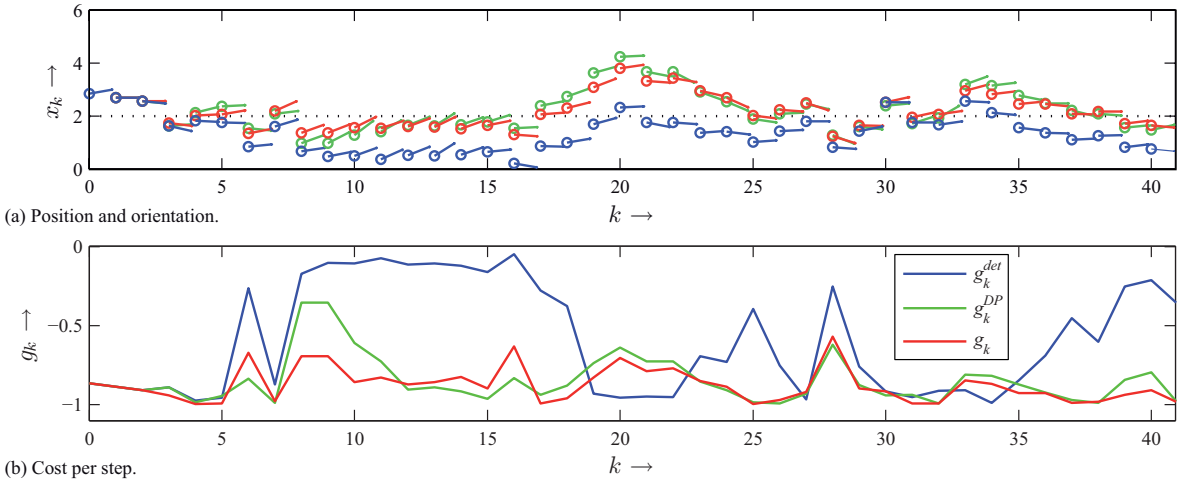
Based on the above example scenario, several simulations are conducted to illustrate the modeling capabilities of the proposed framework as well as to illustrate the benefits that can be gained by the direct consideration of noise in the optimal control optimization. The considered system is given by (2) and (3), with $v \cdot T = 1$ and $u_k \in \{-0.2, -0.1,\ 0,\ 0.1,\ 0.2\}$. The considered noise influences on the system $\boldsymbol{w}_k^x$ and $\boldsymbol{w}_k^\alpha$ are zero-mean white Gaussian noise with standard deviation $\sigma_w^x = 0.5$ and $\sigma_w^\alpha = 0.05 \approx 2.9^\circ$ respectively. The measurement noise is also zero-mean white Gaussian noise with standard deviation $\sigma_v^x = 0.5$ and $\sigma_v^\alpha = 0.1 \approx 5.7^\circ$. All simulations are performed for a $N = 4$ step prediction horizon, with a value function according to (5), where $g_N(\underline{\boldsymbol{x}}_{k+N})$ is the function depicted in Fig. 2 and $g_n(\underline{\boldsymbol{x}}_n, \underline{u}_n) = g_N(\underline{\boldsymbol{x}}_{k+N})\ \forall n$. In addition, the modularization is employed as described above.

To evaluate the benefits of the proposed NMPC framework, three different kind of simulations are performed:

Calculation of the input without noise consideration (deterministic NMPC):
The deterministic control is calculated as a benchmark neglecting the noise influence.

Direct calculation of the optimal input considering all noise influences (stochastic NMPC):
The direct calculation of the optimal input with consideration of the noise is performed using the techniques presented in the previous sections. Thus, it is possible to execute all calculations analytically without the need for any numerical methods. Still, this approach has the drawback that the computational demand for the optimal control problem increases exponentially with the length of the horizon N. Thus, this approach is suitable for short horizons only.

Calculation of the optimal input with a value function approximation scheme and Dynamic Programming (stochastic NMPC with Dynamic Programming):
In order to be able to use the framework efficiently also for long prediction horizons, it is necessary to employ Dynamic Programming (DP). Unfortunately, this is not directly possible, as no closed-form solution for the value function J_n is available. One easy but either not very accurate or computationally demanding solution would be to discretize the state space. More advanced solutions can be found by value function approximation (Bertsekas, 2000). For the simulations, an especially well-suited case of



(a) Position and orientation.

(b) Cost per step.

Figure 4: First 40 steps of a simulation (red solid line: stochastic NMPC, green dotted line: stochastic NMPC with DP, blue dashed line: deterministic NMPC).

value function approximation is employed that has been described by (Nikovski and Brand, 2003). Here, the state space is discretized by covering it with a finite set of Gaussians with fixed means and covariances. Then weights, i.e., scaling factors, are selected in such a way that the approximate and the true value function coincide at the means of every Gaussian. Using these approximate value functions together with the techniques described above, again all calculations can be executed analytically. In contrast to the direct calculation, now the computational demand increases only linearly with the length of the prediction horizon but quadratically in the number of Gaussians used to approximate the value function. Here, the value functions are approximated by a total of 833 Gaussians equally spaced over the state space within $(\hat{x}_n, \hat{\alpha}_n) \in \Omega := [-2, 10] \times [-2, 2]$.

For each simulation run, a particular noise realization is used that is applied to the different controllers. In Fig. 4(a), the first 40 steps of a simulation run are shown. The distance to the wall x_k is depicted by the position of the circles, the orientation α_k by the orientation of the arrows. Besides that the system is heavily influenced by noise, it can be clearly seen that the robot under deterministic control behaves very differently from the other two. The deterministic controller just tries to move the robot to the minimum of the cost function at $\check{x}_k = 2$ and totally neglects the asymmetry of the cost function. The stochastic controllers lead to a larger distance to the wall, as they consider the noise affecting the system in conjunction with the non-symmetric cost function.

In Fig. 4(b), the evaluation of the cost function for each step is shown. As expected, both stochastic controllers perform much better, i.e., they generate less cost, than the deterministic one. This finding has been validated by a series of 100 Monte Carlo simulations with different noise realizations and initial values. The uniformly distributed initial values are sampled from the interval $x_0 \in [0, 8]$ and $\alpha_0 \in [-\pi/4, \pi/4]$. In Table 1, the average step costs of the 100 simulations with 40 steps each are shown. To facilitate the comparison, also normalized average step costs are given. Here, it can be seen that the stochastic controller outperforms the deterministic one by over 10% in terms of cost. In 82% of the runs, the stochastic controller gives overall better results than the deterministic one. By employing dynamic programming together with value function approximation the benefits are reduced. Here, the deterministic controller is only outperformed by approximately 3.5%. The analysis of the individual simulations leads to the conclusion that the control quality significantly degrades in case the robot attains a state which is less well approximated by the value function approximation as it lies outside Ω. Still, the dynamic programming approach produced better results than the deterministic approach in 69% of the runs. These findings illustrate the need for advanced value function approximation techniques in order to gain the very good control performance of the direct stochastic controller together with the efficient calculation of the DP approach.

Table 1: Simulation Results.

controller	average cost	
deterministic	-0.6595	(100.00%)
stochastic	-0.7299	(110.66%)
stochastic DP	-0.6824	(103.48%)

6 CONCLUSIONS

A novel framework for closed-form Nonlinear Model Predictive Control (NMPC) for continuous state space and a finite set of control inputs has been presented that directly incorporates the noise influence in the corresponding optimal control problem. By using the proposed state prediction methods, which are based on transition density approximation by Gaussian mixture densities and complexity reduction techniques, the otherwise not analytically solvable state prediction of nonlinear noise affected systems can be performed in an efficient closed-form manner. Another very important aspect of NMPC is the modeling of the cost function. The proposed methods also use Gaussian mixtures, which leads to a level of flexibility far beyond the traditional representations. By employing the same representation for both the predicted probability density functions and the cost functions, NMPC is solvable in closed-form for nonlinear systems with consideration of noise influences. The effectiveness of the presented framework and the importance of the consideration of noise in the controller have been shown in simulations of a two-wheeled differential-drive robot following a specified trajectory.

Future research is intended to address various topics. One is the optimization of the value function approximation by abandoning a fixed grid in order to increase performance and accuracy. An additional important task will be the consideration of stability aspects, especially in cases of approximated value functions. This can, e.g. be tackled by the use of bounding techniques for the approximation error (Lincoln and Rantzer, 2006). Another interesting extension will be the incorporation of effects of inhomogeneous noise, i.e., noise with state and/or input dependent noise levels. Together with the incorporation of nonlinear filtering techniques this is expected to increase the control quality even more.

Besides the addition of new features to the framework, also the extension to new application fields is intended. Of special interest is the extension of Model Predicted Control to the related emerging field of Model Predictive Sensor Scheduling (He and Chong, 2004), which is of special importance, e.g. in sensor-actuator-networks.

REFERENCES

Bertsekas, D. P. (2000). *Dynamic Programming and Optimal Control*. Athena Scientific, Belmont, Massachusetts, U.S.A., 2nd edition.

Camacho, E. F. and Bordons, C. (2004). *Model Predictive Control*. Springer-Verlag London Ltd., 2 edition.

Deisenroth, M. P., Ohtsuka, T., Weissel, F., Brunn, D., and Hanebeck., U. D. (2006). Finite-Horizon Optimal State Feedback Control of Nonlinear Stochastic Systems Based on a Minimum Principle. In *Proc. of the IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, pages 371–376.

Findeisen, R. and Allgöwer, F. (2002). An Introduction to Nonlinear Model Predictive Control. In Scherer, C. and Schumacher, J., editors, *Summerschool on "The Impact of Optimization in Control", Dutch Institute of Systems and Control (DISC)*, pages 3.1–3.45.

de Freitas, N. (2002). Rao-Blackwellised Particle Filtering for Fault Diagnosis. In *IEEE Aerospace Conference Proceedings*, volume 4, pages 1767–1772.

He, Y. and Chong, E. K. P. (2004). Sensor Scheduling for Target Tracking in Sensor Networks. In *Proceedings of the 43rd IEEE Conference on Decision and Control*, volume 1, pages 743–748.

Huber, M., Brunn, D., and Hanebeck, U. D. (2006). Closed-Form Prediction of Nonlinear Dynamic Systems by Means of Gaussian Mixture Approximation of the Transition Density. In *Proc. of the IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, pages 98–103.

Kalman, R. E. (1960). A new Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME, Journal of Basic Engineering*, (82):35–45.

Lee, J. H. and Ricker, N. L. (1994). Extended Kalman Filter Based Nonlinear Model Predictive Control. In *Industrial & Engineering Chemistry Research*, pages 1530–1541. ACS.

Lincoln, B. and Rantzer, A. (2006). Relaxing Dynamic Programming. *IEEE Transactions on Automatic Control*, 51(8):1249–1260.

Maz'ya, V. and Schmidt, G. (1996). On Approximate Approximations using Gaussian Kernels. *IMA Journal of Numerical Analysis*, 16(1):13–29.

Nikovski, D. and Brand, M. (2003). Non-Linear Stochastic Control in Continuous State Spaces by Exact Integration in Bellman's Equations. In *Proc. of the 2003 International Conf. on Automated Planning and Scheduling*, pages 91–95.

Ohtsuka, T. (2003). A Continuation/GMRES Method for Fast Computation of Nonlinear Receding Horizon Control. *Automatica*, 40(4):563–574.

Qin, S. J. and Badgewell, T. A. (1997). An Overview of Industrial Model Predictive Control Technology. *Chemical Process Control*, 93:232–256.

Schweppe, F. C. (1973). *Uncertain Dynamic Systems*. Prentice-Hall.

EXPLICIT PREDICTIVE CONTROL LAWS
On the Geometry of Feasible Domains and the Presence of Nonlinearities

Sorin Olaru, Didier Dumur
Automatic Control Department, Supélec, 3 rue Joliot Curie, Gif-sur-Yvette, France
{sorin.olaru;didier.dumur}@supelec.fr

Simona Dobre
CRAN, Nancy - Université, CNRS UMR 7039, BP 239, F-54506 Vandœuvre-lès-Nancy Cedex, France
simona.dobre@cran.uhp-nancy.fr

Keywords: Predictive control, parameterized polyhedra, explicit control laws.

Abstract: This paper proposes a geometrical analysis of the polyhedral feasible domains for the predictive control laws under constraints. The fact that the system dynamics influence the topology of such polyhedral domains is well known from the studies dedicated to the feasibility of the control laws. Formally the system state acts as a vector of parameters for the optimization problem to be solved on-line and its influence can be fully described by the use of parameterized polyhedra and their dual constraints/generators representation.
Problems like the constraints redundancy or the construction of the associated explicit control laws at least for linear or quadratic cost functions can thus receive fully geometrical solutions. Convex nonlinear constraints can be approximated using a description based on the parameterized vertices. In the case of nonconvex regions the explicit solutions can be obtain by constructing Voronoi partitions based on a collection of points distributed over the borders of the feasible domain.

1 INTRODUCTION

The philosophy behind Model-based Predictive Control (MPC) is to exploit in a "receding horizon" manner the simplicity of the Euler-Lagrange approach for the optimal control. The control action u_t for a given state x_t is obtained from the control sequence $\mathbf{k}_u^* = [u_t^T, \ldots, u_{t+N-1}^T]^T$ as a result of the optimization problem:

$$\begin{array}{ll} \min\limits_{\mathbf{k}_u} & \varphi(x_{t+N}) + \sum\limits_{k=0}^{N-1} l(x_{t+k}, u_{t+k}) \\ \text{subj. to :} & x_{t+1} = f(x_t) + g(x_t)u_t; \\ & h(x_t, \mathbf{k}_u) \leq 0 \end{array} \quad (1)$$

constructed for a finite prediction horizon N, cost per stage $l(.)$, terminal weight $\varphi(.)$, the system dynamics described by $f(.), g(.)$ and the constraints written in a compact form using elementwise inequalities on functions linking the states and the control actions, $h(.)$.

Unfortunately, the control sequence $\mathbf{k}_u^*$ is optimal only for a single initial condition - x_t and produces an open-loop trajectory which contrasts with the need for a feedback control law. This drawback is overcome by solving the local optimization (1) for every encountered (measured) state, thus indirectly producing a state feedback law. The overall methodology is based on computationally tractable optimal control problems for the states found along the current trajectory. However, two important directions are to be studied in order to enlarge the class of systems which can take advantage of the MPC methodology. One is related to the fact that the measurements can be available faster than the optimal control sequence becomes available (as output of the optimization solver) and thus important information can be lost with irreversible consequences on the close-loop performances. Secondly, the lack of a closed form expression for the feedback law notifies about the difficulties that can be encountered when considering properties such as stability, typically established for regions in the state space.

For the optimization problem (1) within MPC, the current state serves as an initial condition and influences both the objective function and the feasible domain. Globally, from the optimization point of view, the system state can be interpreted as a vector of parameters, and the problems to be solved

are part of the multiparametric optimization programming family. From the cost function point of view, the parametrization is somehow easier to deal with and eventually can be entirely translated towards the set of constraints to be satisfied (the MPC literature contains references to schemes based on suboptimality or even to algorithms restraining the demands to feasible solution of the receding horizon optimization (Scokaert et al., 1999)). Unfortunately, similar observation cannot be made about the feasible domain and its adjustment with respect to the parameters evolution.

The optimal solution is often influenced by the limitations, the process being forced to operate at the designed constraints for best performance. The distortion of the feasible domain during the parameters evolution will consequently affect the structure of the optimal solution. Starting from this observation the present paper focuses on the topological analysis of the domains described by the MPC constraints.

The structure of the feasible domain is depending on the model and the set of constraints taken into consideration in (1). If the model is a linear system, the presence of linear constraints on inputs and states can be easily expressed by a system of linear inequalities. In the case of nonlinear systems, these properties are lost and the domains are in general difficult to handle. However, there are several approaches to transform the dynamics to those of a linear system over the operating range as for example by piecewise linear approximation, feedback linearisation or the use of time-varying linear models.

As a consequence, specific attention for the linear constraints and the associated polyhedral feasible domains may be prolific. More than that, the use of polyhedral domains between the convex sets is not hazardous since they offer important advantages, like the closeness over the intersection or the fact that the polyhedral invariant sets (largely used for enforcing stability) are less conservative than the ellipsoidal ones for example. In the current paper, these polyhedral feasible domains will be analyzed with a focus on the parametrization leading to the concept of parameterized polyhedra (Olaru and Dumur,):

$$\begin{aligned} &\min_{\mathbf{k}_u} && F(x_t,\mathbf{k}_u) \\ &\text{subj. to}: && \begin{cases} A_{in}\mathbf{k}_u \leq b_{in}+B_{in}x_t \\ A_{eq}\mathbf{k}_u = b_{eq}+B_{eq}x_t \\ h(x_t,\mathbf{k}_u)\leq 0 \end{cases} \end{aligned} \tag{2}$$

where the objective function $F(x_t,\mathbf{k}_u)$ is usually linear or quadratic.

Secondly it will be shown that the optimization problem may take advantage during the real-time implementation either from the possible alleviation of the set of constraints for the on-line optimization routines either from the construction of the explicit solution on geometrical basis when possible. With these two aspects, one can consider that MPC awareness is improved both from the theoretical (insight on the global control law) and practical (computational aspects) point of view.

An important remark concerning the presence of nonlinearities in the constraints is that if the feasible domain remains convex, then an approximation in terms of parameterized polyhedra can lead to an approximate explicit solution in terms of piecewise linear control laws. But, if the feasible domain becomes non-convex due to the presence of nonlinearities, then in order to obtain the explicit solution some assumption have to be relaxed. A special role, in finding the explicit control laws in the nonlinear case, is played by the Voronoi partition.

In the following, Section 2 introduces the basic concepts related to the parameterized polyhedra and interprets the feasible domains of the receding horizon optimization problems (2) in this context. Section 3 presents the use of the feasible domain analysis for the construction of the explicit solution for linear and quadratic objective functions. In Section 4 an extension to nonlinear type of constraints is addressed, simple examples illustrating the construction of explicit solutions.

2 PARAMETRIZATION OF POLYHEDRAL DOMAINS

2.1 Double Representation

A mixed system of linear equalities and inequalities defines a polyhedron (Motzkin and R.M., 1953). In the parameter free case, it is represented by the equivalent dual (Minkowski) formulation:

$$\begin{aligned} &\mathcal{P} = \left\{ \mathbf{k}_u \in \mathbb{R}^p \,\middle|\, A_{eq}\,\mathbf{k}_u = b_{eq}; A_{in}\mathbf{k}_u \leq b_{in} \right\} \\ &\Longleftrightarrow \mathcal{P} = \underbrace{conv.hull\mathbf{V} + cone\mathbf{R} + lin.space\mathbf{L}}_{generators} \end{aligned} \tag{3}$$

where *conv.hull*$\mathbf{V}$ denotes the set of convex combinations of vertices $\mathbf{V}=\{\mathbf{v}_1,\ldots,\mathbf{v}_\vartheta\}$, *cone*$\mathbf{R}$ denotes nonnegative combinations of unidirectional rays in $\mathbf{R}=\{\mathbf{r}_1,\ldots,\mathbf{r}_\rho\}$ and *lin.space*$\mathbf{L}=\{\mathbf{l}_1,\ldots,\mathbf{l}_\lambda\}$ represents a linear combination of bidirectional rays (with ϑ, ρ and λ the cardinals of the related sets). This dual representation (Schrijver, 1986) in terms of generators can be rewritten as:

$$\mathcal{P} = \left\{ \mathbf{k}_u \in \mathbb{R}^p | \, \mathbf{k}_u = \sum_{i=1}^{\vartheta} \alpha_i \mathbf{v}_i + \sum_{i=1}^{\rho} \beta_i \mathbf{r}_i + \sum_{i=1}^{\lambda} \gamma_i \mathbf{l}_i; \right. \\ \left. 0 \le \alpha_i \le 1, \; \sum_{i=1}^{\vartheta} \alpha_i = 1 \, , \beta_i \ge 0 \, , \forall \gamma_i \right\} \tag{4}$$

with α_i, β_i, γ_i the coefficients describing the convex, non-negative and linear combinations in (3).

Numerical methods like the Chernikova algorithm (Leverge, 1994) are implemented for constructing the double description, either starting from constraints (3) either from the generators (4) representation.

2.2 The Parametrization

A *parameterized polyhedron* (Loechner and Wilde, 1997) is defined in the implicit form by a finite number of inequalities and equalities with the note that the affine part depends linearly on a vector of parameters $x \in \mathbb{R}^n$ for both equalities and inequalities:

$$\begin{aligned} \mathcal{P}(x) &= \left\{ \mathbf{k}_u(x) \in \mathbb{R}^p \, | A_{eq} \, \mathbf{k}_u = B_{eq} x + b_{eq}; \right. \\ &\qquad \left. A_{in} \mathbf{k}_u \le B_{in} x + b_{in} \right\} \\ &= \left\{ \mathbf{k}_u(x) | \, \mathbf{k}_u(x) = \sum_{i=1}^{\vartheta} \alpha_i(x) \mathbf{v}_i(x) \right. \\ &\qquad \left. + \sum_{i=1}^{\rho} \beta_i \mathbf{r}_i + \sum_{i=1}^{\lambda} \gamma_i \mathbf{l}_i \right\} \\ &0 \le \alpha_i(x) \le 1, \; \sum_{i=1}^{\vartheta} \alpha_i(x) = 1 \, , \beta_i \ge 0 \, , \forall \gamma_i. \end{aligned} \tag{5}$$

This dual representation of the parameterized polyhedral domain reveals the fact that only the vertices are concerned by the parametrization (resulting the so-called *parameterized vertices* - $\mathbf{v}_i(x)$), whereas the rays and the lines do not change with the parameters' variation. In order to effectively use the generators representation in (5), several aspects have to be clarified regarding the parametrization of the vertices (see for exemple (Loechner and Wilde, 1997) and the geometrical toolboxes like POLYLIB (Wilde, 1993)). The basic idea is to identify the parameterized polyhedron with a non-parameterized one in an augmented space:

$$\tilde{\mathcal{P}} = \left\{ \begin{bmatrix} \mathbf{k}_u \\ x \end{bmatrix} \in \mathbb{R}^{p+n} | \, \begin{bmatrix} A_{eq} | - B_{eq} \end{bmatrix} \begin{bmatrix} \mathbf{k}_u \\ x \end{bmatrix} = b_{eq}; \right. \\ \left. \begin{bmatrix} A_{in} | - B_{in} \end{bmatrix} \begin{bmatrix} \mathbf{k}_u \\ x \end{bmatrix} \le b_{in} \right\} \tag{6}$$

The original polyhedron in (5) can be found for any particular value of the parameters vector x through $P(x) = \text{Proj}_{\mathbf{k}_u}\left(\tilde{P} \cap H(x)\right)$, for any given hyperplane $H(x_0) = \left\{ \begin{pmatrix} \mathbf{k}_u \\ x \end{pmatrix} \in \mathbb{R}^{p+n} | x = x_0 \right\}$ and using $\text{Proj}_{\mathbf{k}_u}(.)$ as the projection from $\mathbb{R}^{p+n}$ to the first p coordinates $\mathbb{R}^p$.

Within the polyhedral domains $\tilde{\mathcal{P}}$, the correspondent of the parameterized vertices in (5) can be found among the faces of dimension n. After enumerating these n-faces: $\left\{ F_1^n(\tilde{\mathcal{P}}), \ldots F_j^n(\tilde{\mathcal{P}}), \ldots, F_\varsigma^n(\tilde{\mathcal{P}}) \right\}$, one can write: $\forall i, \exists j \in \{1, \ldots, \varsigma\} \, s.t. \begin{bmatrix} \mathbf{v}_i(x)^T & x^T \end{bmatrix}^T \in F_j^n(\tilde{\mathcal{P}})$ or equivalently:

$$\mathbf{v}_i(x) = \text{Proj}_{\mathbf{k}_u}\left(F_j^n(\tilde{\mathcal{P}}) \cap H(x) \right) \tag{7}$$

From this relation it can be seen that not all the n-faces correspond to parameterized vertices. However it is still easy to identify those which can be ignored in the process of construction of parameterized vertices based on the relation: $\text{Proj}_x\left(F_j^n(\tilde{P})\right) < n$ with $\text{Proj}_x(.)$ the projection from $\mathbb{R}^{p+n}$ to the last n coordinates $\mathbb{R}^n$ (corresponding to the parameters' space). Indeed the projections are to be computed for all the n-faces, those which are degenerated are to be discarded and all the others are stored as validity domains - $D_{\mathbf{v}_i} \in \mathbb{R}^n$, for the parameterized vertices that they are identifying:

$$D_{\mathbf{v}_i} = \text{Proj}_n\left(F_j^n(\tilde{P}) \right) \tag{8}$$

Once the parameterized vertices identified and their validity domain stored, the dependence on the parameters vector can be found using the supporting hyperplanes for each n-face:

$$\mathbf{v}_i(x) = \begin{bmatrix} \underline{A}_{eq} \\ A_{in_j} \end{bmatrix}^{-1} \begin{bmatrix} \underline{B}_{eq} \\ B_{in_j} \end{bmatrix} x + \begin{bmatrix} \underline{b}_{eq} \\ b_{in_j} \end{bmatrix} \tag{9}$$

where $\underline{A}_{in_j}$, $\underline{B}_{in_j}$, $\underline{b}_{in_j}$ represent the subset of the inequalities, satisfied by saturation for $F_j^n(\tilde{P})$. The inversion is well defined as long as the faces with degenerate projections are discarded.

2.3 The Interpretation from the Predictive Control Point of View

The double representation of the parameterized polyhedra offers a complete description of the feasible domain for the predictive control law as long as this is based on a multiparametric optimization with linear constraints.

Using the generators representation, with simple difference operations on convex sets one can compute the region of the parameters space where no parameterized vertex is defined:

$$\aleph = \mathbb{R}^n \setminus \{ \cup D_{\mathbf{v}_i}; \; i = 1 \ldots \vartheta \} \tag{10}$$

representing from the MPC point of view, the set of infeasible states for which no control sequence can be designed due to the fact that the limitations are overly

constraining. As a consequence the complete description of the infeasibility is obtained.

Remark: the presence of rays and lines in the set of generators doesn't imply that the infeasibility is avoided. The feasibility is strictly related with the existence of valid parameterized vertices for the given value of the parameter (state) vector.

The vertices of the feasible domain cannot be expressed as convex combinations of other distinct points and, due to the fact that from the MPC point of view, they represent sequences of control actions, one can interpret them in terms of extremal performances of the controlled system (for example in the tracking applications the maximal/minimal admissible setpoint (Olaru and Dumur, 2005)).

3 TOWARDS EXPLICIT SOLUTIONS

In the case of sufficiently large memory resources, construction of the explicit solution for the multiparametric optimization problem can be an interesting alternative to the iterative optimization routines. In this direction recent results where presented at least for the case of linear and quadratic cost functions (see (Seron et al., 2003),(Bemporad et al., 2002),(Goodwin et al., 2004),(Borelli, 2003),(Tondel et al., 2003)). In the following it will be shown that a geometrical approach based on the parameterized polyhedra can bring a useful insight as well.

3.1 Linear Cost Function

The linear cost functions are extensively used in connection with model based predictive control and especially for robust case ((Bemporad et al., 2001), (Kerrigan and Maciejowski, 2004)). In a compact form, the multiparametric optimization problem is:

$$\begin{aligned} &\mathbf{k}_u{}^*(x_t) = \min_{\mathbf{k}_u} f^T \mathbf{k}_u \\ &\text{subject to } A_{in}\mathbf{k}_u \leq B_{in}x_t + b_{in} \end{aligned} \tag{11}$$

The problem deals with a polyhedral feasible domain which can be described as previously in a double representation. Further the explicit solution can be constructed based on the relation between the parameterized vertices and the linear cost function (as in (Leverge, 1994)). The next result resumes this idea.

Proposition: The solution for a multiparametric linear problem is characterized as follows:

a) For the subdomain $\aleph \in \mathbb{R}^n$ where the associated parameterized polyhedron has no valid parameterized vertex the problem is infeasible;

b) If there exists a bidirectional ray $\mathbf{l}$ such that $f^T\mathbf{l} \neq 0$ or a unidirectional ray $\mathbf{r}$ such that $f^T\mathbf{r} \leq 0$, then the minimum is unbounded;

c) If all bidirectional rays $\mathbf{l}$ are such that $f^T\mathbf{l} = 0$ and all unidirectional rays $\mathbf{r}$ are such that $f^T\mathbf{r} \geq 0$ then there exists a cutting of the parameters in zones where the parameterized polyhedron has a regular shape $\bigcup_{j=1\ldots\rho} R_j = \mathbb{R}^n - \aleph$. For each region R_j the minimum is computed with respect to the given linear cost function and for all the valid parameterized vertices:

$$\underline{m}(x) = \min\left\{ f^T\mathbf{v}_i(x) | \mathbf{v}_i(x) \text{ vertex of } \mathcal{P}(x)\right\} \tag{12}$$

The minimum $\underline{m}(x)$ is attained by constant subsets of parameterized vertices of $\mathcal{P}(x)$ over a finite number of polyhedral zones in the parameters space R_{ij} ($\cup R_{ij} = R_j$). The complete optimal solution of the multiparametric optimization is given for each R_{ij} by:

$$\begin{aligned} &S_{R_{ij}}(x) = conv.hull\{\mathbf{v}_1^*(x),\ldots,\mathbf{v}_s^*(x)\} + \\ &+ cone\{\mathbf{r}_1^*,\ldots,\mathbf{r}_r^*\} + lin.spaceP(\mathbf{p}) \end{aligned} \tag{13}$$

where $\mathbf{v}_i^*$ are the vertices corresponding to the minimum $\underline{m}(x)$ over R_{ij} and $\mathbf{r}_i^*$ are such that $f^T\mathbf{r}_i^* = 0_\square$

This result provides *the entire family of solutions* for the linear multiparametric optimization, even for the cases where this family is not finite (for example there are several vertices attaining the minimum).

Remark: For the regions of the parameters space characterized by the case *(a)*, the set of constraints cannot be fulfilled and the feasible domain is empty.

Remark: If the solution of the optimization problem is characterized by the case *(b)*, then the control law based on such an optimization is not well-posed as the optimal control action needs an infinite energy in order to be effectively applied.

Remark: Due to the fact that the parameterized vertices have a linear dependence on the parameter vector, the explicit solution will be piecewise linear. However, the solution is not unique as it can be seen from the case *(c)* and equation (13) and thus for the practical control purposes a continuous piecewise candidate is preferred, eventually by minimizing the number of partitions in the parameters space.

3.2 Quadratic Cost Function

The case of a quadratic const function is one of the most popular at least for the linear MPC. The explicit solution based on the exploration of the parameters space ((Bemporad et al., 2002), (Borelli, 2003), (Tondel et al., 2003)) is extensively studied lately. Alternative methods based on geometrical arguments or dynamical programming ((Goodwin et al., 2004),

(Seron et al., 2003)) improved also the awareness of the explicit MPC formulations. The parameterized polyhedra can serve as a base in the construction of such explicit solution (Olaru and Dumur,), for a quadratic multiparametric problem:

$$\begin{aligned} &\mathbf{k}_u{}^*(x_t) = \arg\min_{\mathbf{k}_u} \mathbf{k}_u^T H \mathbf{k}_u + 2\mathbf{k}_u{}^T F x_t \\ &\text{subject to } A_{in}\mathbf{k}_u \leq B_{in}x_t + b_{in} \end{aligned} \tag{14}$$

In this case the main idea is to consider the unconstrained optimum:

$$\mathbf{k}_u^{sc}(x_t) = H^{-1}Fx_t$$

and its position with respect to the feasible domain given by a parameterized polyhedron as in (5).

If a simple transformation is performed:

$$\tilde{\mathbf{k}}_u = H^{1/2}\mathbf{k}_u$$

then the isocost curves of the quadratic function are transformed from ellipsoid into circles centered in $\tilde{\mathbf{k}}_u^{sc}(x_t) = H^{-1/2}Fx_t$. Further one can use the Euclidean projection in order to retrieve the multiparametric quadratic explicit solution.

Indeed if the unconstrained optimum $\tilde{\mathbf{k}}_u^{sc}(x_t)$ is contained in the feasible domain $\tilde{\mathcal{P}}(x_t)$ then it is also the solution of the constrained case, otherwise existence and uniqueness are assured as follows:

Proposition: For any exterior point $\tilde{\mathbf{k}}_u(x_t) \notin \tilde{\mathcal{P}}(x_t)$, there exists an unique point characterized by a minimal distance with respect to $\tilde{\mathbf{k}}_u^{sc}(x_t)$. This point satisfies:

$$(\tilde{\mathbf{k}}_u^{sc}(x_t) - \tilde{\mathbf{k}}_u^*(x_t))^T(\tilde{\mathbf{k}}_u - \tilde{\mathbf{k}}_u^*(x_t)) \leqslant 0, \forall \tilde{\mathbf{k}}_u \in \tilde{\mathcal{P}}(x_t) \square$$

The construction mechanism uses the parameterized vertices in order to split the regions neighboring the feasible domain in zones characterized by the same type of projection.

Remark: The use of these geometrical arguments makes the construction of explicit solution to deal in a natural manner with the so-called *degeneracy* (Bemporad et al., 2002). This phenomenon is identified by the parameters' values where the feasible domain changes its shape (the set of parameterized vertices is modified).

4 GENERALIZATION FOR NONLINEAR PROGRAMS

If the feasible domain is described by a mixed linear/nonlinear set of constraints then the convexity properties are lost and a procedure for the construction of exact explicit solutions do not exist for the general case.

4.1 Nonlinear Constraints Handling

As already mentioned, in dealing with the presence of nonlinearities constraints, a special case is when the associated feasible domain is convex. In this case, the main ideas in finding the explicit control laws are the following:

- considering the augmented space (formed by the extended arguments and parameters space $((x, \mathbf{k}_u))$, find a set of points situated on the borders of the feasible domain (points that will correspond to the parameterized vertices in the associated linear feasible domain);

- using this set of extremal points, construct the dual representation in terms of parameterized polyhedra (as a fact, in the presence of linear constraints, this set of points could represent the input to the linear algorithm, as well as a set of linear constraints);

- build the corresponding explicit solution by reporting the unconstrained optimum to these parameterized vertices and their validity domains.

Remark: The solution will be a continuous piecewise affine function in the state vector, due to the nature (convexity) of the feasible domain, and it is obtained by projecting the unconstrained optimum on the linear subset of constraints (associated with the nonlinear ones). In this nonlinear convex case, the precision of the solution is directly dependent of the linearization of the nonlinear constraints using a finite set of points on the frontier of the feasible domain (knowing that the rest of the algorithm retains the qualities of the linear algorithm).

4.1.1 Simple Example of a Multiparametric Nonlinear Program

Consider the discrete-time linear system:

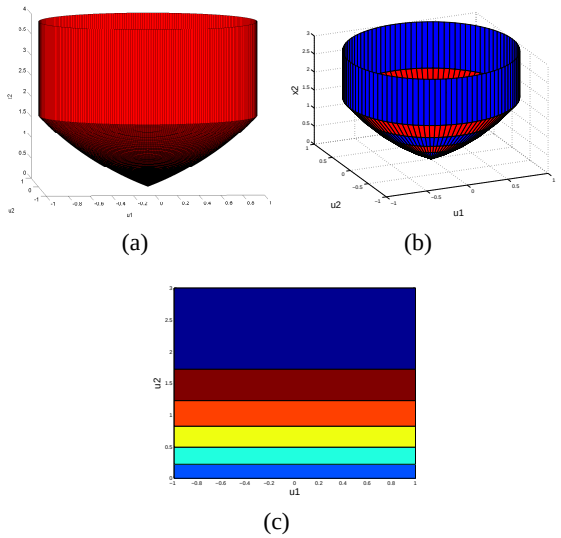
$$x_{t+1} = \begin{bmatrix} 0.9 & 1 \\ 0 & 1 \end{bmatrix} x_t + \begin{bmatrix} 1 \\ -1 \end{bmatrix} u_t \tag{15}$$

and a predictive control law with a prediction horizon of three sampling times and a control horizon of two steps. A nonlinear set of constraints will be also considered:

$$\begin{cases} \sum_{k=0}^{2} u_{t+k}^2 \leq 1 \\ \sum_{k=0}^{2} u_{t+k}^2 \leq \ln(\begin{bmatrix} 0 & 1 \end{bmatrix} x_t + 1) \\ \begin{bmatrix} 0 & 1 \end{bmatrix} x_{t+k} \geq 0; k = 0,1,2 \end{cases} \tag{16}$$

It is obvious that the topology of the feasible domain is changing with the system dynamics, which means that the state vector represents in fact a parameter. More precisely, in our case only the second component of the state, x_t is influencing the shape of the feasible domain and thus one can draw this dependence on the parameter as in figure 1a. Further

this parameterized convex shape can be approximated with a set of parameterized linear inequalities and obtain a double description of a parameterized polyhedron as in figure 1b. A precutting in zones with regular shape (figure 1c) can help in the development of explicit solution due to the important degree of redundancy.



(a) (b)

(c)

Figure 1: (a) The nonlinear dependence of the feasible domain on the parameters (b) The approximation by a parameterized polyhedron; (c) Regions in the parameters' space corresponding to redundancy-free constraints sets.

Finally the nonlinear MPC law for the system (15) and the constraints (16) can be approximated by the explicit solution found in terms of a piecewise linear control law as in figure 2.

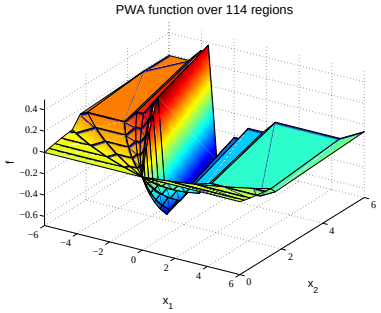


Figure 2: Explicit solution as a piecewise linear function.

4.2 Nonconvex Feasible Domains

In the case when the convexity is lost, one can still construct the convex hull (and an explicit solution) associated with the non-convex domain. From the resulting piecewise affine function one should retain only those regions where the control law is feasible w.r.t. the nonlinear constraints. For the violated constraints a distribution of points will be obtained. Finally the missing regions in the explicit solution will be completed with the Voronoi partition corresponding to these points on the concave border of the feasible domain.

Algorithm:

1. Obtain a set of points ($\mathcal{V}$) on the frontier of the feasible domain D (based on the nonlinear constraints).
2. Considering this set of points, construct the convex $\mathcal{C}_\mathcal{V}$ (which will include the non-convex domain D and some other infeasible domains from the MPC point of view).
3. Split the set $\mathcal{V}$ as $\overline{\mathcal{V}}_L \cup \overline{\mathcal{V}}_{NL} \cup \widetilde{\mathcal{V}} \cup \widehat{\mathcal{V}}$
   - $\widehat{\mathcal{V}} \in \mathfrak{Int}(\mathcal{C}_\mathcal{V})$
   - $\widetilde{\mathcal{V}} \in \mathfrak{F}(\mathcal{C}_\mathcal{V})$ and $\mathcal{C}_\mathcal{V} = \mathcal{C}_{\mathcal{V}\setminus\widetilde{\mathcal{V}}}$
   - $\mathcal{V}_L \in \mathfrak{F}(\mathcal{C}_\mathcal{V})$, $\mathcal{V}_L \cap \widetilde{\mathcal{V}} = \emptyset$ and $\mathcal{V}_L$ saturate at least one linear constraint in the MPC constraints;
   - $\mathcal{V}_{NL} \in \mathfrak{F}(\mathcal{C}_\mathcal{V})$ and $\mathcal{V}_{NL}$ saturate no linear constraint
4. $\mathcal{C}_\mathcal{V}$ is described in the dual representation by the intersection of halfspaces $\mathcal{H}$ (which represent the faces of the convex hull $\mathcal{C}_\mathcal{V}$). Split this set in $\overline{\mathcal{H}} \cup \widehat{\mathcal{H}}$
   - $\widehat{\mathcal{H}} \subset \mathcal{H}$ such that $\exists x \in \mathcal{C}_\mathcal{V}$ with $\mathfrak{Sat}(\widehat{\mathcal{H}}, x) \neq \emptyset$ and $\mathfrak{B}(\mathfrak{R}_{NL}, x) \neq \emptyset$
   - $\overline{\mathcal{H}} = \mathcal{H} \setminus \widehat{\mathcal{H}}$
5. Compute the unconstrained optimum $\mathbf{k}_u^*$
6. Project the unconstrained optimum on $\mathcal{C}_\mathcal{V}$:

$$\mathbf{k}_u^* \leftarrow Proj_{\mathcal{C}_\mathcal{V}}\{-c\}$$

7. If $\mathbf{k}_u^*$ saturates a subset of constraints $\mathcal{K} \subset \widehat{\mathcal{H}}$
   (a) Retain the set of points:

$$S = \Big\{ v \in \widehat{\mathcal{V}} \,|\, \forall x \in \mathcal{C}_\mathcal{V} \; s.t. \; \mathfrak{Sat}(\widehat{\mathcal{H}}, x) = \mathcal{K}; \; \mathfrak{B}(\mathfrak{R}_{NL}, x) = \mathfrak{Sat}(\mathfrak{R}_{NL}, v) \Big\}$$

   (b) Construct the Voronoi partition for the colection of points in S
   (c) Position $\mathbf{k}_u^*$ w.r.t. this partition and map the suboptimal solution $\mathbf{k}_u^* \leftarrow v$ where v is the vertex corresponding to the active region

8. If the quality of the solution is not satisfactory, improve the distribution of the points $\mathcal{V}$ by augmenting the resolution around $\mathbf{k}_u^*$ and restart from the step 2.

The following notations where used:

$\mathfrak{F}(X)$ The frontier of a compact set X

$\mathfrak{Int}(X)$ The interior of a compact set X

$\mathfrak{R}_L(D)$ The set of linear constraints in the definition of the feasible domain D

$\mathfrak{R}_{NL}(D)$ The set of nonlinear constraints in the definition of the feasible domain D

$\mathfrak{Sat}(\mathfrak{R}_*, x)$ The subset of constraints in $\mathfrak{R}_*$ (either $\mathfrak{R}_L$ either $\mathfrak{R}_{NL}$) saturated by the vector x

$\mathfrak{B}(\mathfrak{R}_*, x)$ The subset of constraints in $\mathfrak{R}_*$. violated by the vector x

4.2.1 Numerical Example

Consider the MPC problem implemented using the first control action of the optimal sequence:

$$k_u^* = \arg\min_{k_u} \sum_{i=0}^{N_y-1} x_{t+k|t}^T Q x_{t+k|t} + u_{t+k|t}^T R u_{t+k|t} + \\ + x_{t+N_y|t}^T P x_{t+N_y|t} \tag{17}$$

with

$$Q = \begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix}; R = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}; P = \begin{bmatrix} 13.73 & 2.46 \\ 2.46 & 2.99 \end{bmatrix}$$

subject to

$$\begin{cases} x_{t+k+1|t} = \overbrace{\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}}^{A} x_{t+k|t} + \overbrace{\begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix}}^{B} u_{t+k|t} \;\; \mathrm{k} \geqslant 0 \\ \begin{bmatrix} -2 \\ -2 \end{bmatrix} \leqslant u_{t+k|t} \leqslant \begin{bmatrix} 2 \\ 2 \end{bmatrix} \;\; 0 \leqslant \mathrm{k} \leqslant N_y - 1 \\ (u_{t+k|t}^1)^2 + \left(u_{t+k|t}^2 - 2\right)^2 \geqslant \sqrt{3} \; 0 \leqslant \mathrm{k} \leqslant N_y - 1 \\ (u_{t+k|t}^1)^2 + \left(u_{t+k|t}^2 + 2\right)^2 \geqslant \sqrt{3} \; 0 \leqslant \mathrm{k} \leqslant N_y - 1 \\ u_{t+k|t} = \underbrace{\begin{bmatrix} 0.59 & 0.76 \\ -0.42 & -0.16 \end{bmatrix}}_{K_{LQR}} x_{t+k|t} \;\; N_u \leqslant k \leqslant N_y - 1 \end{cases}$$

One can observe the presence of both linear and nonlinear constraints. By following the previous algorithm, in the first stage, the partition of the state space is performed by considering only the linear constraints (figure 3).

Each such region correspond with a specific projection law. By simply verifying the regions where this projection law obey the nonlinear constraints, the exact part of the explicit solution is obtained (fig. 4).

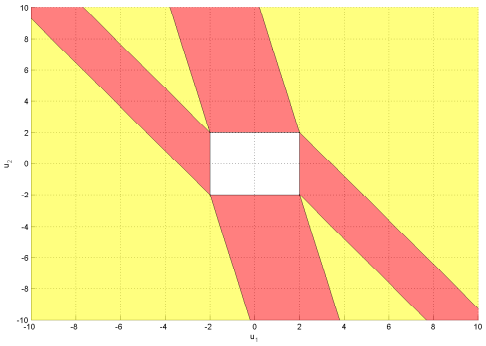


Figure 3: Partition of the arguments space (linear constraints only).

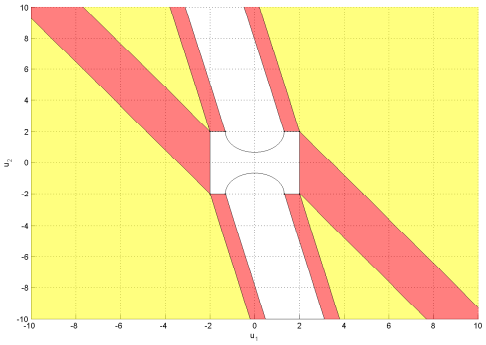


Figure 4: Retention of the regions with feasible linear projections.

Further, a distribution of points on the nonlinear frontier of the feasible domain has to be obtained and based on this distribution of points the associated Voronoi partition constructed. By superposing it to the regions not covered (white zones in the figure 4) at the previous step, one obtain a complete covering of the arguments space. Figure 5 depicts such a complete partition for distribution of 10 points for each nonlinear constraint.

By correspondence, the figure 6 describes the partition of the state space for the explicit solution.

Finally the complete explicit solution is depicted in figure 7. The discontinuities are observable in the regions generated upon the Voronoi partition. In order to give an image of the complexity it can mentioned that the explicit solutions contains 31 regions the computational effort was less than 2*s* mainly spent in the construction of the Voronoi partition.

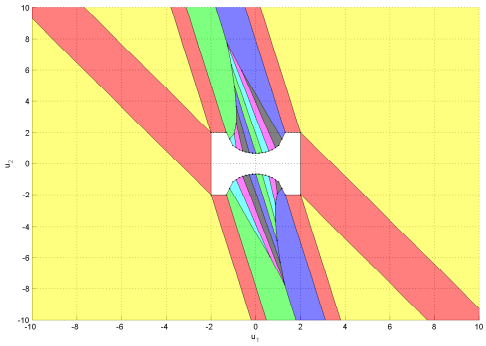


Figure 5: Partition of the arguments space (nonlinear case) - 10 points per nonlinear constraint.

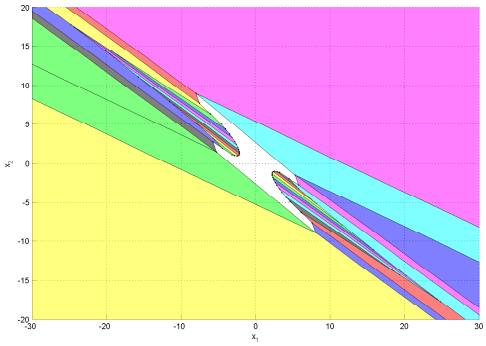


Figure 6: Partition of the state space - 10 points per nonlinear constraint.

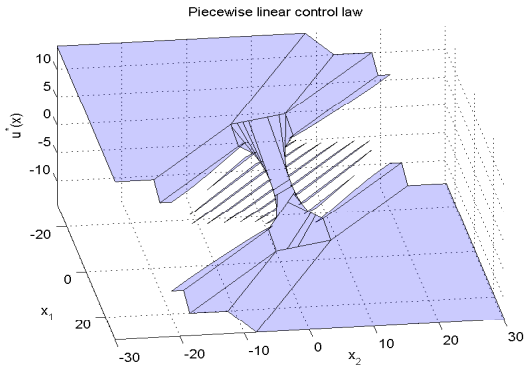


Figure 7: Explicit control law - 10 points per nonlinear constraint.

5 CONCLUSION

The parameterized polyhedra offer a transparent characterization of the MPC degrees of freedom. Once the complete description of the feasible domain as a parameterized polyhedron is obtained explicit MPC laws can be constructed using the projection of the unconstrained optimum. The topology of the feasible domain can lead to explicit solution even if nonlinear constraints are taken into consideration. The price to be paid is found in the degree of suboptimality.

REFERENCES

Bemporad, A., Borrelli, F., and Morari, M. (2001). Robust Model Predictive Control: Piecewise Linear Explicit Solution. In *European Control Conference*, pages 939–944, Porto, Portugal.

Bemporad, A., Morari, M., Dua, V., and Pistikopoulos, E. (2002). The Explicit Linear Quadratic Regulator for Constrained Systems. *Automatica*, 38(1):3–20.

Borelli, F. (2003). *Constrained Optimal Control of Linear and Hybrid Systems*. Springer-Verlag, Berlin.

Goodwin, G., Seron, M., and Dona, J. D. (2004). *Constrained Control and Estimation*. Springer, Berlin.

Kerrigan, E. and Maciejowski, J. (2004). Feedback min-max model predictive control using a single linear program: Robust stability and the explicit solution. *International Journal of Robust and Nonlinear Control*, 14(4):395–413.

Leverge, H. (1994). A note on chernikova's algorithm. In *Technical Report 635*. IRISA, France.

Loechner, V. and Wilde, D. K. (1997). Parameterized polyhedra and their vertices. *International Journal of Parallel Programming*, V25(6):525–549.

Motzkin, T.S., R. H. T. G. and R.M., T. (1953). *The Double Description Method, republished in* Theodore S. Motzkin: Selected Papers, *(1983)*. Birkhauser.

Olaru, S. and Dumur, D. (2005). Compact explicit mpc with guarantee of feasibility for tracking. In *44th IEEE Conference on Decision and Control, and European Control Conference. CDC-ECC '05.*, pages 969–974.

Olaru, S. B. and Dumur, D. A parameterized polyhedra approach for explicit constrained predictive control. In *43rd IEEE Conference on Decision and Control, 2004. CDC.*, pages 1580–1585 Vol.2.

Schrijver, A. (1986). *Theory of Linear and Integer Programming*. John Wiley and Sons, NY.

Scokaert, P. O., Mayne, D. Q., and Rawlings, J. B. (1999). Suboptimal model predictive control (feasibility implies stability). In *IEEE Transactions on Automatic Control*, volume 44, pages 648–654.

Seron, M., Goodwin, G., and Dona, J. D. (2003). Characterisation of receding horizon control for constrained linear systems. In *Asian Journal of Control*, volume 5, pages 271–286.

Tondel, P., Johansen, T., and Bemporad, A. (2003). Evaluation of piecewise affine control via binary search tree. *Automatica*, 39:945–950.

Wilde, D. (1993). A library for doing polyhedral operations. In *Technical report 785*. IRISA, France.

PROCESS CONTROL USING CONTROLLED FINITE MARKOV CHAINS WITH AN APPLICATION TO A MULTIVARIABLE HYBRID PLANT

Enso Ikonen
University of Oulu, Department of Process and Environmental Engineering, Systems Engineering Laboratory
4PYOSYS, FIN-90014 Oulun yliopisto, Finland
Enso.Ikonen@oulu.fi

Keywords: Markov decision process, generalized cell-to-cell mapping, qualitative modelling.

Abstract: Predictive and optimal process control using finite Markov chains is considered. A basic procedure is outlined, consisting of discretization of plant input and state spaces; conversion of a (a priori) plant model into a set of finite state probability transition maps; specification of immediate costs for state-action pairs; computation of an optimal or a predictive control policy; and, analysis of the closed-loop system behavior. An application, using a MATLAB toolbox developed for MDP-based process control design, illustrates the approach in the control of a multivariable plant with both discrete and continuous action variables. For problems of size of practical significance (thousands of states), computations can be performed on a standard office PC. The aim of the work is to provide a basic framework for examination of nonlinear control, emphasizing in on-line learning from uncertain data.

1 INTRODUCTION

For identification and control of stochastic nonlinear dynamic systems, no general method exists. Development of physical models is typically far too time consuming and knowledge intensive, and results in models that are not well suited for process control design. Instead, in industrial practice, linear approximations have turned out to be most useful, commonly extended by considering local linear approximations (cf. gain scheduling, indirect adaptive control, piecewise (multi)linear multimodel systems, etc.), where linear descriptions vary with state and/or time.

For nonlinear plant identification, a multitude of efficient methods exists (Ikonen and Najim, 2002). These include polynomial functions, neural nets, etc. Identification of nonlinear dynamical relations is more difficult. Time series approaches (NARMAX, etc.) are a common and straightforward framework for extending to dynamic nonlinear systems. They rely in that mapping past (delayed) signals through a nonlinear static function will enable capturing the system dynamics. The properties of these models are, in general, difficult to analyze, however; complicating significantly the control design. A common simplification is that of Wiener and Hammerstein systems: to consider static process nonlinearities only and equip the static nonlinear model with separate linear dynamics. This is a powerful approach in that the control design can be largely based on linear analysis, and in that knowledge on plant static nonlinearities can be exploited in the development of plant control. However, only linear dynamics can be dealt with.

This paper focuses on an approach that can cope with a large class of nonlinear systems: the finite Markov chains (Puterman, 1994) (Häggström, 2002) (Poznyak et al., 2000). The basic idea is simple. The system state space is quantized (discretized, partitioned, granulated) into a finite set of states (cells), and the evolution of system state in time is mapped in a probabilistic (frequentist) manner. With controlled Markov chains, the mappings are constructed from each state-action pair. Once equipped with such a model, a control action for each state can be deduced by minimizing a cost function defined in a future horizon, based on specification of immediate costs for each state-action pair. Specification of immediate costs allows versatile means for characterising the desired control behavior. Dynamic programming, studied in the field of Markov decision processes (MDP),

offers a way to solve various types of expected costs. Since a process model is available, the paradigm of model predictive control can also be used to derive the desired controls.

As the basic ideas are old, well-known, and widespread, relevant literature can be found from many fields and under different keywords: generalized cell-to-cell mapping (Hsu, 1987), qualitative modelling (Lunze et al., 2001), and reinforcement learning (Kaelbling et al., 1996), for example. Much of the terminology in the sections that follow originate from (Hsu, 1987): we refer to mappings between cells as (simple or) generalized cell maps, we use a sink cell, etc.

As pointed out in (Lee and Lee, 2004) (see also (Ikonen, 2004) (Negenborn et al., 2005)), applications of MDP in process control have been few; instead, the model predictive control paradigm is very popular in the process control community. Whereas not-so-many years ago the computations associated with finite Markov chains were prohibitive, the computing power available using cheap office-pc's enables the re-exploration of these techniques.

The work described in this paper aims at building a proper basic framework for examining the possibilities of controlled finite Markov chains in nonlinear process control. A majority of current literature on MDP examines means to overcome the problem of curse-of-dimensionality, e.g., by means of function approximation (neuro-dynamic programming, Q-learning, etc.). The main obstacle in such approaches is in that the unknown properties introduced by the mechanisms of function approximation make void the fundamental benefit of applying finite Markov chains: a straightforward and elegant means to describe and analyse the dynamic characteristics of a stochastic nonlinear system. Recall how linear systems are limited by the extremely severe assumption of linearity (affinity), yet they have turned out to be outmost useful for control design purposes. In a similar way, the finite Markov chains are fundamentally limited by the resolution of the problem presentation (discretization of state-action spaces). The hypothesis of this work is that keeping in mind this restriction (just as we keep in mind the assumption of linearity) the obtained results can be most useful. The practical validity of this statement is in the focus of the research.

In particular, the field of process engineering is in our main concern, with applications characterized by: availability of rough process models, slow sampling rates, nonlinearities that are either smooth or appear as discontinuities, expensive experimentation (large-scale systems running in production), and substantial on-site tuning due to uniqueness of products. Clearly, this type of requirements differ from those encountered, e.g., in the field of economics (lack of reliable models), robotics (very precise models are available), consumer electronics (mass production of low cost products), telecommunication (extensive use of test signals, fast sampling), or academic toy problems (ridiculously complex multimodal test functions).

Due to systematical errors, noise, and lack of accuracy in measurements of process state variables, among many other reasons, there is a urgent need for extended means of learning and handling of uncertainties. The finite Markov chains provide straightforward means for dealing with both of these issues.

This paper is organized as follows: The process models are considered in section 2, control design in section 3, open and closed loop system analysis in section 4. The MATLAB toolbox, and an illustrative example is provided in section 5. Discussion on aspects relevant to learning under uncertainties, and conclusions, are given in the final section.

2 GENERALIZED CELL MAPPING

Let the process under study be described by the following discrete-time dynamic system and measurement equations

$$\mathbf{x}(k) = f(\mathbf{x}(k-1),\mathbf{u}(k-1),\mathbf{w}(k-1)) \quad (1)$$

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

where $f : \Re^{n_\mathrm{x}} \times \Re^{n_\mathrm{u}} \times \Re^{n_\mathrm{w}} \to \Re^{n_\mathrm{x}}$ and $h : \Re^{n_\mathrm{x}} \times \Re^{n_\mathrm{v}} \to \Re^{n_\mathrm{y}}$ are nonlinear functions, $w_k \in \Re^{n_\mathrm{w}}$ and $v_k \in \Re^{n_\mathrm{v}}$ are i.i.d. white noise with probability density functions (pdf's) p_w and p_v. The initial condition is known via $p_\mathrm{X}(0)$.

Let the state space be partitioned into a finite number of sets called state cells, indexed by $s \in \mathcal{S} = \{1,2,...,S\}$. The index s is determined from

$$s = \arg\min_{s\in\mathcal{S}} \left\| \mathbf{x} - \mathbf{x}_s^{\mathrm{ref}} \right\|$$

where $\mathbf{x}_s^{\mathrm{ref}}$ are reference points (e.g., cell centers). In addition, let us define a 'sink cell', s_{sink}; a state is categorized into a sink cell if $\min_{s\in\mathcal{S}} \left\| \mathbf{x} - \mathbf{x}_s^{\mathrm{ref}} \right\| > \mathbf{x}^{\mathrm{lim}}$. Similarly, let the control action and measurement spaces be partitioned into cells indexed by $a \in \mathcal{A} = \{1,2,...,A\}$ and $m \in \mathcal{M} = \{1,2,...,M\}$, respectively, and determined using reference vectors $\mathbf{u}_a^{\mathrm{ref}}$ and $\mathbf{y}_m^{\mathrm{ref}}$. The partitioning results in $\mathcal{X} = \cup_{s=1}^{S} \mathcal{X}_s$, $\mathcal{U} = \cup_{a=1}^{A} \mathcal{U}_a$ and $\mathcal{Y} = \cup_{m=1}^{M} \mathcal{Y}_m$.

The evolution of the system can now be approximated as a finite state controlled Markov chain over

the cell space (however, see (Lunze, 1998)). In simple cell mapping (SCM), one trajectory is computed for each cell. Generalized cell mapping (GCM) considers multiple trajectories starting from within each cell, and can be interpreted in a probabilistic sense as a finite Markov chain.

2.1 Evolution of States

Let the state pdf be approximated as a $S \times 1$ cell probability vector $\mathbf{p}_{\mathrm{X}}(k) = [p_{\mathrm{X},s}(k)]$ where $p_{\mathrm{X},s}(k)$ is the cell probability mass. The evolution of cell probability vectors is described by a Markov chain represented by a set of linear equations

$$\mathbf{p}_{\mathrm{X}}(k+1) = \mathbf{P}^{a(k)}\mathbf{p}_{\mathrm{X}}(k)$$

or, equivalently,

$$p_{\mathrm{X},s'}(k+1) = \sum_{s\in\mathcal{S}} p^a_{s',s}\, p_{\mathrm{X},s}(k)$$

where $\mathbf{P}^a$ is the transition probability matrix under action a, $\mathbf{P}^a = \left[p^a_{s',s}\right]$.

3 CONTROL DESIGN

Using a GCM model of the plant, an optimal control action for each state can be solved by minimizing a cost function. In both optimal (Kaelbling et al., 1996) and predictive control (Ikonen and Najim, 2002) the cost function is defined in a future horizon, based on specification of immediate costs for each state-action pair. Whereas optimal control considers (discounted) infinite horizons and solves the problem using dynamic programming, nonlinear predictive control approaches rely on computation of future trajectories (predictions) and exhaustive search.

3.1 Optimal Control

In optimal control, the control task is to find an appropriate mapping (optimal policy or control table) π from states ($\mathbf{x}$) to control actions ($\mathbf{u}$), given the immediate costs $r(\mathbf{x}(k),\mathbf{u}(k))$. The infinite-horizon discounted model attempts to minimize the geometrically discounted immediate costs

$$J(\mathbf{x}) = \sum_{k=0}^{\infty} \gamma^k r(\mathbf{x}(k), \pi(\mathbf{x}(k)))$$

under initial conditions $\mathbf{x}(0) = \mathbf{x}$. The optimal control policy π^* is the one that minimizes J. The optimal cost-to-go is given by $J^* = \min_\pi J$.

Bellman's principle of optimality states that

$$J^*(\mathbf{x}) = \min_{\mathbf{u}} [r(\mathbf{x},\mathbf{u}) + \gamma J^*(f(\mathbf{x},\mathbf{u}))]$$

i.e., the optimal solution (value) for state $\mathbf{x}$ is the sum of immediate costs r and the optimal cost-to-go from the next state, $J^*(f(\mathbf{x},\mathbf{u}))$. Application of the Bellman equation leads to methods of dynamic programming.

3.1.1 Value Iteration

In value iteration, the optimal value function is determined by a simple iterative algorithm derived directly from the Bellman equation. Let the immediate costs be given in matrix $\mathbf{R} = [\mathbf{r}^a]$, with column vectors $\mathbf{r}^a = [r^a_s]$, and collect the values of the cost-to-go at iteration i into a vector $\mathbf{J}^*(i) = [J^*_s(i)]$. Given arbitrary initial values $J^*_s(0)$, the costs are updated for $i = 0, 1, 2, ...$:

$$\begin{aligned} Q^a_s(i) &= r^a_s + \gamma \sum_{s'\in\mathcal{S}} p^a_{s',s} J^*_{s'}(i) \\ J^*_s(i+1) &= \min_{a\in\mathcal{A}} Q^a_s(i) \end{aligned}$$

$\forall s, a$, until the values of $J^*_s(i)$ converge. Denote the converged values by J^*_s. The optimal policy is then obtained from

$$\pi^*_s = \arg\min_{a\in\mathcal{A}} \left[r^a_s + \gamma \sum_{s'\in\mathcal{S}} p^a_{s',s} J^*_{s'} \right].$$

3.2 Predictive Control

Given a system model and the associated costs, we can easily set up a predictive control type of a problem. In predictive control, the costs are minimized in an open loop in a fixed horizon

$$\begin{aligned} & J\left(\mathbf{x}(k), ..., \mathbf{x}\left(k+H_{\mathrm{p}}\right), \mathbf{u}(k), ..., \mathbf{u}\left(k+H_{\mathrm{p}}\right)\right) \\ = \;& \sum_{h=0}^{H_{\mathrm{p}}} r(\mathbf{x}(k+h), \mathbf{u}(k+h)) \end{aligned}$$

under initial conditions $\mathbf{x}$. In practice it is useful to introduce a control horizon, where it is assumed that the control action will remain fixed after a given number of steps, H_{c}. Often only one step is allowed and the optimization problem reduces to the minimization of $J\left(\mathbf{x}(k), ..., \mathbf{x}\left(k+H_{\mathrm{p}}\right), \mathbf{u}(k)\right)$.

Under control action a, the costs are given by

$$\begin{aligned} J_a &= \sum_{h=0}^{H_{\mathrm{p}}} [\mathbf{r}^a]^T \mathbf{p}_{\mathrm{X}}(k+h) \\ &= \sum_{h=0}^{H_{\mathrm{p}}} [\mathbf{r}^a]^T [\mathbf{P}^a]^h \mathbf{p}_{\mathrm{X}}(k) \end{aligned}$$

where $\mathbf{r}^a = [r^a_s]$ is a column vector of immediate costs and $\mathbf{p}_{\mathrm{X}}(k)$ is current state cell pdf. In order to solve the problem, it suffices to evaluate the costs for all $a \in \mathcal{A}$ and select the one minimizing the costs. The prediction horizon H_{p} is a useful tuning parameter; a

long prediction horizon leads to mean level type of control.

The control policy mapping $\pi^\diamond$ can be obtained by solving the above problem in each state s and tabulating the results: $\pi_s^\diamond = \arg\min_a J_a$.

For many practical cases, a good controller design can be obtained using either the optimal control approach, or the predictive control approach with $H_c = 1$. In some cases, however, an engineer may be interested in extending the controller design possibilities to larger control horizons. In principle, this is straightforward to realize in the GCM context: One simply creates A different sequences of control actions, simulates the system accordingly, and selects the sequence that minimizes the cost function.

4 SYSTEM ANALYSIS

The generalized cell-to-cell mapping is a powerful tool for analysis of nonlinear systems. In what follows, it is assumed that the system map (Markov chain) is described by transition probabilities $\mathbf{P}$. This may correspond to the process output under a fixed (open loop) control action a ($\mathbf{P} := \mathbf{P}^a$) or the systems closed loop behavior obtained from the construction of transition probabilities under $\mathbf{u} = \pi(\mathbf{x})$: $\mathbf{P} := \mathbf{P}^\pi = \left[p^\pi_{s',s}\right]$.

4.1 Characterization of Cells

In GCM, an useful characterization of cells is obtained by studying the long term behavior of the Markov chain. A state is said to be recurrent (Najim et al., 2004) iff starting at a given state there will be a return to the state with probability 1. Otherwise the state is said to be transient. If $p_{s,s} = 1$, a state is said to be absorbing.

Decomposing the probability vector into recurrent cells ($i_r \in \mathcal{I}_r$) and transient cells ($i_t \in \mathcal{I}_t$), the Markov chain can be written as follows:

$$\begin{bmatrix} \mathbf{p}_r(k+1) \\ \mathbf{p}_t(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{rt} \\ \mathbf{0} & \mathbf{P}_{tt} \end{bmatrix} \begin{bmatrix} \mathbf{p}_r(k) \\ \mathbf{p}_t(k) \end{bmatrix}$$

As $k \to \infty$, the recurrent cells are visited infinitely often, whereas the transient cells are visited only finitely often. Among the recurrent cells, we can further classify the absorbing cells, ($i_a \in \mathcal{I}_a$): $\mathbf{P}_{aa} = \mathbf{I}$. The absorbing states are never left, once visited.

The recurrent cells form communicating classes (closed subsets), where the cells within each communicating class (inter)communicate with each other, i.e., the probability of transition from one state to the other is nonzero, and do not communicate with other states. Each absorbing state only communicates with itself. A closed communicating class constitutes a sub-Markov chain, which can be studied separately.

A stationary probability distribution satisfies $\bar{\mathbf{p}}_X = \mathbf{P}\bar{\mathbf{p}}_X$ and, consequently, the distribution must be an eigenvector of $\mathbf{P}$; for the distribution to be a probability distribution, the eigenvalue must be one. Therefore, the recurrent cells are found by searching for the unit amplitude eigenvalues of $\mathbf{P}$; the nonzero elements of the associated eigenvectors $\bar{\mathbf{p}}_X$ point to the recurrent cells.

4.2 Stability and Size of Basin-of-attraction

Examination of the behavior of transient cells as they enter the recurrent cells reveals the dynamics of the nonlinear system. We have that

$$\begin{aligned} \mathbf{p}_r(k+1) &= \mathbf{P}_{rr}\mathbf{p}_r(k) + \mathbf{P}_{rt}\mathbf{p}_t(k) \\ &= \mathbf{P}_{rr}\mathbf{p}_r(k) + \mathbf{P}_{rt}\mathbf{P}_{tt}^k\mathbf{p}_t(0) \end{aligned}$$

where $\mathbf{P}_{rt}\mathbf{P}_{tt}^k$ represents the conditional probability that a solution starting from a transient cell will pass into an recurrent cell at time $k+1$. The probability that this will eventually happen, $\mathbf{P}_{t2r}$, is given by

$$\mathbf{P}_{t2r} = \sum_{k=0}^{\infty} \mathbf{P}_{rt}\mathbf{P}_{tt}^k = \mathbf{P}_{rt}(1-\mathbf{P}_{tt})^{-1}$$

Each recurrent cell belongs to a communicating class, for absorbing cells this class consists of a single cell. The probability of transition into a particular communicating class is obtained by summing (column-wise) the entries in $\mathbf{P}_{t2r}$.

The sink cell (Hsu, 1987) is an absorbing cell that represents the entire region outside the domain of interest. A nonzero probability to enter the sink cell indicates unstability of the system (given the resolution of the model). In the experimental section, the stationary probabilities of entering the sink cell are examined.

High probability cells determine the basin-of-attraction. The 'size of the basin-of-attraction' for each state was characterized by taking the sum of probabilities for entering a recurrent cell (from any transient cell, or from any recurrent cell) and weighting it with the probability of occurance within a communicating class (i.e., multiplying this with the stationary mapping $\mathbf{P}_\infty$):

$$\mathbf{B} = \mathbf{P}_\infty \left[\sum_{i \in \mathcal{I}_t} [\mathbf{P}_{t2r}]_{j,i} + \sum_{i \in \mathcal{I}_r} [\mathbf{P}_{rr}]_{j,i} \right]$$

where $\mathbf{P}_\infty$ is a mapping to stationary distribution: $\mathbf{P}_\infty = \lim_{n\to\infty} \frac{1}{n}\sum \mathbf{P}_{rr}^n$, and $[\mathbf{x}]_{a,b}$ denotes an element of $\mathbf{x}$ in a'th row and b'th column. Elements of $\mathbf{B}$ take

values in the interval $[0,S]$, $\sum_{i\in I_r}[\mathbf{B}]_i = S$. A large value in $\mathbf{B}$ indicates the presence of the following ingredients: the communicating class to which a recurrent state belongs to can be accessed from a large number of states; the probability of entering the class from these states is high; the stationary probability for the occurance of a recurrent state (within a communicating class) is high. Recall that when working with large state-action spaces, separate examination of all states is hopeless. In the experimental section we hope to bring some light to some control relevant properties of the (open or closed-loop) system by projecting $\mathbf{B}$ to dimensions of $\mathbf{x}$.

Based on the above analysis, some communicating classes can then be taken under closer examination: simulation of (expected or random) trajectories, examination of cells in basins of attraction, etc. In assessing control performance, examination of the speed of the system (lengths of trajectories converging to communicating classes) is of great interest. The absorbtion time from the i'th state to the j'th state ($i \in I_t$, $j \in I_r$), $E\{\mathbf{k}\}$, is obtained from:

$$E\{\mathbf{k}\} = \mathbf{P}_{rt}\sum_{k=0}^{\infty} k\mathbf{P}_{tt}^{k} = \mathbf{P}_{rt}(1-\mathbf{P}_{tt})^{-2}.$$

5 SIMULATION STUDY

In this section some numerical results based on simulations ar given. The following control design problem set-up was envisioned: A nonlinear state-space model of the plant is available (a set of ordinary differential equations, for example), and a decision on input, state, and output variables has been made. A controller is now seeked for, such that desired transitions between plant output set points would be optimal. A typical GCM control design procedure would involve the following (iterative) steps:

- Set model resolution by specifying discretization of plant inputs, states, outputs and output set points; and sampling time.
- Set control targets by specifying immediate costs.
- Build a GCM plant model (by successive evaluations of the original model, and counting the occurred state transitions) and analyze its behavior.
- Design a controller (e.g., optimal or predictive), based on the GCM plant model.
- Build a GCM closed-loop model and analyze its behavior.

5.1 Experiment Setup

Let us consider a simple example of a two-tank MIMO system (see (Åkesson et al., 2006) and references there). In this hybrid system the action space (space of control inputs) consists of both real-valued and discrete-valued variables. The objective is to keep the temperature (T_2) in the second tank at it setpoint, while keeping the levels of both tanks (h_1, h_2) within preset limits. The system is controlled by a valve for the first tank input flow, a pump between the two tanks, a heater in the second tank, and a valve for the second tank output flow. The heater (u_1) is constrained to continuous values in the interval between 0 and 560 kW, the pump (u_2) has three operational levels {off, medium, high}, the valves (u_3, u_4) are binary {on/off}.

The system is described by the mass and energy balances

$$\begin{aligned}
\frac{d}{dt}h_1 &= \frac{1}{A_1}(v_1u_3 - \alpha u_2)\\
\frac{d}{dt}T_1 &= \frac{1}{A_1h_1}(v_2 - T_1)v_1u_3\\
\frac{d}{dt}h_1 &= \frac{1}{A_2}(\alpha u_2 - v_3u_4)\\
\frac{d}{dt}T_2 &= \frac{1}{A_2h_2}(T_1 - T_2)\alpha u_2 + \frac{u_1}{c_l\rho_l}
\end{aligned}$$

where subscript 1 refers to the first tank (buffer), subscript 2 to the second tank (supply); v_1 is the inflow, v_2 the inflow temperature, and v_3 the outflow; A is the tank area ($A_1 = 3.5$ m^2, $A_2 = 2$ m^2), c_l and ρ_l are the liquid specific heat capacity and density ($c_l = 4.2$ $\frac{\text{kJ}}{\text{kg K}}$, $\rho_l = 1000$ $\frac{\text{kg}}{\text{m}^3}$); α is a pump capacity factor ($\alpha = 1$ $\frac{\text{m}^3}{\text{min}}$).

A discrete time Markov model (1)–(2) $\mathbf{x}(k) = f(\mathbf{x}(k),\mathbf{u}(k))$ for the system was constructed by forming the system state and controls as follows: $\mathbf{x}(k)$ = [$h_1(k)$, $T_1(k)$, $h_2(k)$, $T_2(k)$, $v_1(k)$, $v_2(k)$, $v_3(k)$] and $\mathbf{u}(k)$ = [$u_1(k), u_2(k)$, $u_3(k), u_4(k)$], $y(k) = x_4(k)$. Since the model is based on another (deterministic) model we omit the disturbances here, $\mathbf{w} \equiv \mathbf{v} \equiv \mathbf{0}$ in (1)–(2). The state space was discretized by forming a grid, where tank levels and temperatures were quantized as follows:

$$\begin{aligned}
x_1^{\text{ref}} &= x_3^{\text{ref}} = \{0,1,2,...,9\}\ \text{[m]}\\
x_2^{\text{ref}} &= \{17,18,19\},\ x_4^{\text{ref}} = \{17,18,...,24\}\ \text{[C]}
\end{aligned}$$

disturbances into one and three values:

$$v_1 = v_3 = \{1\},\ v_2 = \{17,18,19\},$$

and heating action in five values :

$$u = \{0,140,...,560\}\ \text{[kW]}.$$

Roughly, the above states that deviations less than 0.5 °C in the supply tank temperature are out of our interest. Since we also want to place constraints on

high and low tank levels, with the above discretization we can set the switching point from allowed to non-desirable to 0.5 and 8.5 m. The quantizations in the disturbances allow to take known (step-wise) disturbances into account when deriving the optimal controls. The immediate costs were set based on the Euclidean norm between desired and reference temperatures, $\|w - y_s\|$ and deviation from nominal controls for u_2, u_3 and u_4 at 1 with weights 0.1, 2 and 2 respectively (see (Åkesson et al., 2006)). For the states where reference points exeeceded either upper or lower limits for the tank level, an additional large cost was added (ten times larger than largest cost so far). A 100 times larger cost was set for the sink cell.

5.2 Computer Simulations

The selected discretization results in a finite state–action space of 7201 states and 60 actions, including the sink cell. For the considered computational platform (a standard office PC: 3GHz Pentium 4 CPU, 1GB RAM, MATLAB R12) this posed no problem. In fact, up to ten times larger state–action spaces were experimented successfully.

A GCM model was built by evaluating the state transitions five hundered times for each possible state-action pair (s,a). The starting state was generated from a random uniform distribution from within the state hypercube. This resulted in roughly 2×10^8 evaluations of the plant model; within each evaluation the ode were solved one sampling time (15 seconds) ahead using a standard ode-solver (MATLAB ode23). While most of the computing time was spent on solving the ode, the computations took a couple of hours. Clearly this presented a significant burden both in terms of computing power and memory, but not excessive at all. As long as the discretization of the state-action spaces is kept fixed during latter stages of control design, the model need not be re-evaluated, even if other parameters such as the immediate costs ($\mathbf{R}$) or controller design parameters (H_p or γ) would be modified.

Given the GCM model, a predictive controller was designed using $H_\mathrm{p} = 5$. Using the obtained control policy $\pi^\diamond$, a closed loop GCM map was constructed.

The stability of the closed loop system is revealed by examination of the probabilities of entering the sink cell. The probabilities for entering the sink cell from any other cell were zero. Consequently, the closed loop system was stable for all initial states and set points.

Figure 1 illustrates the sizes of the basins-of-attraction, projected to three different dimensions of $\mathbf{x}$: level of tank 1 (x_1), level of tank 2 (x_3), and temperature of tank 2 (x_4). The bars in the plots show the size of basin-of-attraction (as a percentage of whole state space), i.e., the cumulative number of states that are mapped to a recurrent state, recurrent states being sorted according to the their projection to dimension of x. From top plots in Fig 1, it can be immediately observed that the constraints on tank levels are fulfilled: the basin-of-attraction is empty for projections to levels 0 or 9, to both tanks.

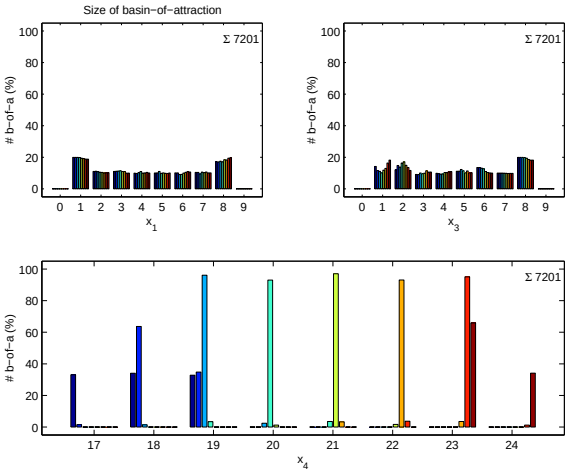


Figure 1: Size of basin-of-attraction. The probability mass for entering a particular state, projected to dimensions of x. Top plots: tank levels, bottom plot: tank 2 temperature.

The projections to tank 2 temperature, see bottom plot in Fig 1, show the success (controllability) of the plant to its set point (in steady state). For set points 19°C... 23°C, almost 100% success is obtained. For low temperatures, the smaller sizes of basins-of-attraction are explained by the lack of means to cool the incoming feed. Since one third of the states characterizes states with input feed equal to 19°C, one third in 18°C and one third in 17°C, it is easy to understand that the setpoint of 17 degrees is attained only when the input feed is 17°C, etc.

In few cases (three initial states), the model predicted transitions that could be judged as impossible using physical arguments. Closer examination of the plant model statistics revealed that these cases were due to the random sampling when building the GCM model, and the slow dynamics of the system. For example, an input feed in 19°C with no heating resulted the 18°C steady state temperature in both tanks. Examining the plant model, it was straightforward to attribute this to the fact that during the 500 simulations, none of the simulated trajectories had lead to another discrete state. Consequently, this state was categorized as absorbing. The remedy for this problem is to increase either the sampling rate, or the number of evaluations.

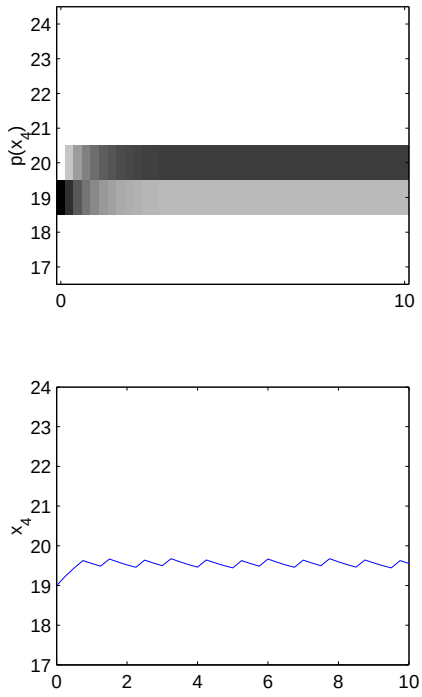


Figure 2: A communicating class consisting of two states. The closed loop system is 'ringing'.

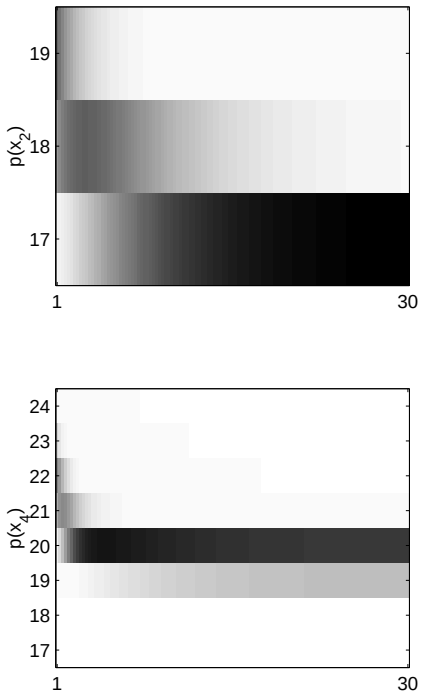


Figure 3: Evolution of probability distribution. The expected time of absorbtion is 9 minutes.

For set points 19°C... 23°C, full 100% sizes of basin-of-attraction were not obtained. Instead, from bottom plot in Fig 1 it can be observed that a small percentage of the probability mass is distributed to the neighboring projections. A closer examination of these cases reveals a typical reason for this. Top plot in Fig. 2 shows the evolution of state in closed-loop from a particular initial state. The top plot shows the state probabilities projected towards the dimension x_4 (tank 2 temperature), for the case of setpoint in 20°C. As suggested by the plot, and as can be detected by examination of the communicating classes, the stationary distribution is a communicating class formed by two states. A sample trajectory is illustrated in the bottom plot, Fig. 2, showing that a phenomenon of 'ringing' clearly takes place. Most of the time is spent in the desired set point, but occasionnally the system crosses the border and visits the state 19°C.

For a particular communicating class, or a state in it, very precise information can be obtained. For example, for the closed-loop system with set point in 20°C there were 182 communicating classes of which 67 were absorbing. For the 'ringing'-class example the basin-of-attraction contained 201 states (there was a nonzero probability of entering this class from 201 states). The expected time of absorbtion within these states ranged from 0 (from the recurrent states) to 8.85 minutes. Evolution of the transition probability distribution (projected to two dimensions of $\mathbf{x}$) from the slowest initial state is shown in Fig. 3, confirming the exactness of this result. Unfortunately, it is not feasible to examine all states with this much care.

As a final example of system analysis, let us examine if there were any cases when a plant shutdown would happen (heating off, pump off, valves off). Examination of the policies (for all setpoints) revealed that only the sink cell resulted in plant shutdown. Another set of interesting control commands could be the one characterized by high pumping and closing of the output valve. It was observed that for each set point, the policy table contained roughly 800 occurances of this control (i.e., in 800 out of 7201 states, this was the control to apply). These states were characterized by low levels in tank 2. Again, this appeared to make sense from an engineering point of view.

It can be concluded that convenient tools for analysis of the closed loop system were found, including examination of stability and steady state performance. The characterization of system performance in terms of speed of response was more tedious. It is not clear what could be done, as –for a nonlinear system– behavior differs from state to state, and computation of expectations and worst case scenarios does not necessarily reveal feasible information. A partial remedy is provided by the simple –and extremely commonly used– approach of simulating the closed loop

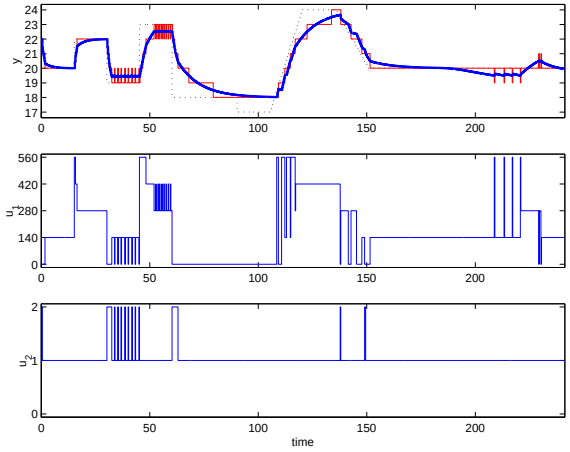


Figure 4: Closed-loop simulation. The set point trajectory consists of a series of steps and ramps. A the end of simulation, an input disturbance affects the system.

system under typical operating conditions. Figure 4 illustrates a trajectory following simulation with (a known) input disturbance. It can be concluded that the behavior of the closed loop system is adequate.

6 DISCUSSION AND CONCLUSIONS

In this preliminary work we have focused on using Markov chains and MDP as process control design tool, to be used bearing in mind the resolution of the problem set up (discretization into a finite state space). Continuing in the same direction, the problem of identication is then related to keeping the original model up-to-date (the ode, for example), or –at least– approximating the original model using function approximation techniques, rather than looking for clever tricks to make counting feasible in the finite state space, doomed to be huge. If doable, the benefits are clear: physical interpretation of estimated parameters. In many process engineering problems, this may turn out to be more fruitful than pure machine learning approaches.

Instead, the problem of uncertainty in measurements can potentially be handled in a very elegant and efficient fashion using finite Markov chains (Ikonen, 2004). Given the finite state probabilistic description of the plant, it is straightforward to construct cost functions taking into account the uncertainty in the predictions (other than discounted conditional expectations). Under the predictive control paradigm, also uncertainties in current state can be taken into account in plant predictions (i.e., there's no need to restrict to ML estimates, etc.). Conducting a literature review on these topics is a major direction in our future research.

REFERENCES

Åkesson, B. M., Nikus, M. J., and Toivonen, H. T. (2006). Explicit model predictive control of a hybrid system using support vector machines. In *Proceedings of the 1st IFAC Workshop on Applications of Large Scale Industrial Systems (ALSIS'06)*, Helsinki-Stockholm, Finland-Sweden.

Häggström, O. (2002). *Finite Markov Chains and Algorithmic Applications*. Cambridge University Press, Cambridge.

Hsu, C. S. (1987). *Cell-to-Cell Mapping - A Method of Global Analysis for Nonlinear Systems*. Springer-Verlag, New York.

Ikonen, E. (2004). Learning predictive control using probabilistic models. In *IFAC Workshop on Advanced Fuzzy/Neural Control (AFNC'04), Oulu, Finland*.

Ikonen, E. and Najim, K. (2002). *Advanced Process Identification and Control*. Marcel Dekker, New York.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Lee, J. M. and Lee, J. H. (2004). Approximate dynamic programming strategies and their applicability for process control: A review and future directions. *International Journal of Control, Automation, and Systems*, 2(3):263–278.

Lunze, J. (1998). On the Markov property of quantised state measurement sequences. *Automatica*, 34(11):1439–1444.

Lunze, J., Nixdorf, B., and Richter, H. (2001). Process supervision by means of a hybrid model. *Journal of Process Control*, 11:89–104.

Najim, K., Ikonen, E., and Ait-Kadi, D. (2004). *Stochastic Processes - Estimation, Optimization and Analysis*. Kogan Page Science, London.

Negenborn, R. R., De Schutter, B., Wiering, M. A., and Hellendoorn, H. (2005). Learning-based model predictive control for Markov decision processes. In *16th IFAC World Congress*.

Poznyak, A. S., Najim, K., and Gómez-Ramírez, E. (2000). *Self-Learning Control of Finite Markov Chains*. Marcel Dekker, New York.

Puterman, M. L. (1994). *Markov Decision Processes – Discrete Stochastic Dynamic Programming*. Wiley et Sons, New York.

TRACKING CONTROL OF WHEELED MOBILE ROBOTS WITH A SINGLE STEERING INPUT

Control Using Reference Time-Scaling

Bálint Kiss and Emese Szádeczky-Kardoss
Department of Control Engineering and Information Technology
Budapest University of Technology and Economics
Magyar Tudósok krt. 2, Budapest, Hungary
bkiss@iit.bme.hu, kardoss@seeger.iit.bme.hu

Keywords: Time-scaling, wheeled mobile robot, flatness, motion planning, tracking control.

Abstract: This paper presents a time-scaling based control strategy of the kinematic model of wheeled mobile robots with one input which is the steering angle. The longitudinal velocity of the mobile robot cannot be influenced by the controller but can be measured. Using an on-line time-scaling, driven by the longitudinal velocity of the robot and its time derivatives, one can achieve exponential tracking of any sufficiently smooth reference trajectory with non-vanishing velocity. The price to pay is the modification of the traveling time along the reference trajectory according to the time-scaling. The measurement of the time derivatives of the velocity is no longer necessary if the tracking controller is designed to the linearized tracking error dynamics.

1 INTRODUCTION

The kinematic model of a wheeled mobile robot (WMR) has generally two inputs namely the longitudinal velocity of the rear axis midpoint and the steering angle of the front wheels. Several strategies are applied to control such WMRs with these two inputs including the tracking error transformation based control reported by (Dixon et al., 2001), the sliding mode controller based solution proposed by (Benalia et al., 2003), and the behavior based control strategy studied by (Gu and Hu, 2002). An important property of the model is its differential flatness (Fliess et al., 1995; Fliess et al., 1999) implying its dynamic feedback linearizability (with a singularity at zero velocities).

However, situations may occur where the longitudinal velocity of the WMR is not generated by a feedback controller, but by an external source. A practical example of this scenario is the tracking problem related to a passenger car without automatic gear. In such a situation the human driver needs to generate the velocity of the car with an appropriate management of the gas, clutch, and break pedals while the tracking controller may influence only the angle of the steered wheels. The kinematic model obtained in such a situation is no longer differentially, but orbitally flat (Respondek, 1998; Guay, 1999), since one of the inputs is lost.

The control problem is still the tracking of the reference trajectory but this tracking may become impossible if the velocity of the reference WMR moving along the reference path is always superior to the real velocity generated by the driver. The opposite is also possible such that the velocity of the reference WMR is always inferior to the real velocity generated by the driver. Nevertheless it is expected that the path of the controlled WMR joins the path of the reference WMR for any velocity profile generated by the driver. To achieve exponential tracking in both cases, this paper suggests a time-scaling of the reference path. This time-scaling uses the measurement of the velocity generated by the driver and eventually its time derivatives. A practical mean to obtain these measurements is the use of the ABS signals available on the CAN bus of the vehicle or the use of alternative sensors (e.g. accelerometers).

Recall that time-scaling is a commonly used concept to find optimal trajectories, to cope with input saturation, to reduce tracking errors, and to establish equivalence classes of dynamical systems.

One may use off-line time-scaling methods to find the time optimal trajectories for robot manipulators (Hollerbach, 1984) or for autonomous mobile vehicles (Cuesta and Ollero, 2005). The problem with these off-line methods is that no sufficient control input margins are always assured for the closed loop

control during the tracking. Other algorithms use therefore on-line trajectory time-scaling for robotic manipulators to change the actuator boundaries such that sufficient margin is left for the feedback controller (Dahl and Nielsen, 1990).

Another concept is to use the tracking error instead of the input bounds in order to modify the time-scaling of the reference path (Lévine, 2004; Szádeczky-Kardoss and Kiss, 2006). These methods change the traveling time of the reference path according to the actual tracking error by decelerating if the movement is not accurate enough and by accelerating if the errors are small or vanish.

Time-scaling is also introduced related to the notion of orbital flatness defined in (Fliess et al., 1999) where a Lie-Bäcklund equivalence of dynamical systems is established such that the transformation involved may change the time according to which the systems evolve. Another approach that relates time-scaling to feedback linearization is reported in (Sampei and Furuta, 1986).

The remaining part of the paper is organized as follows. The next section presents the kinematic models of the WMRs. Section 3 studies briefly the flatness properties of the models. Section 4 presents a simple motion planning method. The time-scaling concept is introduced in Section 5. Two tracking feedback laws, both using time-scaling are reported in Section 6. Simulation results are presented in Section 7 and a short conclusion terminates the paper.

2 KINEMATIC WMR MODELS

Let us introduce some notations first. Figure 1 depicts a WMR in the xy horizontal plane. Let us suppose that the Ackermann steering assumptions hold true, hence all wheels turn around the same point (denoted by P) which lies on the line of the rear axle. It follows that the kinematics of the robot can be fully described by the kinematics of a bicycle fitted to the longitudinal symmetry axis of the vehicle (see Figure 1). The coordinates of the rear axle midpoint are given by x and y. The orientation of the car with respect to the x axis is denoted by the angle θ, hence the WMR evolves on the configuration manifold $\mathbb{R}^2 \times \mathbb{S} = SE(2)$. The angle of the front wheel of the bicycle with respect to the longitudinal symmetry axis of the robot is denoted by φ. We consider $u_2 = \varphi$ as an input. The longitudinal velocity of the rear axle midpoint is denoted by u_1 if it is a control input (two input case) and by v_{car} if not (one input case).

The distance l between the front and rear axles equals to one. Then the model equations can be obtained after some elementary considerations which result (see also for example (Rouchon et al., 1993; Cuesta and Ollero, 2005))

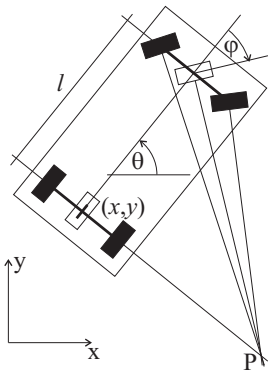


Figure 1: Notations of the kinematic model.

$$\dot{x} = u_1 \cos\theta \quad (1)$$
$$\dot{y} = u_1 \sin\theta \quad (2)$$
$$\dot{\theta} = u_1 \tan u_2. \quad (3)$$

Since time-scaling is involved in the sequel, we precise that the time in this system is denoted by t and $\dot{x}$ denotes the time derivative of the function $x(t)$ such that $\dot{t} = 1$.

Consider now the case where the longitudinal velocity is not a control input but an *external signal* v_{car} which is generated by the driver or by any other mean such that the controller has no influence on its evolution. The corresponding model with one input is defined by the equations

$$\dot{x} = v_{car} \cos\theta \quad (4)$$
$$\dot{y} = v_{car} \sin\theta \quad (5)$$
$$\dot{\theta} = v_{car} \tan u_2. \quad (6)$$

Consider now a time different from the time t, denoted by τ. Based on (1)-(3), let us define a model evolving with the time τ and given by the equations

$$x'_\tau = u_{\tau,1} \cos\theta_\tau \quad (7)$$
$$y'_\tau = u_{\tau,1} \sin\theta_\tau \quad (8)$$
$$\theta'_\tau = u_{\tau,1} \tan u_{\tau,2} \quad (9)$$

where the subscript τ denotes the dependence on the time τ and x'_τ is the derivative of the function $x_\tau(\tau)$ with respect to τ. (No subscript is used for variables dependent on time t except cases where the distinction is necessary). It is obvious that $\tau' = 1$ as $\dot{t} = 1$.

3 FLATNESS OF THE MODELS

One can easily verify or check in the literature (Rouchon et al., 1993; Fliess et al., 1995) that the

model (1)-(3) (respectively (7)-(9)) is differentially flat, hence it can be linearized by a dynamic feedback and a coordinate transformation. The flat output is given by x and y (respectively x_τ and y_τ).

The differential flatness property of the models can be exploited both for motion planning and tracking purposes. Given a reference trajectory $x_{ref}(t)$ and $y_{ref}(t)$ for the flat output variables x and y, which are at least two times differentiable with respect to the time t, one can determine the time functions of θ_{ref}, $u_{1,ref}$, and $u_{2,ref}$ which satisfy (1)-(3) according to a mapping

$$\{x_{ref},\ldots,\ddot{x}_{ref},y_{ref},\ldots,\ddot{y}_{ref}\} \rightarrow \{\theta_{ref},u_{1,ref},u_{2,ref}\}.$$

The same holds true for the model (7)-(9) evolving with the time τ hence there exists a mapping

$$\{x_{\tau,ref},\ldots,x''_{\tau,ref},y_{\tau,ref},\ldots,y''_{\tau,ref}\} \rightarrow \{\theta_{\tau,ref},u_{\tau,1,ref},u_{\tau,2,ref}\}. \quad (10)$$

The model (4)-(6) is not differentially, but orbitally flat for $v_{car} \equiv 1$ as shown by (Guay, 1999).

4 MOTION PLANNING

The motion planning is done for the system (7)-(9) exploiting its differential flatness property. The motion planning realizes the mappings

$$\tau \rightarrow \{x_{\tau,ref},x'_{\tau,ref},x''_{\tau,ref}\} \quad (11)$$
$$\tau \rightarrow \{y_{\tau,ref},y'_{\tau,ref},y''_{\tau,ref}\} \quad (12)$$

for $\tau \in [0,T]$ where T is the desired traveling time along the path such that the mapping (10) allows then to calculate the time functions of the remaining variables of the model.

Several motion planning schemes can be used to realize (11) and (12). One may want to solve an obstacle avoidance problem in parallel with the generation of the references (Cuesta and Ollero, 2005). For the sake of simplicity, seventh degree polynomial trajectories are considered in this paper such that

$$x_{\tau,ref} = \sum_{i=0}^{7} a_{x,i}\tau^i, \qquad y_{\tau,ref} = \sum_{i=0}^{7} a_{y,i}\tau^i. \quad (13)$$

The coefficients are obtained as solutions of a set of linear algebraic equations determined by the constraints that the polynomials and their three successive derivatives must satisfy at $\tau = 0$ and $\tau = T$. Notice that the non-zero constraints are no longer respected in the scaled time t for the derivatives of the references unless $\dot{\tau} \equiv 1$. It follows in particular that the constraints imposed on the longitudinal velocities at $\tau = 0$ (respectively at $\tau = T$) will be scaled by $\dot{\tau}(0)$ (respectively by $\dot{\tau}(t(T))$.

The motion planning can be done off-line prior to the tracking and the time-scaling does not need the redesign of the reference. It follows that more involved methods can be also applied including the one involving continuous curvature pathes with Fresnel integrals (Fraichard and Scheuer, 2004).

5 TIME-SCALING

A time-scaling law, which is defined by the mapping $t \mapsto \tau(t)$ or by its inverse $\tau \mapsto t(\tau)$ can be obtained based on the following consideration. Rewrite (4)-(6) as

$$dx = v_{car}dt\cos\theta \quad (14)$$
$$dy = v_{car}dt\sin\theta \quad (15)$$
$$d\theta = v_{car}dt\tan u_2. \quad (16)$$

Similarly, rewrite also (7)-(9) as

$$dx_\tau = u_{\tau,1}d\tau\cos\theta_\tau \quad (17)$$
$$dy_\tau = u_{\tau,1}d\tau\sin\theta_\tau \quad (18)$$
$$d\theta_\tau = u_{\tau,1}d\tau\tan u_{\tau,2}. \quad (19)$$

Consider now the model equations and the following relations obtained from (14)-(16) and (17)-(19) for the inputs of the models

$$\frac{dt}{d\tau} = t' = \frac{u_{\tau,1}}{v_{car}} \qquad u_2 = u_{\tau,2}. \quad (20)$$

This allows to determine a unique trajectory of the system (4)-(6) for a trajectory of the system (7)-(9) if the time function $v_{car}(t)$ and the initial conditions are given, and one supposes non-vanishing velocity functions v_{car} and $u_{\tau,1}$. As far as the initial conditions are considered one may, for instance, suppose that $x(0) = x_\tau(0)$, $y(0) = y_\tau(0)$, $\theta(0) = \theta_\tau(0)$. The relations in the other direction are similar and read

$$\frac{d\tau}{dt} = \dot{\tau} = \frac{v_{car}}{u_{\tau,1}} \qquad u_{\tau,2} = u_2. \quad (21)$$

The following proposition summarizes the properties of the time-scaling for trajectories with strictly positive (respectively negative) velocities.

Proposition 1 *Suppose that one considers trajectories of the different models of the kinematic car such that the velocities v_{car} and $u_{\tau,1}$ are both strictly positive (respectively strictly negative). Then the time-scaling $t \mapsto \tau(t)$ and $\tau \mapsto t(\tau)$ defined by* (20)-(21) *satisfying $t(0) = \tau(0)$ are such that the functions $\tau(t)$ and $t(\tau)$ are strictly increasing functions of their arguments (the scaled time never rewinds).*

This property is a general requirement for meaningful time-scaling and it is satisfied both for forward and backward motions of the car. The time-scaling has a singularity if the car is in idle position.

Note that for a fixed velocity time function $v_{car}(t)$, the time-scaling can be influenced by $u_{\tau,1}$, one of the inputs of the flat model evolving with τ. Observe moreover that for fixed v_{car} these relations do not define a one-to-one correspondence between the sets of trajectories of the respective systems and the number of inputs is not preserved, hence they are not Lie-Bäcklund isomorphisms (Fliess et al., 1999).

Suppose that the references are obtained for (7)-(9) and one disposes of the time functions $x_{\tau,ref}$, $x'_{\tau,ref}$, $x''_{\tau,ref}$, $y_{\tau,ref}$, $y'_{\tau,ref}$, $y''_{\tau,ref}$. (A simple method is given in preceding section for the planning of the reference motion.) The time-scaling is defined by the mappings

$$\{x_{\tau,ref},\ldots,x''_{\tau,ref}\} \rightarrow \{x_{ref}(t),\ldots,\ddot{x}_{ref}(t)\} \quad (22)$$

$$\{y_{\tau,ref},\ldots,y''_{\tau,ref}\} \rightarrow \{y_{ref}(t),\ldots,\ddot{y}_{ref}(t)\} \quad (23)$$

which can be determined using (21) since

$$\tau(t) = \int_0^t \frac{v_{car}}{u_{\vartheta,1}} d\vartheta \qquad \tau(0) = t(0) = 0 \quad (24)$$

$$v_{car} = \dot{\tau} u_{t,1} \quad (25)$$

$$\dot{v}_{car} = \ddot{\tau} u_{t,1} + \dot{\tau}\dot{u}_{t,1} \quad (26)$$

allow to express $\tau(t)$, $\dot{\tau}$, and $\ddot{\tau}$. Then

$$x_{ref}(t) = x_{\tau,ref}(\tau(t)) \quad (27)$$

$$\dot{x}_{ref}(t) = x'_{\tau,ref}(\tau(t))\dot{\tau} \quad (28)$$

$$\ddot{x}_{ref}(t) = x''_{\tau,ref}(\tau(t))\dot{\tau}^2 + x'_{\tau,ref}(\tau(t))\ddot{\tau} \quad (29)$$

and one obtains similar expressions for the higher order time derivatives and for the mapping (23).

Suppose that a reference trajectory is calculated according to the time τ and that one is looking for an open loop control of the real car to follow the geometry of the reference trajectory. Assume moreover that the reference trajectory is calculated based on the initial conditions of the real car. Then the open loop control signal $u_2(t)$ can be calculated from the reference $u_{\tau,2,ref}$ using the time-scaling defined by (20). Notice however that the real traveling time for a reference trajectory according the time t will be obtained as $t(T)$. If the driver generating v_{car} accelerates with respect to the reference trajectory than $T > t(T)$. If he/she is more careful than the algorithm providing the value for T than $T < t(T)$.

6 TRACKING FEEDBACK DESIGN

Two tracking controllers are presented in this section. The first one is based on the flatness property of the model with two inputs and requires the measurement of the car velocity and its two successive time derivatives. Measurement of the acceleration and its time derivative may be prohibitive for real applications. Therefore another tracking feedback is also suggested which is designed for a system obtained by linearizing the tracking error dynamics around the reference trajectory achieving only local stability of the reference trajectory.

6.1 Flatness-Based Tracking using Time-Scaling

The system (1)-(3) can be linearized by dynamic feedback in virtue of its differential flatness property. The resulting linear system is two chains of integrators

$$x^{(3)} = \omega_x \qquad y^{(3)} = \omega_y. \quad (30)$$

Suppose that one specifies the tracking behavior in terms of the tracking errors $e_x = x - x_{ref}$ and $e_y = y - y_{ref}$ such that the differential equations

$$e_x^{(3)} + k_{x,2}\ddot{e}_x + k_{x,1}\dot{e}_x + k_{x,0}e_x = 0 \quad (31)$$

$$e_y^{(3)} + k_{y,2}\ddot{e}_y + k_{y,1}\dot{e}_y + k_{y,0}e_y = 0 \quad (32)$$

hold true. The coefficients $k_{a,i}$ ($a \in \{x,y\}$, $i = 0,1,2$) are design parameters and have to be chosen such that the corresponding characteristic polynomials have all their roots in the left half of the complex plane. These linear differential equations define another (tracking feedback) for (30)

$$\omega_x = x_{ref}^{(3)} - k_{x,2}\ddot{e}_x - k_{x,1}\dot{e}_x - k_{x,0}e_x \quad (33)$$

$$\omega_y = y_{ref}^{(3)} - k_{y,2}\ddot{e}_y - k_{y,1}\dot{e}_y - k_{y,0}e_y. \quad (34)$$

Consider now the model described by (4)-(6). This single input model is not differentially flat, hence cannot be linearized by feedback. It follows that the flatness property cannot be (directly) used to solve the tracking problem.

Let us study the possibility to use the time-scaling defined above to achieve the desired tracking behavior for the non-differentially flat model (4)-(6) with one input.

The idea is to use the differentially flat model (7)-(9) to solve the motion planning problem with the time τ. Then one would use a tracking feedback controller designed again for the flat model which produces $u_{\tau,1}$ and $u_{\tau,2}$. The signal $u_{\tau,1}$ produced by the controller is used to drive the time-scaling of the reference trajectory designed for the time τ according to (21). The control loop is illustrated in Figure 2 for the model (4)-(6) where the controller provides $u_2 = \varphi$ to the single input model. The tracking feedback is

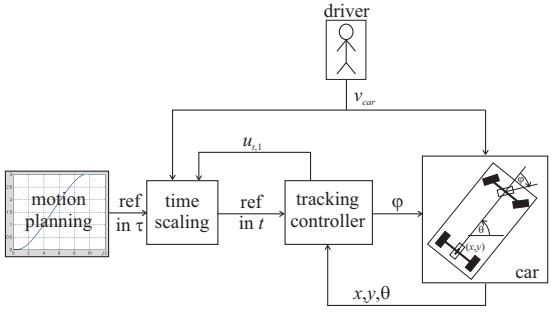


Figure 2: Tracking controller with time-scaling.

designed using the flatness property of the model with two inputs. Define first the dynamics for the feedback as

$$\dot{\zeta}_1 = \zeta_2 \qquad \dot{\zeta}_3 = v_2 \tag{35}$$

$$\dot{\zeta}_2 = v_1 \qquad \varphi = \zeta_3 \tag{36}$$

$$u_1 = \zeta_1 \tag{37}$$

where ζ_1, ζ_2, and ζ_3 are the (inner) states of the feedback. Observe that ζ_2 and v_1 give precisely the derivatives of u_1 which need to realize the time-scaling in (24)-(26), hence no numerical differentiation is needed.

The inputs v_1 and v_2 of the feedback dynamics must be determined such that the tracking errors $e_x(t)$ and $e_y(t)$ satisfy (31) and (32), respectively.

For, one needs to determine first x, $\dot{x}$, $\ddot{x}$, $x^{(3)}$, y, $\dot{y}$, $\ddot{y}$, and $y^{(3)}$ as functions of x, y, θ, ζ_1, ζ_2, and ζ_3 which are the states of the closed loop system including the measured states of the kinematic car model, and the states of the feedback (35)-(37). After some cumbersome but elementary differentiations one obtains

$$\dot{x} = \zeta_1 \cos\theta \tag{38}$$

$$\ddot{x} = \zeta_2 \cos\theta - \zeta_1^2 \sin\theta \tan\zeta_3 \tag{39}$$

$$x^{(3)} = \frac{v_1 \cos\theta \cos^2\zeta_3 - 3\zeta_1\zeta_2 \sin\theta \sin\zeta_3 \cos\zeta_3}{\cos^2\zeta_3} - \frac{\zeta_1^3 \cos\theta - \zeta_1^3 \cos\theta \cos^2\zeta_3 + v_2\zeta_1^2 \sin\theta}{\cos^2\zeta_3} \tag{40}$$

$$\dot{y} = \zeta_1 \sin\theta \tag{41}$$

$$\ddot{y} = \zeta_2 \sin\theta + \zeta_1^2 \cos\theta \tan\zeta_3 \tag{42}$$

$$y^{(3)} = \frac{v_1 \sin\theta \cos^2\zeta_3 + 3\zeta_1\zeta_2 \cos\theta \sin\zeta_3 \cos\zeta_3}{\cos^2\zeta_3} - \frac{\zeta_1^3 \sin\theta - \zeta_1^3 \sin\theta \cos^2\zeta_3 - v_2\zeta_1^2 \cos\theta}{\cos^2\zeta_3}. \tag{43}$$

These expressions allow to calculate e_x, $\dot{e}_x$, $\ddot{e}_x$, e_y, $\dot{e}_y$, and $\ddot{e}_y$ using the reference trajectory (scaled with t) and the states of the closed loop system. Plugging in these expressions into (31) and (32), and using (30) one gets

$$\begin{bmatrix} \cos\theta & -\frac{\zeta_1^2 \sin\theta}{\cos^2\zeta_3} \\ \sin\theta & \frac{\zeta_1^2 \cos\theta}{\cos^2\zeta_3} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \omega_x - A \\ \omega_y - B \end{bmatrix} \tag{44}$$

with

$$\begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} \frac{-3\zeta_1\zeta_2 \sin\theta \sin\zeta_3 \cos\zeta_3 - \zeta_1^3 \cos\theta + \zeta_1^3 \cos\theta \cos^2\zeta_3}{\cos^2\zeta_3} \\ \frac{3\zeta_1\zeta_2 \cos\theta \sin\zeta_3 \cos\zeta_3 - \zeta_1^3 \sin\theta + \zeta_1^3 \sin\theta \cos^2\zeta_3}{\cos^2\zeta_3} \end{bmatrix} \tag{45}$$

where the inverse of the coefficient matrix can be calculated symbolically. One obtains

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ \frac{-\sin\theta \cos^2\zeta_3}{\zeta_1^2} & \frac{\cos\theta \cos^2\zeta_3}{\zeta_1^2} \end{bmatrix} \begin{bmatrix} \omega_x - A \\ \omega_y - B \end{bmatrix}. \tag{46}$$

The tracking feedback law is defined by (33)-(34), (35)-(37), and by (46). A singularity occurs if $\zeta_1^2 = u_1^2 = 0$ which corresponds to zero longitudinal velocity. Another singular situation corresponds to $\zeta_3 = \varphi = u_2 = \pm\pi/2$ which may occur if the steered wheels are perpendicular to the longitudinal axis of the car. Singularities imply the loss of controllability of the kinematic car model.

6.2 Linearized Error Dynamics

The above method needed the time derivatives of the velocity to carry out the time-scaling which may be difficult to measure or estimate in real application. The method presented in this section uses a transformation of the tracking error expressed in the configuration variables, and the non-linear model obtained is linearized around the reference trajectory. The linearized model is controlled by a state feedback similar to the one reported in (Dixon et al., 2001). The lost input, which is the longitudinal velocity of the WMR is again replaced by a virtual input which depends on the time-scaling of the reference trajectory.

A slightly different kinematic model is used for this method such that the longitudinal velocity of the rear axle midpoint and the tangent of the steering angle ($u_3 = \tan\varphi$) are the inputs of the mobile robot. If the one input case is considered, $u_3 = \tan\varphi$ is the single control input.

Suppose, that the desired behavior of the robot is given by the time functions $x_{\tau,ref}(\tau)$, $y_{\tau,ref}(\tau)$, $\theta_{\tau,ref}(\tau)$ such that these functions identically satisfy (7)-(9) for the corresponding reference input signals $u_{\tau,1,ref}$ and $u_{\tau,3,ref} = \tan u_{\tau,2,ref}$.

We suggest to scale this reference trajectory according to the time t. The scaled reference trajectory is given by $x_{ref}(t) = x_{\tau,ref}(\tau)$, $y_{ref}(t) = y_{\tau,ref}(\tau)$, and

$\theta_{ref}(t) = \theta_{\tau,ref}(\tau)$ and similarly to (7)-(9)

$$x'_{\tau,ref} = u_{\tau,1,ref}\cos\theta_{\tau,ref} \quad (47)$$
$$y'_{\tau,ref} = u_{\tau,1,ref}\sin\theta_{\tau,ref} \quad (48)$$
$$\theta'_{\tau,ref} = u_{\tau,1,ref}u_{\tau,3,ref}. \quad (49)$$

The tracking errors are defined for the configuration variables as $e_x = x - x_{ref}$, $e_y = y - y_{ref}$, and $e_\theta = \theta - \theta_{ref}$. Let us now consider the transformation

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} \quad (50)$$

of the error vector (e_x, e_y, e_θ) to a frame fixed to the car such that the longitudinal axis of the car coincides the transformed x axis. Differentiating this equation w.r.t. time t and using the general rule $\dot{a}(\tau) = \frac{da(\tau)}{dt} = \frac{\partial a}{\partial \tau}\frac{\partial \tau}{\partial t} = a'\dot{\tau}$ we get the differential equation

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} = \begin{bmatrix} 0 & v_{car}u_3 & 0 \\ -v_{car}u_3 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} + \begin{bmatrix} 0 \\ \sin e_3 \\ 0 \end{bmatrix} u_{\tau,1,ref}\dot{\tau} + \begin{bmatrix} w_1 & 0 \\ 0 & 0 \\ 0 & w_2 \end{bmatrix} \quad (51)$$

which describes the evolution of the errors with respect to the reference path. The inputs w_1 and w_2 are

$$w_1 = v_{car} - \dot{\tau}u_{\tau,1,ref}\cos e_3 \quad (52)$$
$$w_2 = v_{car}u_3 - \dot{\tau}u_{\tau,1,ref}u_{\tau,3,ref}. \quad (53)$$

Notice that from these inputs w_1 and w_2 the first derivative of the time scaling ($\dot{\tau}$) and the real input $u_3 = \tan\varphi$ can be calculated as

$$\dot{\tau} = \frac{v_{car} - w_1}{u_{\tau,1,ref}\cos e_3} \quad (54)$$
$$u_3 = \frac{w_2 + \dot{\tau}u_{\tau,1,ref}u_{\tau,3,ref}}{v_{car}} \quad (55)$$

if the reference value for the longitudinal velocity $u_{\tau,1,ref} \neq 0$, the error of the orientation $e_3 \neq \pm\pi/2$, and the longitudinal velocity of the car $v_{car} \neq 0$.

This system can be linearized along the reference trajectory, i.e. for $[\ e_1 \quad e_2 \quad e_3\ ]^T = 0$. The linearized system is controllable if at least one of the reference control inputs $(u_{\tau,1,ref}, u_{\tau,3,ref})$ is nonzero. The setpoint of the linearized system obtained from (51) can be locally stabilized by a state feedback of the form

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = -K \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \quad (56)$$

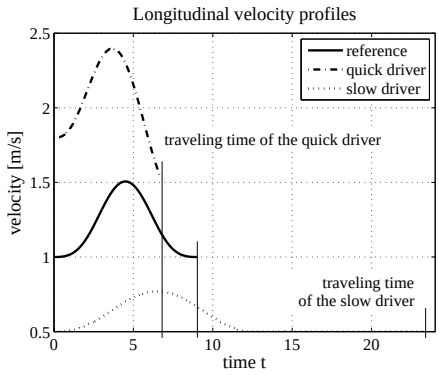


Figure 3: Velocity profiles for the reference, for the quick driver, and for the slow driver. The traveling times are obtained in closed loop.

such that the gain matrix K puts the eigenvalues of the closed loop system in the left half of the complex plane.

The way of calculations is as follows. One suppose that the tracking errors of the configuration variables (e_x, e_y, e_θ) are measured, hence the the error (e_1, e_2, e_3) can be determined using (50). Then the state feedback (56) allows to calculate w_1 and w_2. From the actual value of w_1 one can determine $\dot{\tau}$ using the current value of v_{car}, e_3, and the value of $u_{\tau,1,ref}$ according to the time t obtained by scaling the reference. The input $u_3 = \tan\varphi$ is calculated according to (55) using w_2. The function $\tau(t)$ is obtained by the on-line integration of $\dot{\tau}$ determined by (54) using the initial condition $\tau(0) = 0$. The time distribution of the reference trajectory is finally modified according to τ and $\dot{\tau}$.

Since a linearized model was used for the controller design only local stability is guaranteed. (E.g. if $w_1 \approx 0$ is not fulfilled, $\dot{\tau}$ in (54) can get a negative value, which is not allowed since time cannot rewind.)

7 SIMULATIONS

Examples are shown to demonstrate the functioning of both time-scaling based tracking controllers for the one input case.

7.1 Results of Flatness-Based Solution

We use the feedback described in Subsection 6.1, such that $u_{\tau,1}$ generated by the feedback law drives the time-scaling given in Section 5 together with the measured v_{car} and its two successive time derivatives. The reference trajectory starts from the point $(x_{\tau,ref}(0) = 0, y_{\tau,ref}(0) = 0, \theta_{\tau,ref}(0) = 0)$ and arrives to the point $(x_{\tau,ref}(T) = 10, y_{\tau,ref}(T) = 3.5, \theta_{\tau,ref}(T) = 0)$, all

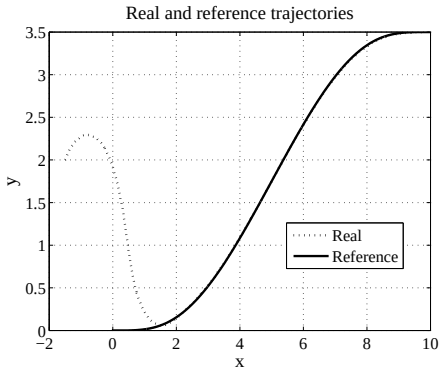


Figure 4: Real and reference trajectories in the horizontal plane – slow driver.

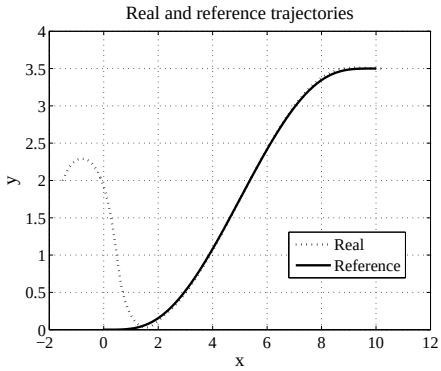


Figure 5: Real and reference trajectories in the horizontal plane – quick driver.

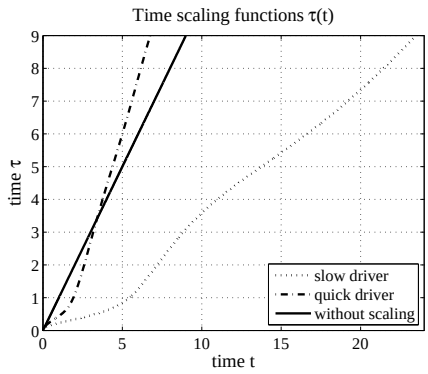


Figure 6: The time-scaling functions $\tau(t)$ along the path for the slow and quick drivers.

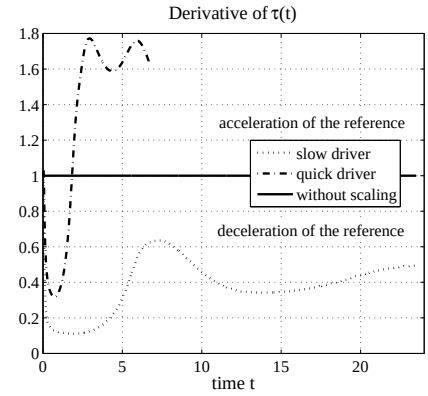


Figure 7: The derivative of the time-scaling functions $\dot{\tau}$.

distances are given in meters and the orientation is given in radians. The traveling time of the reference trajectory is $T = 9$ seconds.

The real initial configuration of the WMR differs from the one used for motion planning, since $x(0) = -1.5$, $y(0) = 2$, and $\theta(0) = \pi/4$.

Two cases are presented such that the geometry of the reference trajectory and the reference velocity profile obtained are the same. The driver's behavior is different for the two cases. In the first case, referred to as the *slow driver* case, the driver imposes considerably slower velocities than those obtained by the motion planning. In the second case, referred to as the *quick driver* case, the driver generates higher velocities than the reference velocity profile. All velocity profiles are given in Figure 3.

The geometries of the reference trajectories and the real trajectories in the horizontal plane are depicted in Figure 4 (slow driver) and in Figure 5 (quick driver). Exponential tracking of the reference trajectory is achieved for each scenario with similar geometry of the real path. Figure 6 and Figure 7 show the effects of on-line time-scaling. If the car is driven by a slow driver it needed more than 23 seconds according to time t to achieve the traveling time $T = 9$ sec which is given for the reference trajectory according to the time τ. The reference in τ was decelerated all along the trajectory ($\dot{\tau} < 1$). The deceleration is also accentuated at low values of t which corresponds to large tracking errors. The time-scaling is completely different for the quick driver who reaches the end of the trajectory faster according to the time t than according to the time τ which means that the reference was accelerated except a short section at the beginning where the tracking error elimination slows down the time-scaling despite the driver's efforts.

7.2 Results Obtained by State Feedback

Here we use the feedback law described in the subsection 6.2 such that the same reference trajectory and initial configuration were used as in the previous subsection.

The reference trajectory and the real path are shown in Figure 8 for the velocity profiles depicted in Figure 9. We achieved exponential tracking.

If the difference between the real and reference initial configurations is larger, the linearized model is

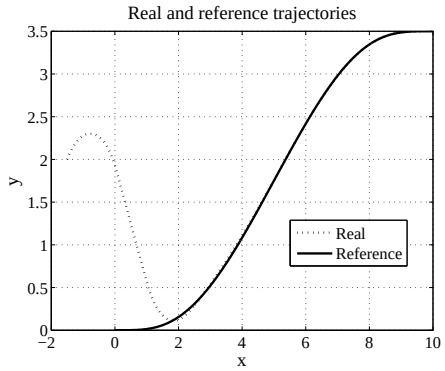


Figure 8: Real and reference trajectories in the horizontal plane – simulation 1.

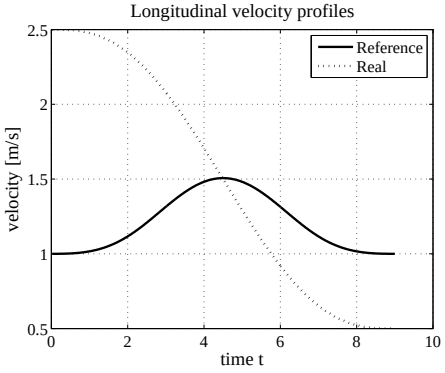


Figure 9: Velocity profiles for the reference and for the car in simulation 1.

no longer valid and the time-scaling may rewind.

8 CONCLUSION

The paper presented two time-scaling based tracking control methods for WMRs with one input such that the longitudinal velocity of the vehicle is generated externally and cannot be considered as a control input. The time-scaling involves the car velocity and its derivatives which need to be measured or estimated. For the tracking controller designed for the linearized error dynamics the time derivatives of the velocity are not needed. The exponential decay of the initial error along the trajectory can be ensured. The results can be extended for the *n*-trailer case.

ACKNOWLEDGEMENTS

The research was partially supported by the Hungarian Science Research Fund under grant OTKA T 068686 and by the Advanced Vehicles and Vehicle Control Knowledge Center under grant RET 04/2004.

REFERENCES

Benalia, A., Djemai, M., and Barbot, J.-P. (2003). Control of the kinematic car using trajectory generation and the high order sliding mode control. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, volume 3, pages 2455–2460.

Cuesta, F. and Ollero, A. (2005). *Intelligent Mobile Robot Navigation*, volume 16 of *Springer Tracts in Advanced Robotics*. Springer, Heidelberg.

Dahl, O. and Nielsen, L. (1990). Torque-Limited Path Following by On-Line Trajectory Time Scaling. *IEEE Trans. Robot. Automat.*, 6(5):554–561.

Dixon, W. E., Dawson, D. M., Zergeroglu, E., and Behal, A. (2001). Nonlinear Control of Wheeled Mobile Robots. In *Lecture Notes in Control and Information Sciences*. Springer.

Fliess, M., Lévine, J., Martin, P., and Rouchon, P. (1995). Flatness and Defect of Nonlinear Systems: Introductory Theory and Examples. *Int. J. of Control*, 61(6):1327–1361.

Fliess, M., Lévine, J., Martin, P., and Rouchon, P. (1999). A Lie-Bäcklund Approach to Equivalence and Flatness of Nonlinear Systems. *IEEE Trans. Automat. Contr.*, 44(5):922–937.

Fraichard, T. and Scheuer, A. (2004). From Reeds and Shepps to Continuous-Curvature paths. *IEEE Transaction on Robotics and Automation*, 20.

Gu, D. and Hu, H. (2002). Neural Predictive Control for a Car-like Mobile Robot. *Robotics and Autonomous Systems*, 39:73–86.

Guay, M. (1999). An Algorithm for Orbital Feedback Llinearization of Single-Input Control Affine Systems. *Systems and Control Letters*, 38:271–281.

Hollerbach, J. M. (1984). Dynamic Scaling of Manipulator Trajectories. *Trans. of the ASME, J. of Dynamic Systems, Measurement, and Control*, 106(1):102–106.

Lévine, J. (2004). On the Synchronization of a Pair of Independent Windshield Wipers. *IEEE Trans. Contr. Syst. Technol.*, 12(5):787–795.

Respondek, W. (1998). Orbital Feedback Linerization of Single-Input Nonlinear Control Systems. In *Proceedings of the IFAC NOLCOS'98*, pages 499–504, Enschede, The Netherlands.

Rouchon, P., Fliess, M., Lévine, J., and Martin, P. (1993). Flatness and Motion Planning: The Car with *n*-Trailers. In *ECC'93, Proceedings of the European Control Conference*, pages 1518–1522.

Sampei, M. and Furuta, K. (1986). On Time Scaling for Nonlinear Systems: Application to Linearization. *IEEE Transactions on Automatic Control*, AC-31:459–462.

Szádeczky-Kardoss, E. and Kiss, B. (2006). Tracking Error Based On-Line Trajectory Time Scaling. In *INES 2006, Proc. of 10th Int. Conf. on Intelligent Engineering Systems*, pages 80–85.

CLEAR IMAGE CAPTURE
Active Cameras System for Tracking a High-speed Moving Object

Hiroshi Oike, Haiyuan Wu, Chunsheng Hua and Toshikazu Wada
Department of Computer and Communication Sciences, Wakayama University, 930 Sakaedani, Wakayama City 640-8510, Japan
{*oike, wuhy, hua,twada*}*@vrl.sys.wakayama-u.ac.jp*

Keywords: Object tracking, Binocular Active camera, Clear image.

Abstract: In this paper, we propose a high-performance object tracking system for obtaining high-quality images of a high-speed moving object at video rate by controlling a pair of active cameras that consists of two cameras with zoom lens mounted on two pan-tilt units. In this paper, "high-quality image" implies that the object image is in focus and not blurred, the size of the object in the image remains unchanged, and the object is located at the image center. To achieve our goal, we use the K-means tracker algorithm for tracking objects in an image sequence captured by the active cameras. We use the results of the K-means tracker to control the angular position and speed of each pan-tilt-zoom unit by employing the PID control scheme. By using two cameras, the binocular stereo vision algorithm can be used to obtain the 3D position and velocity of the object. These results are used in order to adjust the focus and zoom. Moreover, our system allows the two cameras to gaze at a single point in 3D space. However, this system may become unstable when the time response deteriorates by excessively interfering in a mutual control loop or by strict restriction of the camera action. In order to solve these problems, we introduce the concept of reliability into the K-means tracker, and propose a method for controlling the active cameras by using relative reliability. We have developed a prototype system and confirmed through extensive experiments that we can obtain focused and motion-blur-free images of a high-speed moving object at video rate.

1 INTRODUCTION

It is likely that a captured image may include a blurred object if the object is moving at a high speed and the camera is relatively stable. In such cases, we will lose important information (e.g., the object's edge and colors), which is required in several computer vision researches. With regard to the problems in recognizing and understanding high-speed moving objects, capturing a high-quality image is as important as analyzing the captured object image.

To solve these problems, we propose a high-performance object tracking system for obtaining high-quality images of a high-speed moving object at video rate.

1.1 Related Work

Related works on active vision tracking systems have been developed by many researchers. The following are examples of some representative researches.

(a) Active tracking system using vision chip (Komuro et al., 2003).

(b) Active tracking system using high-speed cameras (Okada et al., 2004).

(c) Object detection and tracking system using fixed viewpoint pan-tilt-zoom camera (Matsuyama et al., 2000).

(d) Active tracking system using binocular stereo heads (Bjorkman and Eklundh, 2002).

(e) Binocular pursuit system (Coombs and Brown, 1993).

(f) Visual closed-loop system using Dynamic effect (Corke and Good, 1996).

The system in (a) may be the fastest driven tracking system in the world. Since it uses a special sensor (Vision Chip), the resolution of the obtained image is quite low. Further, the system can only pursue an object in an illumination-controlled indoor room.

The system in (b) was constructed using a high-speed camera, and can thus only obtain very dark images due to the short exposure time.

The system in (c) uses an *fixed viewpoint pan-tilt-zoom camera* (hereafter referred to as an FV-PTZ

camera) similar to our system. Due to the use of background subtraction as the object tracking algorithm, the active camera must stop its motion for image capture. Therefore, moving target images captured by this system may appear blurred or out of focus. Moreover, the tracking performance of this system is a few dozen degrees per second.

System (d) was constructed using binocular cameras similar to our system. The advantage of this system is that it seldom fails in tracking a target that is at a different depth when compared to its surrounding objects. This is a fast-driven system; however, it is very expensive because it uses very complex special hardware.

In (e), the system was constructed using binocular camera head fixed on the robot arm, and In (f), it presented the effect of the introduction the dynamic control (like feed-forward control) into closed-loop tracking system. In these manuscripts, the methods of object tracking are not described clearly.

As is evident from the details stated above, the related works require complex, expensive, and special hardware; in addition, their operation is seldom stable in a real environment. Moreover, their systems obtain target object images regardless of their quality, because the architecture of these systems is based on the concept of only tracking the object.

1.2 Our Approach

In our system, we use two computers and video cameras that are available in the market to obtain the object image, track the target object, and control the active cameras at 30 fps. In our system, the camera is controlled such that it moves at the same angular speed and direction as the target object. Therefore, we can obtain an image with a blurred background and clear target at the image center (Fig.1). Additionally, we use binocular active cameras to track the object. We can then estimate the 3D position and velocity of the object. The 3D information can be used for adjusting the focus and zoom of the camera. Therefore, we can obtain target images that are clearer than those

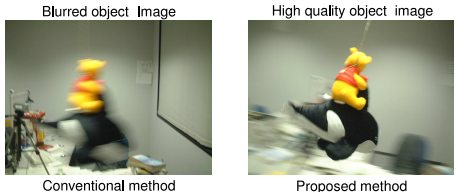


Figure 1: Obtained by conventional method (left) and proposed method (right).

obtained by using only a single active camera. The requirements for our system are as follows:

(1) binocular active cameras to focus their optic axis on a point in 3D space

(2) target to appear at the center of the images

This is because condition (1) helps us to avoid the contradiction between detection with two cameras. Epipolar geometry can work stably only if there is no contradiction between the two cameras.

Condition (2) is required because the view angle will become narrow when zooming in and the target will easily escape from the image. In such cases, the object tracking may completely fail, and thus, there will be no means to control the cameras.

In fact, it is difficult to estimate the *absolute correct epipolar line* because of the errors in object tracking or estimation of camera directions. The system becomes unstable if it is controlled according to an incorrect epipolar line. Further, the system may become unstable and lose its smoothness if the time response deteriorates under the influence of excessive interactions between the two control loops, or if the actions of the cameras are overly restricted. If the tracking system's action is not smooth, the target in the image will appear blurred and the accuracy of the object tracking will deteriorate. Therefore, the control of the active cameras will become increasingly unstable. As previously described, to focus the optic axis of two cameras on one 3D point, we must solve the following problems.

(A) The information sent from the other camera may be incorrect or of low accuracy if a tracking failure occurs.

(B) Excessive constraint from the other camera will make the tracking system's action unstable.

The conventional related studies on active vision tracking systems do not mention the methods for controlling the binocular active camera with an emphasis on the quality of the captured images by solving problems (A) and (B).

Therefore, in this paper, we have proposed a new method for solving these problems and constructing a high-speed-tracking active camera system, that can continuously obtain high quality images. Our system can automatically control the direction, zoom, and focus of the two cameras to focus on a point in 3D space.

To solve problem (A), we introduce the concept of reliability into the K-means tracker and propose a method to constrain the camera action by using this reliability, which is based on the calculation of the distance from the K-means clusters.

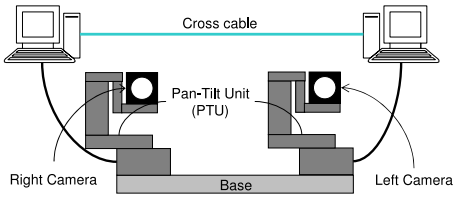


Figure 2: Construction of high-speed-tracking active cameras.

Figure 3: FV-PTZ camera(Active camera).

To solve problem (B), we propose a method for determining the level of constraint with a relative reliability.

By using our proposed methods, the binocular active cameras can be smoothly controlled and their optic axes can intersect at a point in 3D space.

2 CONSTRUCTION OF THE PROPOSED SYSTEM

We construct our high-speed-tracking active cameras with two active cameras and two computers (Fig.2). The two computers can communicate with each other and share their observations.

For the active camera in our proposed system, we employ a FV-PTZ camera (Fig.3). This camera is calibrated such that its optical center corresponds with the point of intersection of the pan and tilt axes. Therefore, the optical center of the FV-PTZ camera does not move if the pan or tilt angle is changed (Fig.4).

Therefore, by using this camera, we can ignore the movement of the optical center. As a result, it is easy to estimate the angular velocity of the target from the captured image and represent the target velocity by the angular velocity of the optical center.

In our system, since two cameras are fixed on the base, as shown in Fig.2, the distance between the cameras is constant. Therefore, we can only consider the rotational relationship between the two cameras and ignore the translation of the optical center.

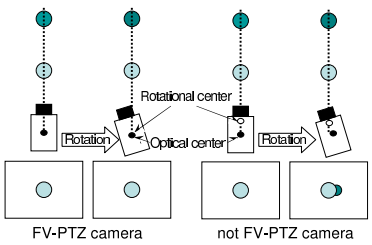


Figure 4: Particularity of FV-PTZ camera.

3 ANALYSIS OF THE OBJECT TRACKING ALGORITHM ON THE IMAGE

Numerous powerful algorithms for object tracking have been developed, such as Condensation(Isard and Blake, 1998), mean-shift(Comaniciu et al., 2003) and K-means tracker(C. Hua and Wada, 2006a; C. Hua and Wada, 2006b).

Condensation is a very robust algorithm that allows for ambiguity in the target position in the image. However, it cannot accurately estimate the target position because of this ambiguity.

For controlling the direction of the active cameras to track a moving object, the tracking algorithm must output a unique result but not an ambiguous one. Thus, we consider the mean-shift and K-means tracker algorithms to be suitable for our system.

Authors of mean-shift algorithm claim that it can adapt to variations in the target shape, color distribution, and size. However, through our experiments, we found that tracking becomes unstable if the target size varies greatly. Another problem is that when the target is monochromatic and in the plan shape, the mean-shift algorithm becomes sensitive to the color shift.

We have developed a K-means tracker algorithm that utilizes K-means clustering by using both the positive samples of the target and the negative samples surrounding the target. In this algorithm, the target feature is composed of the position as well as its color because the clustering is performed in a 5D space spanned by 3D color and 2D position parameters. This implies that this algorithm can adapt to not only target position but also the color shift of the target. Therefore, it can adapt to the shift in the illumination environment caused by changing the camera direction.

Another feature of the K-means tracker is that it can adapt to variations in the target shape and size in the image since it uses a variable ellipse model.

Thus, K-means Tracker is the most suitable algorithm for the active vision tracking system.

3.1 Summary of K-means Tracker

In the K-means tracker, robust object tracking is realized by using a variable ellipse model that is updated in each frame according to the clustering results. The pixels on the variable ellipse contour are defined as the representative non-target samples, and the area within the ellipse is the target search area.

In the first frame, we manually select some target cluster centers whose number is roughly the same as the number of colors contained by the target. In addition, we select one non-target cluster center $\mathbf{b}$ on the background for the tracking system. Then, the selected non-target center and the centroid $\mathbf{c}$ of the target centers will constitute the initial variable ellipse model in the form of a circle. The distance $\|\mathbf{c}-\mathbf{b}\|$ is the radius of this circle.

3.1.1 Clustering in the 5d Feature Space

To represent the properties of the target features in the K-means Tracker, each pixel in an image is described by a 5D feature vector $\mathbf{f} = [\mathbf{k}\ \mathbf{p}]^T$, where, $\mathbf{k} = [Y\ U\ V]^T$ describes the color similarity and $\mathbf{p} = [x\ y]^T$ describes the position approximation of the pixel. The feature vector of the i th target cluster center is represented as follows:

$$\mathbf{f}_{\mathrm{T}}(i) = [\mathbf{k}_{\mathrm{T}}(i)\ \mathbf{p}_{\mathrm{T}}(i)]^{\mathrm{T}} \quad (i = 1 \sim \mathrm{n}). \tag{1}$$

The feature tor of the j th non-target cluster center on the ellipse contour is represented as follows:

$$\mathbf{f}_{\mathrm{N}}(j) = [\mathbf{k}_{\mathrm{N}}(j)\ \mathbf{p}_{\mathrm{N}}(j)]^{T} \quad (j = 1 \sim \mathrm{m}). \tag{2}$$

Here, n and m describe the number of cluster center of target and non-target, respectively.

To distinguish whether pixel u belongs to the target center or not, we calculate the distances from $\mathbf{f}_u$ to the target and non-target cluster centers, respectively.

$$d_T(\mathbf{f}_u) = \min_{i=1\sim\mathrm{n}} \{\|\mathbf{f}_{\mathrm{T}}(i) - \mathbf{f}_u\|^2\} \tag{3}$$

$$d_N(\mathbf{f}_u) = \min_{j=1\sim\mathrm{m}} \{\|\mathbf{f}_{\mathrm{N}}(j) - \mathbf{f}_u\|^2\} \tag{4}$$

Here, within the search area, $\mathbf{f}_u$ describes the feature vector at pixel u, $d_T(u)$ and $d_N(u)$ describe the distances from $\mathbf{f}_u$ to its nearest target cluster center and nearest non-target cluster center, respectively.

If $d_T(\mathbf{f}_u) < d_N(\mathbf{f}_u)$, the pixel is detected as a target pixel; otherwise, it is a non-target pixel.

3.1.2 Updating Variable Ellipse Model

To estimate the search area represented as a variable ellipse, we represent the equation of ellipse parameters as a relation of the Mahalanobis distance and the Gaussian probability density function.

$$[\mathbf{y}-\mathbf{c}]^{\mathrm{T}}\overline{\Sigma}^{-1}[\mathbf{y}-\mathbf{c}] = J \tag{5}$$

Where,

$$J = -2\ln(1 - \frac{P}{100}) \tag{6}$$

$$\overline{\Sigma} = \frac{1}{e}\sum_{\mathbf{y}\in S}[\mathbf{y}-\mathbf{c}][\mathbf{y}-\mathbf{c}]^{\mathrm{T}}. \tag{7}$$

Equations (5), (6), and (7) indicate that the search area ellipse will contain P % of the target pixels existing within the ellipse when applying Gaussian probability function is applied to fit to the set of target points (Fig. 5).). This has the added effect of removing outlying pixels.

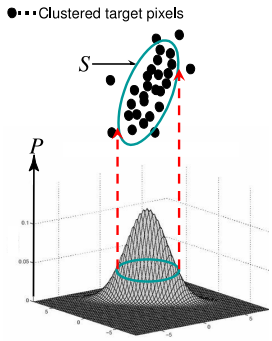


Figure 5: Estimation of the ellipse parameter by using Mahalanobis distance and Gaussian probability density function.

The center $\mathbf{c}$ of the variable ellipse at the next frame is calculated as follows:

$$\mathbf{c} = \frac{1}{e}\sum_{\mathbf{y}\in S}\mathbf{y}. \tag{8}$$

Here, S describes the pixel set inside the ellipse; e describes the number of target pixels inside the ellipse; and $\mathbf{y} = [y_x, y_y]^{\mathrm{T}}$ describes the target pixels.

In our system, we use $\mathbf{c}$ to represent the target center. By updating $\mathbf{c}$ in every frame, our system can track the target and estimate its angular velocity.

4 PROCESS FLOW

The K-means tracker discriminates each pixel within the search area into a target and non-target pixel. In our method, based on this discrimination, we propose the concept of reliability into the K-means tracker. The reliability represents how similarly each pixel belongs to each target cluster. With this reliability, our proposed system can determine which camera tracks the target more correctly and restrain the action of the camera with a lower reliability based on the output of the higher one.

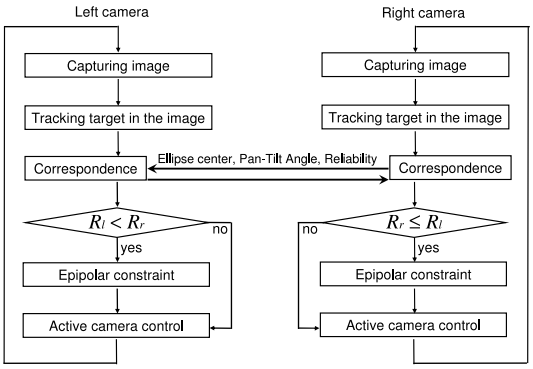


Figure 6: Flow chart of the our system.

In Fig.6, we show the flowchart of our proposed active camera system where the constraint on the camera action is based on the reliability. The reliability of the left and right cameras is described by R_l and R_r , respectively. The left and right active cameras can be independently controlled using the result of K-means tracker. In such cases, each camera can independently track the target, but it cannot automatically control the zoom and focus.

Our system tracks the target in the images captured from both cameras by using the K-means tracker. It then estimates the target position in the each image. With this position information, the direction from each optical center to the target can be calculated.

With the result of K-means tracker, we can calculate the reliability of each camera. We can then estimate the epipolar line on the lower reliability camera according to the target position in the image captured by the higher reliability camera. The ellipse center in the image taken from the lower reliability camera is constrained on the estimated epipolar line. Since the optical axis of the active camera is controlled based on the target position in the image, the optical axis of the active camera with the lower reliability is constrained on the estimated epipolar line. If the optical axis constrains the ellipse position excessively, the FV-PTZ unit response is lost; the velocity and direction of the active cameras will then be different from those of the tracked target. Due to these reasons, the tracking system may easily become unstable.

To solve this problem, we propose a method in which the ellipse position is not completely constrained on the epipolar line but is weighted by the higher reliability value of the higher reliability camera.

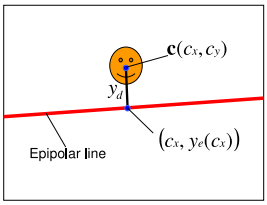


Figure 7: Ellipse position constraint on the epipolar line along the vertical direction.

4.1 Calculation of the Reliability

The reliability of each camera is calculated according to the result of the distance calculated in the feature space of the K-means tracker.

In this paper, u represents a pixel in the search area restricted in an ellipse. $r(u)$ represents the reliability of u and describes the similarity of target clusters. $r(u)$ is calculated as follows:

$$r(u) = \begin{cases} \frac{d_N(\mathbf{f}_u)}{d_N(\mathbf{f}_u)+d_T(\mathbf{f}_u)} & (d_T(\mathbf{f}_u) < d_N(\mathbf{f}_u)) \\ 0 & (otherwise). \end{cases} \quad (9)$$

Here, the distances of $d_T(\mathbf{f}_u)$ and $d_N(\mathbf{f}_u)$ are calculated by Eq.(3) and Eq.(4), respectively.

Further, R is the reliability of all pixels in the search area restricted in the same ellipse, and is calculated by

$$R = \frac{1}{e}\sum_{u=1}^{C} r(u). \quad (10)$$

Here, C is the number of pixels in the ellipse and e is the number of target pixels. To correct the difference between ellipse size of the two cameras, R is normalized by e.

4.2 Constraint Ellipse on Epipolar Line by using Relative Reliability

In our proposed system, based on the results of tracking with the high-reliability camera, the ellipse position is constrained on the epipolar line only in the low-reliability camera.

In many cases, the two epipolar lines drawn in each image become horizontal because the active cameras are fixed on the horizontal base. Therefore, the ellipse position is constrained on the epipolar line only along the vertical direction.

In Fig.7, y_d is the distance along the vertical direction from the ellipse center $\mathbf{c} = (c_x, c_y)$ to the epipolar line. $y_e(x)$ is the vertical coordinate of the intersection point of the epipolar line and the vertical line that passes through the ellipse center $\mathbf{c}$. y_d is

calculated as follows:

$$y_d = y_e(c_x) - c_y. \tag{11}$$

If the ellipse position is constrained on the epipolar line, only with y_d for a single frame, the camera action may become unstable, as described above. Thus, in our system, we let the constraint Δy be calculated by

$$\Delta y = w y_d. \tag{12}$$

Here, w is the weight of the relative reliability and is determined by

$$w = \frac{R_b}{R_l + R_r} \quad (R_b = \max\{R_l, R_r\}). \tag{13}$$

Here, R_l and R_r are calculated by Eq.(10). If the reliability of the lower reliability camera equals zero, w becomes 1, and if the reliability of the two cameras are identical, w becomes almost 0.5.

5 ACTIVE CAMERA CONTROL

In order to controlling the FV-PTZ unit, we employ the PID control scheme.

P component is assigned as the target angular velocity represented as follow:

$$\mathbf{v}_{\mathrm{obj}t} = \mathbf{v}_{\mathrm{Robj}t} + \mathbf{v}_{\mathrm{cam}t}. \tag{14}$$

Here, $\mathbf{v}_{\mathrm{Robj}t}$ and $\mathbf{v}_{\mathrm{cam}t}$ represent the relative angular velocity of the object and the rotational angular velocity of the active camera at time t. $\mathbf{v}_{\mathrm{Robj}t}$ is computed by using angle between the object center and the image center which is represented as $\mathbf{s}$.

$$\mathbf{v}_{\mathrm{Robj}t} = \frac{\mathbf{s}_t - \mathbf{s}_{(t-1)}}{\Delta t} \tag{15}$$

Here, Δt represents the time between continuous two frames. The system can know $\mathbf{v}_{\mathrm{cam}t}$ by response from PTU controller.

I component rectifies the difference between the object center and the image center and it calculated by follow:

$$\mathbf{v}_{\mathrm{dx}t} = \frac{\mathbf{s}_t}{\Delta t}. \tag{16}$$

Because the command format for PTU is angular velocity, I component has to be transformed to angular velocity form.

D component is represented by the angular acceleration calculated as follow:

$$\mathbf{a}_t = \frac{\mathbf{v}_{\mathrm{obj}t} - \mathbf{v}_{\mathrm{obj}(t-1)}}{\Delta t}. \tag{17}$$

Thus, the PID control scheme is suitable for simultaneously controlling the angular speed and position of the pan-tilt unit. Therefore, the PID-based pan-tilt control is effective for motion synchronization between the target and the active camera.

The control value $\mathbf{v}_{\mathrm{u}}$ is computed by

$$\mathbf{v}_{\mathrm{u}t} = \mathrm{K_p}\mathbf{v}_{\mathrm{obj}t} + \mathrm{K_i}\mathbf{v}_{\mathrm{dx}t} + \mathrm{K_d}\mathbf{a}_t\Delta t. \tag{18}$$

Furthermore, using the proposed system, we can obtain 3D position information of the target because two cameras are used. Therefore, the system can automatically control each camera's zoom and focus based on the estimated distance from the cameras. In our system, zoom is controlled for keeping the resolution of the target appearance.

Focus is controlled based on the relationship between the distance and the best focus value which has already been known by pre-experiment.

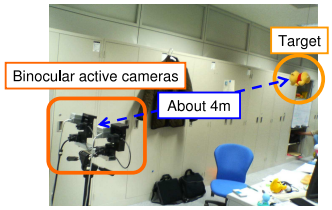


Figure 8: The Environment of Comparative Experiments.

6 EXPERIMENT

6.1 Comparative Experiment

We carried out comparative experiments to verify the performance of our proposed tracking system by comparing it with two other methods.

Method 1: Epipolar constraint is not used to control the active cameras. This implies that the two active cameras are controlled independently.

Method 2: The epipolar constraint is applied to the camera with the lower reliability and the weight factor is set as $w = 1$.

Method 3: The proposed method. The epipolar constraint is applied to the camera with the lower reliability and the weighted quantity of the constraint with a relative reliability is used.

Figure 8 shows the environment in which the comparative experiments were performed.

The target object is a doll suspended from the ceiling and swinging like a pendulum. In this experiment, the doll was at a distance of about 4 meters from the active cameras in the initial state.

We show a part of the sequence obtained in this experiment as captured by the right camera in Fig.9.

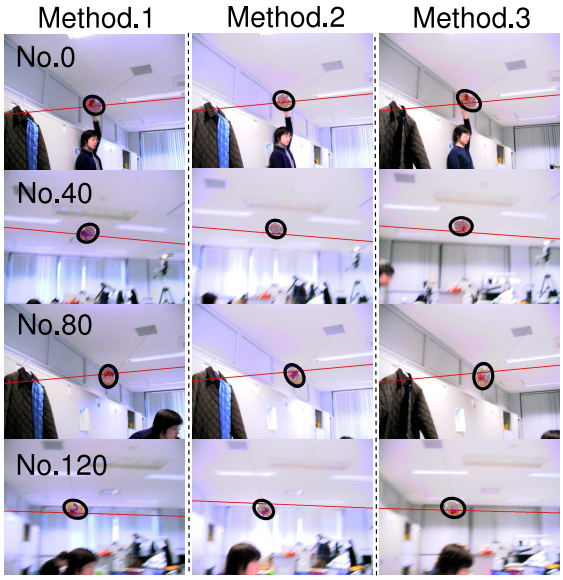


Figure 9: Several frames of the sequence obtained in the experiment.

To compare the three methods impartially, we allow the tracked target to assume the same motion in each experiment. We release the target object at the same height and allow it to move with an inertial motion three times.

The red lines in Fig.9 indicate the visual line of the left camera projected onto the image of the right camera.

To evaluate the performance of: *Controlling the binocular active camera to make the direction of the cameras intersect at a point in 3D space*, we used the error of the visual lines of the cameras and compared this among the three methods. As an evaluation measure, the error between each normalized vector of the epipolar planes calculated based on each camera was used. If the visual lines of the cameras perfectly intersect at a point, the error between each normalized vector will be zero.

We tracked the target object during 120 frames and calculated the absolute average and deviation of the errors between each normalized vector. Fig.10 shows a graph that indicates the error between each normalized vector with time and the absolute average and deviation of the error is shown in Table 1.

The error between each normalized vector in Method 3 was less that in Method 1. Therefore, our proposed method effectively achieves our goal. The error between each normalized vector in Method 2 was greater than that in the other two methods.

This is because the control of the directions of the active cameras became unstable since they could not respond quickly when the estimated ellipse center was constrained on the epipolar line. Next, we demonstrate the target pursuit of our proposed system. To evaluate it, we use the difference between the rotational velocities of the target and the active camera. This is calculated as the difference between the target ellipse center axis of the image in the current frame and that in the previous frame. We call this difference *position error between frames*. This number is equal to zero if the rotational velocity of the target equals that of one of the active cameras. On the contrary, if there is a difference between the velocities, the position error between frames increases.

Table 1: Average and deviation of the error between each normalized vector(unit: degree).

Method	Ave	Dev
1	0.84	0.28
2	1.06	0.71
3	0.33	0.20

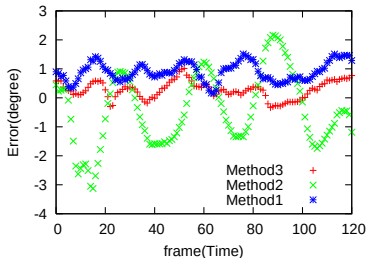


Figure 10: The error between each normalized vector.

In Table 2, we show the average and deviation of the absolute value of the position error between frames of both the cameras when experiments were conducted using Method 1 and Method 3.

The vertical deviation in Method 3 was marginally greater than that in Method 1 while the horizontal values were almost constant.

According to the result of this experiment, we verified that the tracking performance that uses the proposed epipolar constraint method will not degrade.

6.2 Tracking a Human Head

Figure 11 show several sequential frames that track a human head. The person in these frames walked

Table 2: The average and deviation of the absolute value of the position error between frames (unit: degree).

Method:direction	Ave	Dev
1:horizontal	3.8	2.7
3:horizontal	3.7	2.5
1:vertical	3.5	2.2
3:vertical	3.8	2.9

Table 3: The average and deviation of the absolute value of the horizontal position error between frames in the human head tracking experiment (unit: degree).

Right camera		Left camera		Summary	
Ave	Dev	Ave	Dev	Ave	Dev
2.4	2.6	2.1	2.2	2.2	2.4

Figure 11: Several frames in the sequence that track a human head.

straight from right to left. In frame no. 080, the person was closest to the cameras.

Table 3 shows the average and deviation of the absolute values of the horizontal position error between frames in the human head tracking experiment.

Because both the velocity and acceleration of the target were less than those in experiment **6.1**, the tracking error was smaller.

6.3 Zoom and Focus Control

Figure 12 shows several sequential frames that track a ball with automatic zoom-focus control. The purpose of the zoom and focus control is to maintain the tracked target in focus with constant resolution.

In order to test the effect of focus control, the target was defocused in the first frame. By comparing the images of the right camera with those of the left camera, it was observed that the size of the ball in the left camera images became smaller than that in the first frame. In contrast, the ball size in the right camera images remained unchanged because of the zoom control, which automatically zoomed in when the ball moved away from the camera.

Moreover, the defocus state in the first frame was automatically canceled by the focus control and the ball in the image was focused.

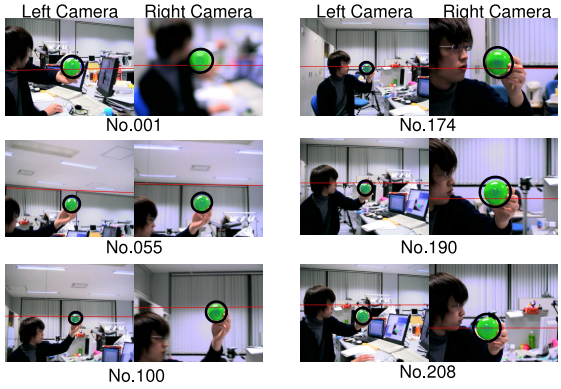


Figure 12: Several frames in the sequence for tracking a target in the zoom-focus control experiment.

7 CONCLUSION

In this paper, we have developed a high-performance object tracking system that can successfully capture high-quality images of a high-speed moving object at video rate. To increase the robustness and accuracy of object tracking in the video image, we introduced the concept of reliability into the *K-means Tracker*. In order to follow the movement of the object, two active cameras were controlled so that the object appeared at the center of the image plane. This was realized by positioning the optic axis of the two active cameras at the center of the object in the 3D space. To achieve this, we proposed the concept of relaxed epipolar constraint between the two cameras based on the reliability of object tracking and applied it to the control loop of the two active cameras. The extensive comparative experimental results demonstrated the usefulness and the effectiveness of our proposed method.

ACKNOWLEDGEMENTS

This research is partially supported by the Ministry of Education, Culture, Sports, Science and Technology, Grant-in-Aid for Scientific Research (A)(2)16200014, and (C)(2) 18500131.

REFERENCES

Bjorkman, M. and Eklundh, J. O. (2002). Real-time epipolar geometry estimation of binocular stereo heads. *PAMI*, 24-3:425–432.

C. Hua, H. Wu, Q. C. and Wada, T. (2006a). Kmeans tracker: A general tacking algorithm for tracking people. *Journal of Multimedia*, 4.

C. Hua, H. Wu, Q. C. and Wada, T. (2006b). Object tracking with target and background samples. *IEICE (accepted)*.

Comaniciu, D., Ramesh, V., and Meer, P. (2003). Kernel-based object tracking. *PAMI*, 25-5:564–577.

Coombs, D. and Brown, C. (1993). Real-time binocular smooth pursuit. *International Journal of Computer Vision*, 11-2:147–165.

Corke, P. I. and Good, M. C. (1996). Dynamic effects in visual closed-loop systems. *IEEE transaction on Robotics Automation*, 12-5:671–683.

Isard, M. and Blake, A. (1998). Condensation-conditional density propagatiojn for visual tracking. *IJCV*, 29-1:5–28.

Komuro, T., Ishii, I., Ishikawa, M., and Yoshida, A. (2003). A digital vision chip specialized for high-speed target tracking. *IEEE transaction on Electron Devices*, 50:191–199.

Matsuyama, T., Hiura, S., Wada, T., Murase, K., and Yoshioka, A. (2000). Dynamic memory: Architecture for real time integration of visual perception, camera action, and network communication. *CVPR*, pages 728–735.

Okada, R., Oaki, J., and Kondo, N. (2004). High-speed computer vision system for robot. *TOSHIBA Review (in Japanese)*, 59-9:29–32.

PRELIMINARY TESTS OF THE REMS GT-SENSOR

Eduardo Sebastián and Javier Gomez-Elvira
Lab. de Robótica y Exploración Planetaria, Centro de Astrobiología, Ctra. Ajalvir Km.4, Torrejón de Ardoz, Spain
sebastianme@inta.es

Keywords: Environmental monitoring, infrared temperature detector, system identification and sensor calibration.

Abstract: This paper describes and tests a mathematical model of the REMS GT-sensor (Ground Temperature), which will be part of the payload of the NASA MSL mission to Mars. A short review of the instrument most critical aspects like the in-flight calibration system and the small size, are presented. It is proposed a mathematical model of the GT-sensor based on an energy balance theory, which considers the internal construction of the thermopile, and allows the designer to model independently the change in any of its parameters. The instrument includes an in-flight calibration system which accounts for dust build up on the thermopile window during operations. Pre-calibration tests of the system are presented, demonstrating the good performance of the proposed model, as well as some required improvements.

1 INTRODUCTION

This paper describes a set of preliminary tests to validate a mathematical model of the REMS (Rover Environmental Mars Station) GT-sensor (Ground Temperature). The REMS is a meteorological station designed at the Centro de Astrobiología, which is part of the payload of the MSL (Mars Science Laboratory) NASA mission to Mars. This mission is expected to be launched in the final months of 2009. The detection of Mars surface temperature is essential to develop meteorological models of Mars atmospheric behavior (Richardson *et al.*, 2004). Mars suffers very extreme ground temperature gradients, from -135ºC to 40ºC between winter and summer. Also, differences of ±40ºC between the ground and the atmosphere at 1.5m over the surface are expected (Smith *et al.*, 2004).

The GT-sensor, as its name indicates, is dedicated to measure the brightness temperature of the Mars surface, using an infrared detector that measures the emitted thermal radiation. The detector focuses a large surface area, which is far enough from the rover as to minimize its influence, measuring the average temperature and avoiding local effects. The main GT-sensor requirement is to achieve an accuracy of 5K, in which the errors created by rover influence, ground emissivity uncertainty and sensor noise must be included.

The selected infrared detector is a thermopile. These sensors have the advantage that they can work at almost any operational temperature, are small and lightweight and comparative cheap, as well as they are sensible to all the infrared spectra. Taking into account the restricted resources available for the REMS, there is hardly any alternative to thermopiles. Contrary, thermopiles are not standard parts for space or military applications. Therefore, at present no formally space qualified thermopile sensors exist. It should be noted here, that the IRTM experiment on the VIKING mission and the MUPUS experiment of the ROSETA mission have proven the suitability of this kind of detector to measure low object temperatures under space conditions.

The paper is organized as follows; section 2 introduces a brief description the REMS GT-sensor. In section 3 the mathematical model of the sensor is presented. Section 4 shows preliminary real tests results, using the proposed model. Finally, section 5 summarizes the results.

2 THE REMS GT-SENSOR

The GT-sensor measures the emitted radiation of the Mars surface in two infrared wavelength channels, by using two detectors, looking directly to ground without any optical system. The selected measurement channels of the thermopiles are the 8-14μm and 16-20μm (Vázquez *et al.*, 2005). These channels avoid the abortion band of the CO_2 centred in 15μm (Martin, 1986), and minimize the influence

of sun radiation. The thermopiles are of the model TS-100 (IPHT, 2007), previously used for the ROSETA mission, which include a RTD sensor and a filter build to the specification and pre-bonded onto them as the thermopile window.

The use of two measurement channels is justified in two ways. First, each channel is specialised in the measure of a temperature range, based on the Planck law and higher S/N ratio. And second, the output signal of the two channels can be combined in order to apply colour pyrometry techniques. This can help to estimate the emissivity of the Mars ground, despite both channels appears to have nearly unit emissivity (Vázquez *et al.*, 2005).

The thermopiles are mounted inside a boom, figure 1, which is placed in the rover mast at 1.5m height. The boom has the form of a small arm of 150mm long, and it also hosts the electronics dedicated to amplify thermopiles signal. The boom is made of aluminium and is used as a thermal mass to ensure and acceptably low drift in thermopiles temperature. The boom's form permits to avoid the existence of lateral lobules in the thermopile FOV (field of view), minimizing the rover direct vision.

The GT-sensor includes an in-flight calibration system whose main goal is to compensate the detector degradation due to the deposition of dust over its window (Richardson *et al.*, 2004). The system is implemented, without moving parts, by a high emissivity, low mass calibration plate at a temperature of our choosing. It is placed in front of each detector, so that each detector looks at the ground through a hole in the plate. In this way the part of the FOV obstructed by the calibration system is an annulus, limiting the measurement solid angle.

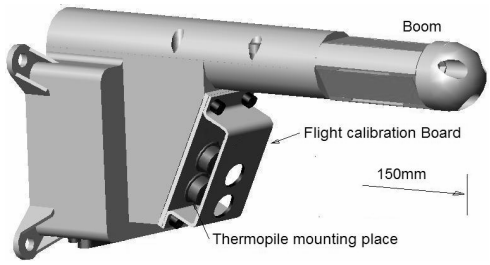


Figure 1: REMS boom and thermopile sensors.

3 MATHEMATICAL MODEL

Usually, the mathematical equation to model a thermopile considers it as a black box with an input, the incident energy, and an output, the output voltage. Therefore, a thermopile is characterised using a gain with units [V/W], which depends on thermopile temperature. This equation behaves properly for high target temperatures, and when no thermopile worsening is expected during operation. Essentially, if there is degradation, a parameterized model is required in order to compensate it.

Contrary, the proposed model is based on an energy balance theory (Richardson *et al.*, 2004), which considers the internal thermopile structure and operation. It behaves better for low target temperatures, and for a wide range of thermopile temperatures. It also permits to establish adaptation algorithms for the change in model parameters.

3.1 Thermopile Model Equations

The proposed model uses two equations. The first one shows the response of the thermocouples, which form the thermopile. The thermopile is integrated by 100 thermocouples connected in series and embedded between the can and the bolometer (IPHT, 2007). The equation (1) determines the relation between the thermopile output voltage and the temperature difference between the hot (bolometer, T_s) and the cold junction (can, T_c). Therefore, from the measurement of (T_c) and the output voltage (V_{out}), the value of T_s can be obtained.

$$V_{out} = f(T_c)(T_s - T_c), \quad (1)$$

The function $f(T_c)$ can be approximated by a polynomial expression provided by the thermopile manufacturer, which depends on thermopile or can temperature (T_c).

The second one is the energy balance equation, and it accounts for the heat fluxes into the thermopile bolometer from all the bodies around it. As the bolometer is designed to be well insulated from the can and to have low thermal mass, the equilibrium condition of the equation, is reached after a setting time of a few milliseconds. The equation considers a simplified model of energy exchange by thermal radiation (Q_R) and conduction (Q_C), see figure 2.

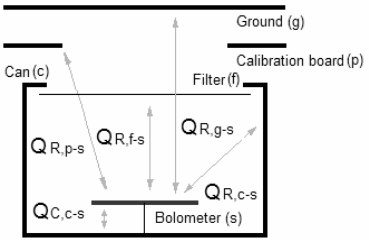


Figure 2: GT-Sensor energy terms.

From the analysis of the energy terms, and after the thermal equilibrium is reached, the equation (2) represents the thermal circuit

$$Q_{R,g-s} + Q_{R,p-s} + Q_{R,f-s} + Q_{R,c-s} + Q_{C,c-s} = 0 \quad (2)$$

Based on simplified heat flux models, the equation (2) can be expressed in the following way,

$$\begin{aligned} 0 = (1-\alpha)\cdot K_1\cdot\left(E_g^I - E_s^I\right) + \alpha\cdot K_1\cdot\left(E_p^I - E_s^I\right) + \\ K_4\cdot\left(E_f^O - E_s^O\right) + K_2\cdot\left(E_c^T - E_s^T\right) + K_3\cdot\left(T_c - T_s\right) \end{aligned} \quad (3)$$

where α represents the factor of the thermopile FOV obstructed by the flight calibration board, and K_1, K_2, K_3 and K_4 are constants which modulate the weight of the different terms. These constants depend on physical factors of the bodies around like: emissivity, FOV factors, viewed areas, and in the case of K_3 on thermal conductance of the materials. The energy terms E_x^y are calculated integrating the Planck law (4) for each body temperature (T_x).

$$E_x^y = \int_{y1}^{y2} T(\lambda)\cdot 2hc^2 \Big/ \lambda^5\cdot\left(e^{hc/\lambda KT_X} - 1\right) d\lambda \quad (4)$$

where the subscript x represents the body, g(ground), p(calibration board), f(filter), c(thermopile can) and s(bolometer). $T(\lambda)$ is the transmittance of the filter. And the superscript T, I or O denotes if the energy flux is calculated in the total spectra, in band or out of filter band respectively.

3.2 Calibration System Equations

The main origin of thermopile degradation, while operating on Mars conditions, is dust deposition. During landed operations dust will collect on the thermopile's filter. Dust, which has high emisivity, will block light both into and out the detector, and it will equilibrate to the same temperature as the filter it is now in contact. It can therefore be seen as a changing the area of the filter into something similar to the can. In other words, if the factor β represent the part of the FOV that has not been obstructed by the dust, the equation (3) can be rewritten,

$$\begin{aligned} 0 = \beta\cdot(1-\alpha)\cdot K_1\cdot\left(E_g^I - E_s^I\right) + \beta\cdot\alpha\cdot K_1\cdot\left(E_p^I - E_s^I\right) + \\ \beta\cdot K_1\cdot\left(E_c^O - E_s^O\right) + (1-\beta)\cdot K_1\cdot\left(E_c^T - E_s^T\right) + \\ K_2\cdot\left(E_c^T - E_s^T\right) + K_3\cdot\left(T_c - T_s\right) \end{aligned} \quad (5)$$

The equation (5) includes two simplifications: The filter temperature is supposed to be equal to the can temperature, and the factor that weights the filter influence K_4 is equal to the ground factor (K_1), due to both shares the same FOV.

Therefore, it is the factor β that must be determined during operations. This can be done by varying the temperature of the calibration board if the ground brightness temperature can be trust to remain constant while the temperature changed (Smith *et al.*, 2004). The temperature of the thermopile, the flight calibration board, and the output voltage of the thermopile must be collected before and after the temperature changed. Finally, using the data collected and the equation (5), the system of equations (6) can be defined,

$$0 = \beta\cdot\left[a\cdot E_g^I + b\cdot E_{p1}^I + c_1\right] + d_1 \quad (6.1)$$

$$0 = \beta\cdot\left[a\cdot E_g^I + b\cdot E_{p2}^I + c_2\right] + d_2 \quad (6.2)$$

where a, b, c and d are a set of known energy terms. And, the system can be solved for the factor β,

$$\beta = d_2 - d_1 \Big/ b\cdot\left(E_{p1}^I - E_{p2}^I\right) + c_1 - c_2 \quad (7)$$

4 TEST RESULTS

In this section, four preliminary experimental tests dedicated to validate and show the performance of the sensor model are presented. The tests pretend to be a simple exercise in ambient conditions of the experiments to calibrate the REMS GT-sensor.

Prior to start with the description of the tests, it is necessary to define the experiment setup, figure 3. A thermopile with a band past filter of 8-14μm, looking at the calibrated blackbody source MIKRON M315 and covering its all FOV, was used. The temperature of the flight calibration board and the thermopile's can have been measured using two individual T type thermocouples, glued to these elements. The temperatures of both elements have been controlled using two control systems CAL3200 and the associated thermocouple.

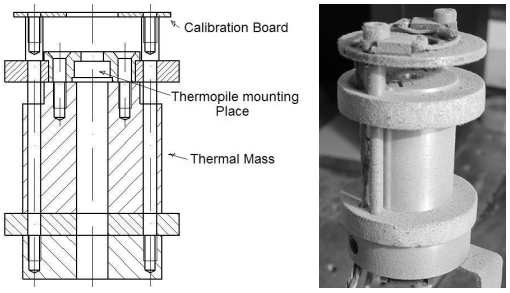


Figure 3: Thermopile model layout and real test model after dust deposition.

The first test tries to identify the value of those unknown constants of the thermopile model, table 1. In order to do it, different blackbody and thermopile temperatures were consigned, figure 4, while the flight calibration board was removed, which means that α is equal to 0. Therefore, based on the energy balance equation (2), where the energy terms are known, a least-squares problem for the tested points is established. Finally, the values of the constants, which minimize the least-squares error, are obtained.

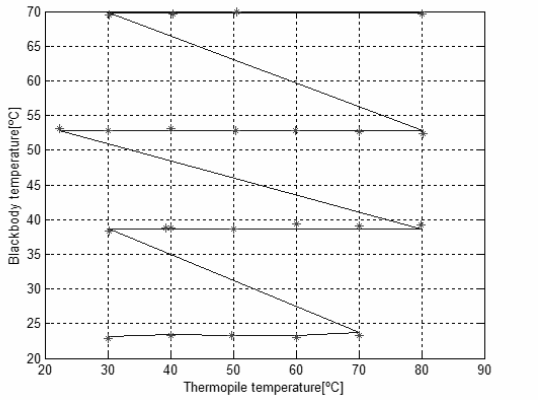


Figure 4: Real blackbody temperature (-) and estimated blackbody temperature (*) after the identification process.

Table 1: Thermopile and model variables.

Variable	Value
K_1	1
K_2	28.1695
K_3	128.4088W/K
$Error_{RMS}$	0.28K
α	0.34
Polynomial coefficients	[1.0826 -4.0577]
β	0.415

The second test has been carried out with the purpose of identifying the factor of the FOV obstructed by the flight calibration board, this is α. This value is an essential parameter, necessary for the flight calibration process.

During the test, the thermopile and the calibration board must be kept at ambient temperature, to ensure that their temperatures are homogeneous and stable. This requirement is necessary due to the radiance of the flight calibration has not previously calibrated, and in this way the error introduced by this factor is avoided. The blackbody temperature is set over the ambient temperature. In order to avoid the thermopile heating, due to the energy radiated by the blackbody, an opaque surface was introduced in between. This surface was removed during the measurement time of bodies temperature and thermopile output. From these data, the energy terms of equation (2) were calculated, and we were able to solve for the values of α, for each blackbody temperature. Finally, these values of α were averaged in order to obtain a unique value, table 1.

The third test pretends to know the real temperature of the calibration board, which is required to calculate its real radiometric emission. The test consists of varying the flight calibration board temperature over the temperature of the thermopile, while the temperature of the thermopile and the blackbody are kept constant.

In this case, for each calibration board temperature, the blackbody and the thermopile temperatures were collected, as well as the thermopile's output. Therefore, based on these data and the energy valance equation (2), the radiometric emissions of the flight calibration board are derived, and from them the real temperatures, which were compared with the measured temperatures to determine the absolute calibration error, figure 5. Also, a first order polynomial, interpolating this error, has been obtained, table 1.

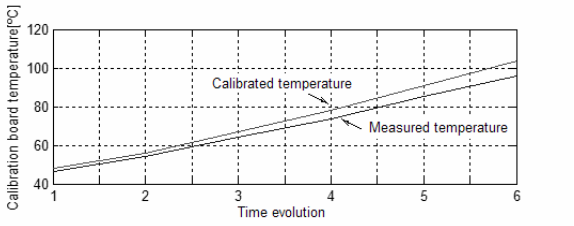


Figure 5: Temperature calibration error of the flight calibration board.

The final test is dedicated to analyse the behaviour of the in-flight calibration algorithm, after depositing a certain among of dust over the thermopile window, simulating Mars environment.

The test is divided in two different steps. In the first one, the flight calibration algorithm is run for different calibration board temperatures. The figure 6(Top) shows the obtained values of β, with and without considering the previous calibration of the calibration board. The data after calibration are more stable, validating this calibration and reducing algorithm error. As a result the average value of β is shown, table 1.

The second step is dedicated to measure the temperature of the blackbody from: the thermopile output, the calibrated thermopile model, the calculated value of the factor β due to dust deposition. During the test, the temperature of the blackbody was almost constant, while the

temperature of the thermopile was changed. Figure 6(bottom) shows the result after applying the measurement algorithm (5) and solving for T_g. The high temperature error is due to the flight calibration system is calibrating a small surface or annulus in the external part of the thermopile filter, while the filter surface used to measure the ground brightness temperature is in the middle, and the deposition appears to be no homogeneous. Thus, the obstruction factor β is higher than the real one.

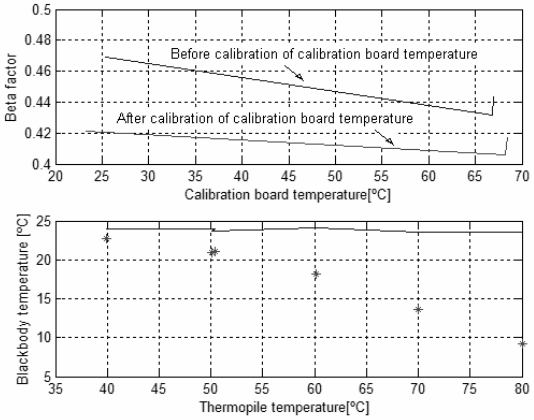


Figure 6: (top)Value of β factor. (bottom)Real blackbody temperature (-), and measured blackbody temperature (*).

5 CONCLUSIONS

The thermopile mathematical model presented in this paper is a valid and precise method to characterize a thermopile, due to the low least-squares error of 0.28K for an extensive thermopile temperature range of almost 60K.

The FOV obstruction factor, generated by the flight calibration board, reaches a value of 34%. It can be reduced in order to increase the S/N ratio for ground temperature signals.

The radiometric calibration of the flight calibration board is necessary due to the error introduced by different factors: the calibration of the temperature sensor, the temperature homogeneity of the calibration board and the position and anchoring of the temperature sensor.

Dust deposition, based on dust electrical characteristics, tends to form small balls around the union between the thermopile can and window. This is exactly the area calibrated by the flight calibration board, justifying the higher value of β. As a future work, a new calibration system, using a heated cable, will be studied. The cable will cross the FOV of the thermopiles, obstructing a homogenous part of the FOV, and not only a ring in the most external part. This will minimize the error generated by the way dust is built up on the window.

ACKNOWLEDGEMENTS

The authors would like to express special thanks to all members of the REMS project who in different ways are collaborating in the development of REMS GT-sensor.

REFERENCES

Richardson M., McEwan I., Schofield T., Smith M. Souères P., Courdesses M. and Fleury S. 2004. MIDAS Mars Ice Dust Atmospheric Sounder. *MSL proposal.*

Vázquez L., Zorzano M.P., Fernández D., McEwan I. 2005. Considerations about the IR Ground Temperature Sensor. *CAB, REMS Technical Note 1-101722005.* Madrid.

Smith M.D. *et al.*, 2004. First Atmospheric Science Results from the Mars Exploration Rovers Mini.-TES. *SCIENCE EEE, 306*, 1750-1753.

Martin T.Z. 1986. Thermal infrared Opacity Of The Mars Atmosphere. *Icarus, 66*, 2-21.

www.ipht-jena.de. 2007. IPHT web page.

SHORT PAPERS

SAFETY VALIDATION OF AUTOMATION SYSTEMS: APPLICATION FOR TEACHING OF DISCRETE EVENT SYSTEM CONTROL

Pascale Marange, François Gellot and Bernard Riera
Centre de Recherche en STIC - UFR des Sciences Exactes et Naturelles
Université de Reims Champagne-Ardenne, Moulin de la Housse, BP 1039, 51687 REIMS Cedex 2
{pascale.marange, francois.gellot, bernard.riera}@univ-reims.fr

Keywords: Discrete event systems, Validation, Control, Functional identification, Learning.

Abstract: We propose in this paper, to introduce a method to validate logic controller programs adapted to the teaching of Discrete Event Systems. The use of real systems for teaching raises two problems. The first one concerns the security of human beings (students and teachers) and materials. The second problem is the necessity to be able to detect possible errors done by students and to bring an explanation. We propose a method to define a level of system abstraction, to validate the student's control by the mean of a validation filter placed between the plant and the controller. The specifications contained in the filter make it possible to detect errors and to generate an explanation automatically. We applied this method to an original project where it was proposed to 7 year-old children, to discover automation, by programming a tablets packaging system.

1 INTRODUCTION

The implementation of a control in a PLC (Programmable Logic Controller) raises necessarily validation problems: "Is the running (plant: PO and control part PC) safe?"; "Is the specification respected?", and if it is not the case, "Which control errors have been done?"… Our research aims at ensuring that the control is valid with respect to the safety and running technical system requirements. For that, we propose to set up a module which validates on line the logic controller program. At each evolution of the system (PO and PC), the validation module authorizes or not the outputs sent from the PC to PO. This work finds an interest in the field of remote maintenance as well as education. The paper focuses on the last point. We are interested in the problem of the control validation carried out by students in automatic-control during work practise in the field of the Discrete Event System (DES) and the PLC.

The teaching of automatic control in broad sense requires the transfer of knowing and know-how to learners. Know-how concerns for instance the use and the programming of PLC by means of software respecting standards like IEC 1131.3. The acquisition of this technical know-how requires practical work in specialized and expensive rooms including PLC and simplified manufacturing systems which are a replica on a reduced scale of real systems found in the industry. The use of PLC raises two problems: safety and explanations. Indeed, if a programming error occurs, safety has to be guarantee for materials as well as human operators being around and explanation has to be given about the error and its effects on the system. The suggested solution in this paper is articulated around a validation filter defined by means of safety and liveness constraints. The first guarantee the system safety by prohibiting any evolution being able to deteriorate it. The second make sure that the suggested control is coherent with the running specification (defined in our case by the teacher). It is important to note that the constraints definition is related to the possibility of learner's authorized actions (i.e. errors done by learners depend on the possibilities that he has to act on PO). It is very interesting in the pedagogy field, to propose various actions levels related to various abstraction levels. To achieve this goal, the constraints are defined in reference to a PO functional identification which makes it possible to fix the abstraction or granularity degree chosen by the teacher. The use of constraints makes possible to supply explanatory capacities to the validation tool. That makes it possible to guarantee an efficient Human Machine dialogue.

In a first part, the suggested approach of validation is presented in a general way. This one is thus based on a functional identification of the system. The functional model obtained is used to define the selected abstraction degree. The writing of the constraints is based on an original classification which distinguishes the constraints not only according to their type (safety and liveness) but also according to their intrinsic characteristic (combinative or sequential). This classification is the object of second part. In a third part of the paper, the approach is applied to a concrete example where we propose to "young novice" engineers to control a packaging system.

2 VALIDATION APPROACH

Work in the field of the automatic control validation aims to insure that mathematical properties are respected by model (Canet, 2001), (Lampérière and al, 2000). The work undertaken within the framework of tool UPPAAL (Behramm and al, 2004) defines three types of properties: attainability, safety and liveness. We chose to use the safety constraints: what the system should not do, and liveness constraints: what the system should do according to the running specification. The validation can be considered off line or on line. In the first case, the control is completely validated before being sent to the PO (Machado, 2006). Within this framework, we proposed an off line approach (Tajer and al., 2006) based on the Ramadge Wonham supervisory control theory (Womham and Ramadge, 1987) and the synthesis algorithm by Kumar (Kumar, 1991). The suggested approach makes it possible to guarantee that the control behaviour is safe, deterministic and without deadlocks. However, it presents several disadvantages: the combinatorial explosion, the difficulty to give a comprehensible explanation to learner. So, we directed our work towards an on line approach of control validation.

The idea is to inhibit the evolutions which can lead the system to a situation of deterioration, of setting in danger of the operators or which does not respect running specification. Cruette's work (Cruette, 1991) for the monitoring of automation systems proposes to intercalate a filter between the PO and control. The filter ensures on the one hand coherence between the output and the expected one, and on the other hand coherence between the evolution of the expected PO and that produced. This idea of an approach on line by filter is taken up partially and adapted to ensure the control validation i.e. with each new control evolution, the filter receives in inputs: the evolution of the outputs (controllable events *Ec*: actuators) coming from the control designed by the student as well as the evolutions of the inputs (uncontrollable events *Euc*: sensors) of PO. In the same time that the command execution, the validation filter authorize or not the new evolution. For that, the filter contains the safety and liveness constraints and according to the sensor and actuators information, it tests the constraints. If all constraints are true, the evolution is authorized and the control continues. If not, the evolution is prohibited and the system is stopped.

2.1 Functional Identification

The suggested approach is based on a hierarchical functional identification of the system. The first idea consists in using the functional decomposition to determine the authorized student's actions. Indeed, it seems us that for a beginner for example his possible actions on PO must be reduced. The second idea deals with a constraints definition on two levels: One on the low level (sensor actuator level) to ensure the safety and the other based on the functional decomposition to detect programming errors and to generate an explanation (high level). That means that the constraints will be defined starting from the functions that learner is allowed to implement.

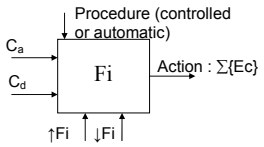


Figure 1: Definition of « Function » notion.

To identify the system functions, of the methods as SADT (I.G.L technology, 1989), MERISE (Tardieu and al., 2000) make it possible to cut out the system in functions and under functions according to a downward hierarchical approach. Within the framework of the suggested approach, it is necessary for each function identified to know the activation conditions *Ca* (initial conditions), the deactivation conditions *Cd* (function goal) and its execution mode (controlled or automatic) (figure 1). The procedure will make possible to define an additional degree of freedom in the control, for a level of functional decomposition fixed. In the case of the controlled mode, the learner must manage the activation and the deactivation of a function, when the conditions become true. On the other hand, in the

case of the automatic mode, student must only activate the function which is deactivated when the deactivation conditions are true.

The model of figure 1 shows that each function can be activated ($\uparrow Fi$) or be deactivated ($\downarrow Fi$). The action of activation is always controllable. On the other hand, the action of deactivation is controllable when the execution is in controlled mode and uncontrollable when the procedure is automatic. According to the selected granularity (the degree of decomposition), the term "function" represents the control (activation, deactivation) of a whole station as well as a simple actuator. The decomposition or abstraction degree will allow a teaching at various levels and adapted to the learner knowledge.

2.2 Use in the Teaching

In the framework of the DES teaching, the level of granularity will allow to propose more or less difficult and evolutionary exercises adapted to the training level. The granularity is at the responsibility of the teacher who must adapt this one to the level of learning. Indeed, if teaching is addressed to:

- a novice, description can stop at the functional level (high level) of the plant. It is the teacher who gives the control of each function and learner has to provide the chronology.
- a beginner, with regard to the difficulty of each function, he can control several of them (high level) and program completely the others (low level). It is the teacher who has to decide which functions are automated.
- an advanced student is able to control the system as a whole and thus he acts on the plant at the low level.

Once the teacher has defined the granularity of the system, he can choose according to two procedures already presented.

The validation approach must make possible to ensure the safety and the respect of the running specification. For that we propose to set up safety and liveness constraints which are defined starting from the functional identification of the procedure. The suggested method requires to make some assumptions and to specify certain terms. The functional identification gives a finished set of functions and the functions are independent to each other. Moreover, it is supposed that it is not possible to have multiple activation of the same function (i.e. an only instance at a time).

On the low level, the description of a control model can be made by means of the input/outputs vector. The vector value at time *t* corresponds to the current state of the system. By analogy, on a fixed level of functional decomposition, it is possible to define a vector of state corresponding to the input-outputs of the function. The inputs are then the activation and deactivation conditions of the function and the outputs the actions related to the function represent.

3 VALIDATION FILTER

To ensure the safety and the correct running of the system, the validation filter uses some specifications to detect and to bring an automatic explanation. The definition of the specifications is not easy. Thus, we propose to carry out a distinction on their role (safety or liveness) and on their intrinsic characteristics (combinational, sequential, dynamic or static).

3.1 Safety Validation of System

The safety constraints characterize what the system should not do. It seems us important to place the safety constraints at the sensors - actuators level. Three types of safety constraint are defined.

3.1.1 Static Safety Constraint

The static safety constraints express physical and technical impossibilities of the system elements. The static safety constraints depend only on the controllable events. The Syntax is: $C = Ec_i \wedge Ec_j$. For example, if the event Ec_1 cannot be carried out at the same time as the event Ec_2, then: $Ec_1 \wedge Ec_2 = 0$.

3.1.2 Dynamic Safety Constraint

The dynamic safety constraints relate to the occurrence of an event which is not compatible with an other event. The event corresponds either at the activation of controllable event ($\uparrow Ec$), or with the validation of the deactivation condition ($\uparrow Euc$):

- In the first case, the constraint is written in the following way: $Euc_i \wedge \uparrow Ec_j = 0$. Indeed, if the deactivation conditions are present, the sending of the associated controllable event is prohibited.
- In the second case, the constraint is written: $Ec_j \wedge \uparrow Euc_i = 0$. Indeed, as soon as the deactivation conditions are present, the actuator must be deactivated.

The safety constraints make it possible to protect the system against deteriorations. For these constraints, the validation filter can be placed in the

control part (PLC). The validation filter prohibits the sending of a command if this one of the safety constraints does not respect constraints.

The definition of the safety constraints is re-used at the functional level, in a redundant way to bring an automatic explanation to the learner's errors:

- If functions cannot be activated at the same time: $F_i \wedge F_j = 0$
- If the deactivation condition of a function is present, the sending of the function is prohibited: $Cd\_F_i \wedge \uparrow F_i = 0$.
- If the activation condition is not true, the function cannot always be activated: $/Ca\_F_i \wedge \uparrow F_i = 0$.
- As soon as the deactivation condition is true, the function must be deactivated: $F_i \wedge \uparrow Cd\_F_i = 0$

It is necessary now to determine if functioning is correct compared to the running specification. For that it is proposed to set up liveness constraints.

3.2 Liveness Validation

The control validation compared to functioning, goes through by the definition of liveness constraints (what the system must do compared to the running specification). Contrary to the safety constraints, the liveness constraints are placed only at the functional level. Two types of constraints are defined: combinational and sequential liveness constraints.

3.2.1 Combinational Liveness Constraints

The combinational liveness constraints allow activation or deactivation when the conditions are present. The combinational liveness constraints are defined in a similar way to the dynamic safety constraints. For example the function F_i can occur only under the condition Ca_i: $\uparrow F_i \wedge Ca_i = 1$ or the function F_1 must be deactivated when the condition Cd_i is true: $\downarrow F_i \wedge Cd_i = 1$.

3.2.2 Sequential Liveness Constraints

By the sequential liveness constraints, the function sequencing is described. The idea is thus to represent the sequence described by the running specification without to describe one unique behaviour. The logical equations do not make it possible to manage this sequential aspect simply.

The possibility to carry out a function compared to the expected behaviour depends on the system situation, i.e.: the functions which have been carried out. We point out that the possibility to carry out a function compared to the system state is expressed by the combinational liveness constraints. To take into account the functions sequencing, for each function, we define the deactivation conditions and the functions which had to be fulfilled. In the same way, the function execution will influence the future behaviours and thus the functions which will not be realizable any more. To express the functioning sequencing, it is proposed to draw up a table with information: the deactivation conditions, the functions which had to precede, the functions which will not be realizable any more. For each function, we define Grafcet with the states {not carried out, in execution, carried out}. Grafcet evolves at the same time as the command. Grafcet makes it possible to know the functions authorized or not compared to at functioning awaited.

According to the functional identification, if the functions are carried out the ones after the others or in parallel, all the constraints will not be defined. Indeed, if the execution is in automatic mode, the function is deactivated automatically when the deactivation conditions are present. In this case, it is not necessary to define the dynamic safety constraints on the uncontrollable events.

3.3 Use in the DES Teaching

Within the framework of teaching, we can propose a tool that ensures in priority the system and operators safety, thanks to the definition of the safety constraints on the sensor actuator level. The definition at the functional level, of the safety constraints makes it possible to generate simply and automatically explanations related to an error. The detection and the management of the errors will be done in a simple way by the constraint validation or not. During the validation stage, three different cases are possible:

- All the specifications are validated, the validation filter allows the sending of the controllable event to the plant
- One or more safety specifications or constraints are not respected. In this case, the validation filter does not authorize the controllable event and informs the student on the specifications which are not respected.
- One or more liveness constraints are not respected. That means that the running specifications are not all respected. In this case, if all safety constraints are OK, controllable events can be accepted because there is no risk for the system. The explanation generation associated with an error, is also done in a simple way. Indeed, the distinction that we could establish in the two previous parts for the safety and liveness constraints enables us to find an automatic explanation to the non respect. The safety constraints must be validated permanently

independently to each other. If there is one or more safety constraint violation that means:

- For the static safety constraints, the control wants to send an order whereas the system is making the contrary order.
- For the dynamic safety constraints on an uncontrollable event, the non deactivation of an order whereas a sensor indicates that it should be deactivated.
- For the dynamic safety constraints on a controllable event, either the associated sensor with this action is already in the wished state, or the evolution of this order is impossible compared to the system conditions.

For the combinational liveness constraints, we use the same approach as previously, if the constraint equation is not respected, that means that the control wants to send an order whereas it is not the waited behaviour. The automatic generation of explanation for the sequential liveness constraints is more difficult, because they are not defined by logical equations. We can go up the evolution which has just occurred by the mean of the active states in the different Grafcets. With this information, the teacher should find by himself an explanation.

4 APPLICATION

The idea was to collaborate (Riera et al., 2005) with a teacher of "primary" school. We wanted to allow the child to discover and to control really the system by programming his/her own sequence.

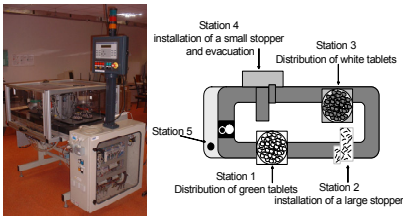


Figure 2: Productis Machine.

The system used for this project is the "Productis" machine. This system allows the packaging of tablets (figure 2). The system is composed of 5 stations and a conveyor: Station 1: distribution of green tablets by counting, Station 2: installation of a large stopper on the large tube, Station 3: distribution of white tablets by counting, Station 4: installation of a small stopper on the small tube and/or evacuation of the tube in a box, and Station 5: Feeding of the system. In order to make the activity of control design funny for the child, we propose the following original scenario. The instructions to use the machine have been lost. So, it is impossible for us to manufacture drug to heal sick fairies. Children have to find the running of the machine in order to manufacture specific drug. We have to adapt the vocabulary used to describe the system, at the age of the children. The activity is proposed with children, they know to rebuild a history according to a chronology. The children must create a sequence of functions to manufacture a tablets bottle. The functions execution is done in automatic mode. The child sends the order to activate a function which is automatically deactivated when the deactivation conditions are true. The functions are entirely carried out the one after the others. For this activity, we decided that the simultaneous sending of functions was impossible. Only one pallet circulates in the system in order to simplify the system comprehension. After functional identification of the system, we selected 20 functions could be programmed by children. For that, we analysed the system by stations. The pallet is manually loaded (station 5). The child presses on a button to release the pallet. We choose to describe only the station 4: positioning of a small stopper and evacuation. This station is composed of a prehensor, i.e. two cylinders (one for the vertical movement and one for the horizontal movement), and a vacuum system. To install a stopper, it is necessary to position the cylinder to the top, go down, take the cap, go up, advance the cylinder, go down and release the aspiration. In order to avoid synchronization in the control program designed by children, functions "put the stopper" have been divided into respectively two functions: "Take the stopper" and "Loosen the stopper". With regard to the functional analysis, children also have to program the control of the ejection by the mean of the gripper. In this part, only the constraints at the functional level of station 4 will be developed and explained:

- Static safety constraints: it is not necessary because the learner cannot send several actions simultaneously.
- Dynamic safety constraints on a controllable event:

$/up_4 \wedge (\uparrow Go\_out \vee \uparrow Go\_in) = 0;\ \uparrow Close_4 \wedge (down_4 \wedge in_4) = 0$

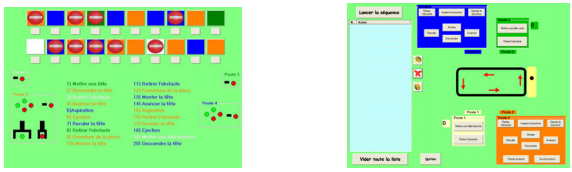
- Dynamic safety constraints on uncontrollable events are not necessary because learner must not deactivate the function. The execution of the functions is in automatic mode
- Combinational liveness constraints:
  - the aspiration of the stopper is authorized only in the position : down and in (in the same way for the gripper)

$\uparrow Take_4 \wedge (down_4 \wedge in_4) = 1$; $\uparrow Close\_gripper4 \wedge (down_4 \wedge in_4) = 1$

- the ejection of stopper can only be done in the position : down and out

$\uparrow Loosen_4 \wedge (down_4 \wedge out_4) = 1$; $\uparrow Open\_gripper4 \wedge (down_4 \wedge out_4) = 1$

- Sequential liveness constraints have to ensure that: the bottle is closed before carrying out the function of bottle evacuation.



a) "Step by step" mode b) Sequence mode

Figure 3: Interfaces.

The activity with the children proceeds in two parts. In the first, the child has at his disposal an interface (figure 3.a) with 20 buttons. The 20 buttons represent the 20 functions of the system. In this activity, the child has to understand the role of each button. For that, the child presses a button of the interface. Thus, the child causes the movement or the movements corresponding to the function on the machine and he has to associate a function to a button. According to the state of the system, all the buttons are not activated. For example, if the cylinder of station 4 is in position "in", the button "To Go_in cylinder" of station 4 cannot be pressed. After having understood the role of each button, the child can perform the second part of work (second interface). During the second activity (figure 3.b), the child programs his own sequence of functions to build a bottle of drugs. The sequence execution is validated on line. Hence, if the constraints are respected, the function having to be performed is performed, and the sequence can continue. If not, the validation system informs the child what are the constraints which are not respected.

5 CONCLUSIONS

We bring in this article some answers to the problems raised by the provision of automated material, for the teaching of automated systems control. For that, a validation approach on line by filter was proposed. This approach makes it possible to filter the evolutions which are dangerous for the system, or which do not answer the running specifications. The proposed approach can generate automatically an explanation. For that, the validation filter uses safety and liveness constraints of which definitions have been proposed. The modelling of sequential liveness constraints is a point that has to be developed, particularly to be able to generate automatic explanations. The proposed modelling of sequential liveness constraints has been designed to manage only one product in the manufacturing system. This extension has to be thought of doing. In addition, we also must improve the error explanation stage for the teacher. It seems possible, for example, at the same time as the system evolution (real plant) to use a simulated plant where the errors effects are displayed. In the simulated plant, learner could observe the consequences of his error.

REFERENCES

Behramm G., David A., Larsen K.G., A tutorial on UPPAAL, novembre 2004

Canet G., Vérification automatique de programmes écrits dans les langages IL et ST de la norme IEC, *thèse de doctorat*, Ecole Normale Supérieur de Cachan. December 2001

Cruette D., Méthodologie de conception des systèmes complexes a événements discrets : application à la conception et à la validation hiérarchisée de la commande de cellules flexibles de production dans l'industrie manufacturière, *Thèse de doctorat, Université de Lille*, 1991

International Electrotechnical Commission, Preparation of function charts for control systems, International Standard, *CEI/IEC 848*, 1991 (revised version).

I.G.L. Technology, SADT, un langage pour communiquer, *Eyrolles*, Paris, 1989.

Kumar R., Supervisory Synthesis Techniques for Discrete Event Dynamical Systems, *Thesis for Ph. D. Degree*, Université du Texas, 1991.

Lampérière S., Lesage J.J, Formal verification of the sequential part of PLC programs, *Proc. Of 5th IFAC Wodes*, pp 247-254, Ghent, Belgium, August 2000

Machado, Influence de la prise en compte d'un modèle de processus en vérification formelle des systèmes à événements discrets, *Thèse de doctorat de l'école normale supérieure de Cachan et de l'université de Minho (Portugal),* juin 2006

Riera B., Gellot F., Marangé P., Chemla J-P., Sayed Mouchawed M., Un projet original en commande et supervision des systèmes automatisés : Des enfants de 5ans au secours d'animaux malades !, CETSIS'05, Nancy, France, 25-27 octobre 2005

Tajer A., Marangé P., Gellot F., Carré Ménétrier V., Synthèse d'une commande supervisée à base de contraintes logiques, *revue électronique e-STA* , 2006

Tardieu H., A. Rochfeld, R. Colletti, La méthode Merise – Principes et outils, *Edition d'organisation*, 2000

Wonham W. M., Ramadge P.J., On the supremal controllable sublanguage of has given language, *SIAM J Control Optimization*, flight 25, n°3, p.637-659, 1987

A SAMPLING FORMULA FOR DISTRIBUTIONS

W. E. Leithead and E. Ragnoli
Hamilton Institute, NUI Maynooth
william.leithead@nuim.ie, emanuele.ragnoli@nuim.ie

Keywords: Sampled-data systems, Frequency response, Multirate systems, Hybrid systems, Discrete-time systems.

Abstract: A key sampling formula for discretising a continuos-time system is proved when the signals space is a subclass of the space of Distributions. The result is applied to the analysis of an open-loop hybrid system.

1 INTRODUCTION

Consider the hybrid system of Figure 1, where $x(t)$ and $y(t)$ are input and output, $(A/D)_T$ is an A/D converter with sampling period T, $(D/A)_T$ is a zero-order hold (ZOH) and P and C are the plants of a continuous time system and a discrete time system, respectively. In order to perform the transform domain analysis of the hybrid system of Figure 1, the transform domain response of a sampled signal must be related to the transform response of its correspondent continuous time signal. This is done by building the transform response of the sampled signal upon the superposition of infinitely many copies of its continuous time transform response, using the formula

$$G_d(e^{st}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} G(s+jk\omega_s) \qquad (1)$$

where G is the Laplace transform of a continuous time signal g, G_d is the z transform of the sequence of its samples $\{g(kT)\}_{k=0}^{\infty}$ and T and $\omega_S = 2\pi/T$ are the sampling period and the sampling frequency, respectively.

Till 1997, with the publication of (Braslavsky et al., 1997), 1 was mathematical folklore. In fact, it was very often used in the digital control literature ((M.Araki and T.Hagiwara, 1996), (J.S.Freudenberg and J.H.Braslavsky, 1995), (T.Hagiwara and M.Araki, 1995)), (Leung et al., 1991), (Y.N.Rosenvasser, 1995a), (Y.N.Rosenvasser, 1995b) and (Yamamoto and Araki, 1994)) and it appeared in many control textbooks ((K.J.Astrom and B.Wittenmark, 1990), (T.Chen and B.A.Francis, 1995), (G.F.Franklin and M.L.Workman, 1990)), (B.C.Kuo, 1992) and (K.Ogata, 1987)), but it was not established by a rigorous proof that indicated the relevant classes of signals considered.

Attempts to provide 1 with a proof are in (E.I.Jury, 1958), (K.J.Astrom and B.Wittenmark, 1990) and (T.Chen and B.A.Francis, 1995). Those proofs are based on the use of impulse trains of impulse trains, those defined as the function

$$\sum_{k=-\infty}^{\infty} \delta(x-nT)$$

where $\delta(x)$ is the impulse function or Dirac function or Dirac impulse such that

$$\delta(x) = \begin{cases} +\infty & x = 0 \\ 0 & otherwise \end{cases}$$

and

$$\int_{-\infty}^{\infty} \delta(x)dx = 1$$

However, the proofs lack rigour, since the impulse function, and hence the impulse trains, cannot be defined as functions.

In (J.R.Ragazzini and G.F.Franklin, 1958) it is shown the similarity between 1 and the Poisson Summation Formula

$$\sum_{n=-\infty}^{\infty} f(n) = \sum_{k=-\infty}^{\infty} \int_{-\infty}^{\infty} f(s)e^{-2\pi iks}ds$$

Consequently, 1 is often indicated as the Poisson Sampling Formula. In (G.Doetsch, 1971) a rigorous

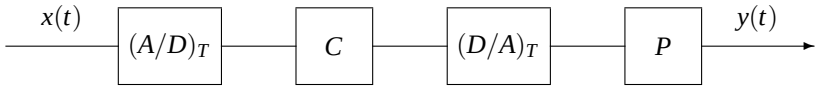


Figure 1: Open Loop Hybrid System.

proof,that avoids the use of the impulse trains, for

$$G_d(e^{st}) = \frac{g(0^+)}{2} + \frac{1}{T}\sum_{k=-\infty}^{\infty} G(s+jk\omega_s)$$

is derived under the assumption that the series $\sum_k G(s+jk\omega_s)$ is uniformly convergent. However, since this condition is a transform domain condition, it is not obvious when a time domain function satisfies it.

In (Braslavsky et al., 1997) it is pointed that for 1 to hold, it is not enough to require that the Laplace transform G of g and its sampled version, G_D, are well defined. It is shown that, for $n_p = 2^{2^{2p}}$ and the continuous function

$$g(t) = sin((2n_p+1)t),\ t \in [p\pi,(p+1)p],\ p \in \mathbb{N}$$

1 does not hold, despite the fact that $G_d(e^{st})$ and its sampled version with period $T = \pi$, are both well defined in the open right-half plane. In fact, it is proved that

$$\lim_{n=\infty}\sum_{k=-n}^{n} G(s+jk\omega_s)$$

does not converges for any $s \geq 0$. Because of the rapid oscillations of g as $t \to \infty$ the class of signals is restricted to functions with bounded and uniform bounded variation.

Definition 1 ((Braslavsky et al., 1997))**.** A function g defined on the closed real interval $[a,b]$ is of bounded variation (BV) when the total variation of g on $[a,b]$,

$$V_g(a,b) = \sup_{a=t_0<t_1<\ldots<t_{n-1}<t_n=b}\sum_{k=1}^{n}|g(t_k)-g(t_{k-1})|$$

is finite. The supremum is taken over every $n \in \mathbb{N}$ and every partition of the interval $[a,b]$ into subintervals $[t_k,T_{k+1}]$ where $k = 0,1,\ldots,n-1$ and $a = t_0 < t_1 < \ldots < t_{n-1} < t_n = b$.

A function g defined on the positive real axis is of uniform bounded variation (UBV) if for some $\Delta > 0$ the total variation $V_g(x,x+\Delta)$ on intervals $[x,x+\Delta]$ of length Δ is uniformly bounded, that is, if

$$\sup_{x\in\mathbb{R}_0^-} V_g(x,x+\Delta) < \infty$$

With the class of signals restricted to UBV functions, a proof for

$$G_d(e^{st}) = \frac{g(0^+)}{2} + \sum_{k=1}^{\infty}\frac{g(kT^+)-g(kT^-)}{2}e^{-skT} + \frac{1}{T}\sum_{k=-\infty}^{\infty} G(s+jl\omega_s)$$

a more general formulation of 1, is provided.

Note that the well posedness of 1 is proved for an open loop context, when the system considered is stable. Despite the fact that it is rather common to analyse a hybrid feedback system with the help of 1, even if the class of signals is restricted to UBV functions, there is no proof of the well posedness of the feedback when applying 1.

The discussion about the consistency of Mathematical Frameworks in Systems Theory that started with the exposure of the Georgiou Smith paradox in (Georgiou and Smith, 1995) made Leithead and al., in (Leithhead and J.O'Reilly, 2003) and (W.E.Leithead et al., 2005), to attempt a Mathematical Framework that expands the class of signals to the class of Distributions (an advantage of a Framework using Distributions is that signals like steps, train pulses and delta functions can be rigorously defined as distributions). Consequently, when dealing with hybrid systems, as the one of Figure 1, in a Distributions Framework, the well posedeness of 1 must be proved again.

However, despite 1 being quoted in Theorem 16.8 of (D.C.Champeney, 1987), no proof could be found in the literature. In this paper a rigorous proof of Theorem 16.8 of (D.C.Champeney, 1987), establishing 1 in a Distributions context, is provided in the Appendix. Furthermore, an application of this result to a open loop hybrid system is provided. In particular, a correct formulation for the D/A and A/D converters in a Distributions context is established.

2 SAMPLING THE TRANSFORMS OF A DISTRIBUTION

The following notations and conventions are adopted.

The value assigned to each $\phi(t) \in D$, the class of good functions with finite support, by the functional $x \in \mathcal{D}$, the class of distributions , is denoted by $x[\phi(t)]$. The symbols for, respectively a regular functional in $\mathcal{D}$ and the ordinary function by which it is defined, e.g. x and $x(t)$, are distinguished by the explicit presence in the latter of the variable. The following subclasses of $\mathcal{D}$ are required.

$\mathcal{D}_B =$ $\{x \in \mathcal{D} : x$ regular with $x(t)$ BV on each finite interval and $|x(t)| \leq c(1+|t|)^N$ for some $c > 0\}; N \geq 0$

$\mathcal{D}_{BN} =$ $\{x \in \mathcal{D} : x$ regular with $x(t)$ BV on each finite interval and $|x(t)| \leq c(1+|t|)^N$ for some $N \geq 0$ and $c > 0\}$

$\mathcal{D}_V =$ $\{x \in \mathcal{D} : x$ regular with $Var_{[a+t,b+t]}\{x(t)\} \leq c(1+|t|)^N$ for each finite interval $[a,b]$ for some $N \geq 0$ and $c > 0\}$

$\mathcal{D}_{VN} =$ $\{x \in \mathcal{D} : x$ regular with $Var_{[a+t,b+t]}\{x(t)\} \leq c(1+|t|)^N$ for each finite interval $[a,b]$ for some $c > 0\}; N \geq 0$

$\mathcal{D}^T =$ $\{x \in \mathcal{D} : x = \sum_{-\infty}^{\infty} a_k \delta_{kT}\}; T > 0$

$\mathcal{D}_B^T =$ $\{x \in \mathcal{D} : x = \sum_{-\infty}^{\infty} a_k \delta_{kT}$ with $|a_k| \leq (1+|k|)^N$ for some $c > 0$ and $N \geq 0\}; T > 0$

$\mathcal{D}_{BN}^T =$ $\{x \in \mathcal{D} : x = \sum_{-\infty}^{\infty} a_k \delta_{kT}$ with $|a_k| \leq (1+|k|)^N$ for some $c > 0\}$; $N \geq 0, T > 0$

where $Var_{[a,b]}\{x(t)\}$ is the variation of $x(t)$ on the interval $[a,b]$ and the functional δ_τ is the delta functional in $\mathcal{D}$ defined by

$$\delta_\tau[\phi(t)] = \phi(\tau)$$

Each functional $x \in \mathcal{D}$ is related by a linear bijections to a functional $\mathcal{U}$ such that

$$x[\phi(t)] = 2\pi X[\Phi(\omega]$$

for all $\phi(t) \in D$ with

$$\Phi(\omega) = \mathcal{F}[\phi(t)](\omega)$$

The functionals x and X constitutes a Fourier transform pair with

$$X = \mathcal{F}\{x\} \text{ and } x = \mathcal{F}^{-1}\{X\}$$

The subclasses $\mathcal{U}_B$, $\mathcal{U}_{BN}$, $\mathcal{U}_V$, $\mathcal{U}_{VN}$, $\mathcal{U}_B^T$ and $\mathcal{U}_{BN}^T$ are the Fourier transforms of the the corresponding subclass of $\mathcal{D}$. The members of $\mathcal{U}^T$ and its subclasses are periodic with period $2\pi/T$.

A multiplier in $\mathcal{D}$ is an ordinary function $f(x)$ that is infinitely differentiable at each real value of x. The multipliers in $\mathcal{D}$ are denoted by $\mathcal{M}$. The subclass $\mathcal{M}^T$ is the class of periodic multipliers with period $2\pi/T$.

The relations between the transform of a distribution and its sampled version is established in the following Theorem.

Theorem 2 (16.8 (D.C.Champeney, 1987))**.** *Suppose $\tilde{f} \in \mathcal{U}$ has a transform $\tilde{F} \in \mathcal{D}$ that is regular and equal to a function F that is of bounded variation on each finite interval (though not necessarily on $(-\infty,\infty)$): then*

(i) $F(y)$ will be equal a.e. to a function $F_D(y)$ such that, at all y,

$$F_D(Y) = \frac{1}{2}[F_D(y^-) + F_D(y^+)]$$

(ii) also

$$X\sum_{-\infty}^{\infty} \tilde{f}(x - nX) \tag{2}$$

will converge in $\mathcal{U}$ to define a periodic functional $\tilde{g}$ whose Fourier coefficients G_n are given by

$$G_n = F_D(n/X),\ n = 0, \pm 1, \pm 2, \ldots$$

(iii) if in addition $\tilde{f} \in \mathcal{D}_S$ and $F(y)/(1+|y|)^N$ is of bounded variation on $(-\infty,\infty)$, then 2 will converge in $\mathcal{D}_S$.

A proof of 2 is given in the Appendix.

3 OPEN LOOP HYBRID FEEDBACK SYSTEM

Reconsider the plants P and C of the open loop hybrid system of Figure 1 as the stable systems on $\mathcal{D}_E$ and $\mathcal{D}_E^T$, respectively.

$$C : x \in \mathcal{D}^T \mapsto y \in \mathcal{D}^T, y = \Psi * x$$

$$P : x \in \mathcal{D} \mapsto y \in \mathcal{D}, y = \Phi * x$$

where Ψ and Φ are convolutes on $\mathcal{D}^T$ and $\mathcal{D}$, respectively. However, since it is required that the idealised sampling of continuous time signal is well-defined, a more appropriate reformulation of continuous time

signals is provided by the subclass of distributions $\mathcal{D}_B$.

Consequently, the convolutes Ψ and Φ corresponding to plants C and P must be restricted to $\mathcal{D}_B^T$ and $\mathcal{D}_B$, respectively. In transform domain the Fourier transforms of signals are represented by functionals in $\mathcal{U}_B$ and the transfer functions of systems are functionals in $\mathcal{M}_B$, the class of multipliers on $\mathcal{U}_B$ mapping $\mathcal{U}_{BN}$ into itself for all $N \geq 0$. It remains to establish a correct formulation of the D/A and A/D converters.

3.1 Frequency Domain Analysis - D/A Converter

Consider an ideal D/A converter which acts, with a time constant T, on a discrete time signal, $\{x[k]\}$ to produce a piecewise constant continuous time signal, $y(t)$; that is, it acts as an ideal zero-order-hold (ZOH). The linear relationship between $\{x[k]\}$ and $y(t)$ in the frequency domain is established by the following Theorem.

Theorem 3. *A discrete time signals $\{x[k]\}$ is acted on by a ZOH, with time constant T, to produce a piecewise constant time signal $y(t)$ such that*

$$y(t) = \sum_{k=-\infty}^{\infty} x[k]h^T(t-k)$$

where $h^T(t) = 1$ when $t \in [0,T)$, zero otherwise. Provided there exists a periodic functional $X \in \mathcal{U}_{BN}^T$ with Fourier coefficients $\{x[k]\}$, then $y(t)$ defines a regular functional, $y \in \mathcal{D}_{BN} \cap \mathcal{D}_{VN}$ such that $Y = H^T X$ where $Y = \mathcal{F}\{y\} \in \mathcal{U}_{BN} \cap \mathcal{U}_{VN}$ and $H^T = \mathcal{F}\{h^T\}$ with h^T the functional in $\mathcal{D}$ defined by $h^T(t)$.

Proof. $y(t)$ is of bounded variation on any finite interval, and, since $X \in \mathcal{U}_{BN}^T$ implies $|x[k]| \leq c(1+|k|)^N$ for some c, $|y(t)| < c^*(1+|t|)^N$ for some c^*. Hence $y = \sum_{k=-\infty}^{\infty} x[k]h_{kT}^T \in \mathcal{D}_{BN}$. Furthermore for all $b_i \in \{-1,1\}$ and $\{\tau_1, \tau_2, ..., \tau_{n+1}\}$ satisfying $a \leq \tau_1 < \tau_2 < ... < \tau_{n+1} \leq b$

$$\begin{aligned}
&\sum_{i=1}^{n} b_i(y(t+\tau_{i+1}) - y(t+\tau_i)) \\
&\quad = \sum_{i=1}^{\bar{n}} b_i(y(t+\tau_{i+1}) - y(t+\tau_i)) \\
&\quad \leq \sum_{i=1}^{\bar{n}} (|y(t+\tau_{i+1})| + |y(t+\tau_i)| \\
&\quad \leq \sum_{i=1}^{\bar{n}} (c^*(1+|t+\tau_{i+1}|) + c^*(|t+\tau_i|) \\
&\quad \leq 2c^*\bar{n}(1+|t+b|)^N
\end{aligned}$$

where $\bar{n} = int(t/(kT))$. Hence, $Var_{[a+t,b+t]}\{y(t)\} \leq \bar{c}(1+|t|)^N$, for some $\bar{c} > 0$, and $y \in \mathcal{D}_{VN}$. In addition, since h^T is a convolute on $\mathcal{D}$,

$$\begin{aligned}
y &= \lim_{n\to\infty} h^T * \sum_{k=-n}^{n} x[n]h_{kT}^T \\
&= \lim_{n\to\infty} * \sum_{k=-n}^{n} x[k]\delta_{kT} \\
&= h^T * \lim_{n\to\infty} \sum_{k=-n}^{n} x[k]\delta_{kT} = h^T * x
\end{aligned}$$

with $x = \mathcal{F}^{-1}\{X\}$ and $Y = H^T X$ as required. ■

Therefore, a D/A converter is represented in the frequency domain by the multiplier H^T mapping $\mathcal{U}_{BN}^T$ into $\mathcal{U}_{BN} \cap \mathcal{U}_{VN}$. Moreover, as a consequence, a discrete time subsystem positioned before a D/A converter is equivalent to a continuous time subsystem positioned after the D/A converter, provided their frequency response functions are the same.

3.2 Frequency Domain Analysis - A/D Converter

Consider an ideal A/D converter which samples, with a sampling interval T, a continuous time signal, $x(t)$, to produce a discrete time signal $\{y[k]\} = \{x[k]\}$. The linear relationship between $x(t)$ and $\{y[k]\}$ in the frequency domain is established by the following Theorem.

Theorem 4. *A continuous time signal, $x(t)$, is acted by a sampler with sampling interval T to produce a discrete time signal $\{y[k]\}$. Provided there exists a regular functional $x \in \mathcal{D}_{BN}$ defined by $x(t)$ then*

(i) $x(t)$ is equal almost everywhere to a function $x_D(t)$ such that, at all t,

$$x_D(t) = \frac{(x_D^-(t) + x_D^+(t))}{2}$$

and so sampling is well defined with $y[k] = x_D(kT)$.

(ii) the summation $\frac{1}{T}\sum_{k=-\infty}^{\infty} X_{2\pi k/T}$ converges in $\mathcal{U}$, where $X = \mathcal{F}\{x\} \in \mathcal{U}_{BN}$, and $\{y[k]\}$ are the Fourier coefficients for a periodic functional $Y \in \mathcal{U}_{BN}^T$ with period $2\pi/T$ such that $Y = o^T[X] = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_{2\pi/T}$

Proof. Since $X \in \mathcal{D}_{BN}$, $x(t)$ is of bounded variation on each finite interval and part (i) follows from Theorem 2. In addition, there exists a periodic functional $Y \in \mathcal{U}$, with period $2\pi/T$ and Fourier coefficients $y_k[k] = x_D(kT)$ such that the summation $\frac{1}{T}\sum_{k=-\infty}^{\infty} X_{2\pi k/T}$ converges in $\mathcal{U}$ and $Y = o^T[X] = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_{2\pi k/T}$. Furthermore, since $x \in \mathcal{D}_{BN}$, $y = \mathcal{F}^{-1}\{Y\} \in \mathcal{D}_{BN}^T$ as required by part (ii). ■

Therefore, an A/D converter is represented in the frequency domain by the linear operator $\mathcal{O}^T$ on $\mathcal{U}_B$ mapping $\mathcal{U}_{BN}$ into $\mathcal{U}_{BN}^T$ for all $N \geq 0$. Further properties of the operator $\mathcal{O}^T$ are established in the following Theorem.

Theorem 5. *If X is a functional in $\mathcal{U}_B$ with n^{th} derivative $X^{(n)}$, Y is a functional in $\mathcal{U}_B$ and M^T is a periodic multiplier in $\mathcal{M}_B$ with period $2\pi/T$ then*

(i) $\mathcal{O}^T[X]$ is a periodic multiplier in $\mathcal{M}_B$ with period $2\pi/T$ provided $j^n X^{(n)} \in \mathcal{U}_{B0}$ for all $n \geq 0$;

(ii)$\mathcal{O}^T[M^T X] = M^T \mathcal{O}^T[X]$;

(iii) $\mathcal{O}^T[Y\mathcal{O}^T[X]] = \mathcal{O}^T[Y]\mathcal{O}^T[X]$ provided $j^n X^{(n)} \in \mathcal{U}_0$ for all $n \geq 0$.

Proof. (i)The regular functional $x = \mathcal{F}^{-1}\{X\} \in \mathcal{D}_B$ is defined by a function $x(t)$, which by Theorem 4 part (i) is equal almost everywhere to a function $x_D(t)$ such that, at all t,

$$x_D(t) = \frac{(x_D^-(t) + x_D^+(t))}{2}$$

For all $n \geq 0$, since $j^n X^{(n)} \in \mathcal{U}_{B0}$, $y \in \mathcal{D}_{B0}$, where y is the functional defined by $t^n x(t)$, and the series $\sum_{k=-\infty}^{\infty}(kT)^n x_D(kT)e^{-jk\omega T}$ converges for all ω. Hence, by Theorem 4 part (ii), $\mathcal{O}^T[X]$ is an infinitely differentiable regular functional. Furthermore, the n^{th} derivative of $\mathcal{O}^T[X]$ is continuous and periodic and so bounded. Consequently, $\mathcal{O}^T[X]$ is a multiplier in $\mathcal{M}_B$ with period $2\pi/T$.

(ii) For any $X \in \mathcal{U}_{BN}$, $M^T X \in \mathcal{U}_{BN}$ and by Theorem 4 both $\mathcal{O}^T[X] \in \mathcal{U}_{BN}^T$ and $\mathcal{O}^T[M^T X] \in \mathcal{U}_{BN}^T$ exist. Moreover, since M^T is a multiplier in $\mathcal{M}_B$ with period $2\pi/T$,

$$\begin{aligned}
&\frac{1}{T}\lim_{n\to\infty}\sum_{k=-n}^{n} M_{kT}^T X_{kT} \\
&\quad = \frac{1}{T}\lim_{n\to\infty}\sum_{k=-n}^{n} M^T X_{kT} = \frac{1}{T}\lim_{n\to\infty} M^T \sum_{k=-n}^{n} X_{kT} \\
&\quad = M^T \frac{1}{T}\lim_{n\to\infty}\sum_{k=-n}^{n} X_{kT}
\end{aligned}$$

and $\mathcal{O}^T[M^T X] = M^T \mathcal{O}^T[X]$ as required.

(iii) It follows directly from part (i) and (ii). ■

A consequence of Theorem 4 part (ii) is that, in frequency domain, a continuous time sub systems positioned before an A/D converter is equivalent to a discrete time subsystem positioned after the A/D provided their frequency response functions are the same.

3.3 The Response of the Open Loop Hybrid Feedback System

In time domain the stable hybrid feedback system of Figure 1 has solution

$$y = \Phi * [(D/A)_T(\Psi * [(A/D)_T x])] \qquad (3)$$

Define K_C^T and K_P the multipliers in $\mathcal{M}_B$, the transfer functions of the convolutes Ψ and Φ, respectively. Therefore, by Theorems 3 4 and 5, in Frequency Domain, to 3 corresponds the solution

$$Y = K_P[H^T(K_C[\mathcal{O}^T X])]$$

where Y and X are functionals in $\mathcal{U}_B$, the Fourier transforms of y and x.

4 CONCLUSION

In this paper the proof of the well posedness of the sampling of a the transform of a distribution is given, establishing the correctness of the Sampling Theorem 16.8 quoted in (D.C.Champeney, 1987). Moreover, the result is applied to the frequency domain response of an open loop hybrid system, through the correct formulation for the D/A and A/D converters.

REFERENCES

B.C.Kuo (1992). Digital control systems. *Saunders College Pubblishing*.

Braslavsky, J., Meinsma, G., Middleton, R., and Freudenberg, J. (1997). On a key sampling formula relating the laplace and z transforms. *Systems Control Lett.*, 29(4):181–190.

D.C.Champeney (1987). A handbook of fourier theorems. *CAmbridge, CUP*.

E.I.Jury (1958). Sampled-data control systems. *Wiley, New York*.

G.Doetsch (1971). Guide to the applications of laplace and z transforms. *D. van Nostrand, Princeton, NJ*.

Georgiou, T. T. and Smith, M. C. (1995). Intrinsic difficulties in using the doubly-infinite time axis for input-output control theory. *IEEE Trans. Automat. Control*, 40(3):516–518.

G.F.Franklin, J. and M.L.Workman (1990). Digital control of dynamic systems. *Addison-Wesley*.

J.R.Ragazzini and G.F.Franklin (1958). Sampled-data control systems. *McGraw-Hill, New York*.

J.S.Freudenberg, R. and J.H.Braslavsky (1995). Inherent design limitations for linear sampled-data feedback systems. *Internat. J. Control*, 61:1387–1421.

K.J.Astrom and B.Wittenmark (1990). Computer-controlled systems: Theory and design. *Prentice-Hall*.

K.Ogata (1987). Discrete-time control systems. *Prentice-Hall*.

Leithhead, W. E. and J.O'Reilly (2003). A consistent time-domain and frequency-domain representation for discrete-time linear time-invariant feedback systems. In *Proceedings of the American Control Conference*, pages 429–434, Denver, Colorado.

Leung, G., Perry, T., and Francis, B. (1991). Performance analysis of sampled-data control systems. *Automatica J. IFAC*, 27(4):699–704.

M.Araki, Y. and T.Hagiwara (1996). Frequency respponse of sampled-data systems. *Automatica*, 32:482–497.

T.Chen and B.A.Francis (1995). Optimal sampled-data control systems. *Springer*.

T.Hagiwara, Y. and M.Araki (1995). Computation of the frequency response gains and h_∞-norm of a sampled-data system. *Systems Control Lett.*, 25:281–288.

W.E.Leithead, E.Ragnoli, and J.O'Reilly (2005). Open-loop unstable feedback systems with doubly-sided inputs: an explicit demonstration of self-consistency.

Yamamoto, Y. and Araki, M. (1994). Frequency responses for sampled-data systems—their equivalence and relationships. *Linear Algebra Appl.*, 205/206:1319–1339.

Y.N.Rosenvasser (1995a). Mathematical description and analysis of multivariable sampled-data systems in continuous-time: Part i. *Autom. Remote Control*, 56(4):526–540.

Y.N.Rosenvasser (1995b). Mathematical description and analysis of multivariable sampled-data systems in continuous-time: Part ii. *Autom. Remote Control*, 56(4):684–697.

APPENDIX

Theorem 2 (D.C.Champeney, 1987)

Proof. (i) and (ii) Let $\tilde{f}_N \in \mathcal{D}$ be the regular functional defined by $f_N(x)$ where

$$f_N(x) = \sum_{n=-N}^{N} e^{jn(2\pi/X)x} = \frac{sin(\pi(2N+1)x/(2X))}{sin(\pi x/(2X))}$$

$\tilde{f}_N$ is a multiplier on $\mathcal{D}$ and $f_N(x)$ is periodic with period X such that

$$\int_{-X/2}^{X/2} f_N(x)dx = X$$

For any regular $\tilde{g} \in \mathcal{D}$, with $g(x)$ of bounded variation on any finite interval, and any $\psi(x) \in D$,

$$(\tilde{f}_N\tilde{g})[\psi(x)] = \tilde{g}[f_N(x)\psi(x)] = \int_{-\infty}^{\infty} g(x)\, f_N(x)\Psi(x)dx$$

Since $\psi(x)$ is of finite support, $\exists K$ such that $\psi(x) = 0$ for $|x| > (K+\frac{1}{2})X$. Hence,

$$\begin{aligned}\tilde{f}_N\tilde{g}[\psi[x]] &= \int_{-(K+1/2)X}^{(K+1/2)X} g(x)\, f_N(x)\psi(x)dx \\ &= \int_{-X/2}^{X/2} \left\{ \sum_{k=-K}^{K} f_N(x)g(x+kX)\psi(x+kX) \right\} dx \\ &= \int_{-X/2}^{X/2} f_N(x)\phi_K(x)dx \\ &= \int_{-X/2}^{X/2} \left(\frac{sin\left(\frac{\pi(2N+1)x}{2X}\right)}{x} \right) \left\{ \frac{\phi_k(x)x}{sin\left(\frac{\pi x}{2X}\right)} \right\} dx\end{aligned}$$

where

$$\phi_k(x) = \sum_{k=-K}^{K} g(x+kX)\psi(x+kX)$$

For all k, $g(x)$ is of finite variation on $[(k-1/2)X,(k+1/2)X]$ and so $\phi_K(x)x/(sin(\pi x/(2X)))$ is of finite variation on $[(k-1/2)X,(k+1/2)X]$. Consequently, by Theorem 5.10 of (D.C.Champeney, 1987), $x = 0$ is a Dirichlet point and

$$\lim_{N\to\infty} \int_{-X/2}^{X/2} (sin(\pi(2N+1)x/(2X))/x) \{\phi_k(x)x/sin(\pi x/(2X))\}dx = X(\phi_k(0^+) + \phi_k(0^-))/2$$

It follows that

$$\begin{aligned}&\lim_{N\to\infty} (\tilde{f}_N\tilde{g})[\psi(x)] \\ &\quad = X \sum_{k=-K}^{K} \frac{1}{2}(g(kX^-) + g(kX^+))\psi(kX) \\ &\quad = X \sum_{k=-K}^{K} \frac{1}{2}(g(kX^-) + g(kX^+))\tilde{\delta}_{kX}[\psi(x)]\end{aligned}$$

Hence, $\frac{1}{N}\tilde{f}_N\tilde{g}$ converges to

$$\tilde{h} = \sum_{k=-K}^{K} \frac{1}{2}(g(kX^-) + g(kX^+))\tilde{\delta}_{kX}$$

in $\mathcal{D}$. Furthermore,

$$\tilde{H} = \mathcal{F}\{\tilde{h}\} = \sum_{k=-\infty}^{\infty} \frac{1}{2}(g(kX^-) + g(kX^+))\tilde{e}_{k(2\pi/X)} \in \mathcal{U}$$

and by Theorem 16.3 of (D.C.Champeney, 1987), $\tilde{H}$ is periodic with period $2\pi/X$ and Fourier coefficients $\left\{\frac{1}{2}(g(kX^-) + g(kX^+))\right\}$. However

$$\begin{aligned}\mathcal{F}\left\{\frac{1}{X}\tilde{f}_N\tilde{g}\right\} &= \frac{1}{X}\mathcal{F}\{\tilde{f}_N\} * \mathcal{F}\{\tilde{g}\} \\ &= \frac{1}{X}\left(\sum_{n=-N}^{N} \tilde{\delta}_{n(2\pi/X)}\right) * \tilde{G} = \frac{1}{N}\sum_{n=-N}^{N} \tilde{G}_{n(2\pi/X)}\end{aligned}$$

It immediately follows that $\frac{1}{X}\sum_{n=-\infty}^{\infty}\tilde{G}_{n(2\pi/X)} \in \mathcal{U}$ and is equal to $\tilde{H}$. Thus part (i) part and (ii) are established.

(iii) Let f_N as above. For any function $g(x)$, with $g(x)/(1+|x|)^M$ of bounded variation on $(-\infty,\infty)$ for some $M>0$, and any $\psi(x) \in S$

$$|g(x)\psi(x)| < c/(1+|x|)^2$$

for some $c>0$. Hence,

$$\begin{aligned}
&\int_{-\infty}^{\infty} g(x) f_N(x)\psi(x)dx \\
&\quad = \lim_{K\to\infty}\left\{\int_{-(K+1/2)X}^{(K+1/2)X} f_N(x)g(x)\psi(x)dx\right\} \\
&\quad = \lim_{K\to\infty}\int_{-X/2}^{X/2} f_N(x) \\
&\quad \left\{\sum_{k=-K}^{K} g(x+kX)\psi(x+kX)\right\}dx
\end{aligned}$$

In addition, for any x,

$$|g(x+kX)\psi(x+kX)| < c/(1+|kX|)^2$$

for some $c>0$ and the series

$$\phi_K(x) = \sum_{k=-K}^{K} g(x+kX)\psi(x+kX)$$

is absolutely convergent. Hence, there exists a function, $\phi(x)$, such that $\phi_K(x)$ converges pointwise to $\phi(x)$ and there exists a constant, A, such that, for all $K>0$, $|\phi_K(x)|<A$, $\forall x \in [-X/2, X/2]$. Consequently, by Theorem 4.1 of (D.C.Champeney, 1987),

$$\begin{aligned}
&\lim_{K\to\infty}\int_{-X/2}^{X/2} f_N(x)\left\{\sum_{k=-K}^{K} g(x+kX)\psi(x+kX)\right\}dx \\
&\quad = \int_{-X/2}^{X/2} f_N(x)\phi(x)dx \\
&\quad = \int_{-X/2}^{X/2}\left(\frac{sin\left(\frac{\pi(2N+1)x}{2X}\right)}{x}\right)\left\{\frac{\phi(x)}{\left(\frac{sin\left(\frac{\pi x}{2X}\right)}{x}\right)}\right\}dx
\end{aligned}$$

Furthermore, $\phi(x)x/(sin(\pi x/(2X))$ is of bounded variation on $[-X/2, X/2]$. By Theorem 5.10 of (D.C.Champeney, 1987), $x=0$ is a Dirichlet point and

$$\begin{aligned}
&\lim_{N\to\infty}\int_{-X/2}^{X/2}\left(\frac{sin\left(\frac{\pi(2N+1)x}{2X}\right)}{x}\right) \\
&\quad \left\{\frac{\phi_k(x)x}{sin\left(\frac{\pi x}{2X}\right)}\right\}dx = X(\phi_k(0^+)+\phi_k(0^-))/2
\end{aligned}$$

Since, for $|x| < X/2$,

$$|g(kX+x)\psi(kX+x)| < c/(1+|kX|)^2$$

for some $c>0$

$$\phi(0^+) = \sum_{k=-\infty}^{\infty} g(kX^+)\psi(kX^+)$$

and

$$\phi(0^-) = \sum_{k=-\infty}^{\infty} g(kX^-)\psi(kX^-)$$

and it follows that

$$\begin{aligned}
&\lim_{N\to\infty}\int_{-\infty}^{\infty} f_N(x)g(x)\psi(x)dx \\
&\quad = \frac{1}{2}X\sum_{k=-\infty}^{\infty}((g(kX^+)\psi(kX^+)) \\
&\quad +(g(kX^-)\psi(kX^-))) \\
&\quad = \frac{1}{2}X\sum_{k=-\infty}^{\infty}(g(kX^+)+g(kX^-)\psi(kX^-))\psi(kX)
\end{aligned}$$

Let $\tilde{f}_N \in \mathcal{D}_S$ be the regular functional defined by $f_N(x)$ then $\tilde{f}_N$ is a multiplier on $\mathcal{D}_S$. For the regular functional $\tilde{g} \in \mathcal{D}_S$ defined by $g(x)$

$$(\tilde{f}_N\tilde{g})[\psi(x)] = \tilde{g}[f_N(x)\psi(x)] = \int_{-\infty}^{\infty} g(x) f_N(x)\psi(x)dx$$

From the foregoing, it follows that

$$\begin{aligned}
&\lim_{N\to\infty}(\tilde{f}_N\tilde{g})[\psi(x)] \\
&\quad = X\sum_{k=-\infty}^{\infty}\frac{1}{2}(g(kX^-)+g(kX^+))\psi(kX) \\
&\quad = X\sum_{k=-\infty}^{\infty}\frac{1}{2}(g(kX^-)+g(kX^+))\tilde{\delta}_{kX}[\psi(x)]
\end{aligned}$$

Hence, $\frac{1}{X}\tilde{f}_N\tilde{g}$ converges to

$$\tilde{h} = \sum_{k=-\infty}^{\infty}\frac{1}{2}(g(kX^-)+g(kX^+))\tilde{\delta}_{kX}$$

in $\mathcal{D}_S$. Furthermore,

$$\tilde{H} = \mathcal{F}\{\tilde{h}\} = \sum_{k=-\infty}^{\infty}\frac{1}{2}(g(kX^-)+g(kX^+))\tilde{e}_{k(2\pi/X)} \in \mathcal{S}$$

and by Theorem 16.3 of (D.C.Champeney, 1987), $\tilde{H}$ is periodic with period $2\pi/X$ and Fourier coefficients $\left\{\frac{1}{2}(g(kX^-)+g(kX^+))\right\}$. However

$$\begin{aligned}
&\mathcal{F}\left\{\frac{1}{X}\tilde{f}_N\tilde{g}\right\} = \frac{1}{X}\mathcal{F}\{\tilde{f}_N\} * \mathcal{F}\{\tilde{g}\} \\
&\quad = \frac{1}{X}\left(\sum_{n=-N}^{N}\tilde{\delta}_{n(2\pi/X)}\right) * \tilde{G} = \frac{1}{N}\sum_{n=-N}^{N}\tilde{G}_{n(2\pi/X)}
\end{aligned}$$

It immediately follows that $\frac{1}{X}\sum_{n=-\infty}^{\infty}\tilde{G}_{n(2\pi/X)} \in \mathcal{D}_S$ and is equal to $\tilde{H}$. Thus part (iii) is established. ■

DECENTRALIZED APPROACH FOR FAULT DIAGNOSIS OF DISCRETE EVENT SYSTEMS

Moamar Sayed Mouchaweh[a], Alexandre Philippot[b] and Véronique Carré-Ménétrier[a]
[a]*Université de Reims, CReSTIC, Moulin de la Housse 51687 Reims - France*
[b]*LURPA, ENS de Cachan, 61 avenue du Président Wilson, 94235 Cachan Cedex, France*
{moamar.sayed-mouchaweh alexandre.philippot, veronique carre} @univ-reims.fr

Keywords: Fault diagnosis, Discrete Event Systems, Decentralized diagnosis, Co-diagnosability notion.

Abstract: This paper proposes a decentralized approach to realize the diagnosis of Discrete Event Systems (DES). This approach is based on a set of local diagnosers, each one of them diagnoses faults entailing the violation of the local desired behavior. These local diagnosers infer the fault's occurrence using event sequences, time delays between correlated events and state conditions, characterized by sensors readings and commands issued by the controller. An adapted codiagnosability notion is formally defined in order to ensure that the set of local diagnosers is able to diagnose all faults entailing the violation of the global desired behavior. An example is used to illustrate the proposed approach.

1 INTRODUCTION

Manufacturing systems are too large to perform a centralized diagnosis. Moreover, they are informationally and geographically decentralized. Thus a diagnosis module with a decentralized structure is the most adapted one for this kind of systems. However, the challenge of decentralized diagnosis methods is to perform local diagnosis equivalent to the centralized one. Indeed, the partial observation of the system may lead to an ambiguity of the final diagnosis decision. Examples of DES decentralized diagnosis methods can be found in (Debouk, 2000), (Pandalai, 2000), (Qiu, 2005), and the references therein.

Failure diagnosis in DES requires that once a failure is occurred, it must be detected and isolated within a bounded delay or number of events. This property is verified using a notion of diagnosability. This notion can be formalized differently according to whether the fault is modelled as the execution of certain faulty events, event-based notion, or as the consequence of reaching at certain faulty states, state-based notion. In (Sampath, 1994), an event-based diagnosability notion is defined. The system model is based on a finite-state automaton. This notion defines a diagnoser that uses the history of events to detect the occurrence of a failure. Consequently, a system is diagnosable if and only if any pair of faulty/non-faulty behaviors can be distinguished by their projections to observable behaviors. The event-based diagnoser can diagnose actuator and sensor permanent and intermittent failures. However, the diagnoser and the system model must be initiated at the same time to allow the system model and diagnoser to response simultaneously to events. This initialization is hard to obtain in manufacturing systems since their initial state may not be known. To enhance the diagnosability, the above framework is extended to dense-time automata (Tripakis, 2002). This extension is useful since it permits to model plants with timed behavior.

In (Pandalai, 2000), an event-based approach is proposed to monitor manufacturing systems. In this approach, the timed sequence events, generated by the DES, is compared with a set of specifications of normal functioning called templates. These templates are based on the notion of expected event sequencing and timing relationships. They are suitable for modelling processes in which both single-instance and multiple-instance behaviors are exhibited concurrently. However, these templates do not allow the analysis of diagnosability properties, which are based on a diagnosability notion.

To find a remedy to the initialization problem, a state-based diagnosability notion is proposed in (Lin, 1994), (Zad, 2003). In this notion, since the system states describe the conditions of its components,

diagnosing a fault can be seen as the identification in which state or set of states the system belongs to. However, the diagnosis is limited to the case of actuator faults. While manufacturing systems use many sensors entailing the necessity of diagnosing also their faults.

This paper presents a decentralized diagnosis approach to perform the diagnosis of manufacturing systems. The paper is structured as follows. Firstly, the different steps of the proposed approach necessary to construct the local diagnosers are detailed. Secondly, a timed-event-based diagnosability notion is presented. Then, in order to verify the codiagnosability property of local diagnosers, this notion is extended to the codiagnosability notion. Finally, a simple example is used to illustrate the proposed approach.

2 DECENTRALIZED DIAGNOSIS APPROACH

2.1 System Boolean Models

We use Boolean DES (BDES) modelling, introduced in (Wang, 2000), to model the equipments (sensors and actuators) behavior of the system. The system model G consists of n local models: $G_1,\ldots, G_n$, each one owns its local observable events responsible of a restricted area of the process. $G^i = (\Sigma^i, Q^i, Y^i, \delta^i, h^i, q_0{}^i)$ is represented as Moore automaton and $L = L(G)$ denotes its corresponding prefixed closed language. Σ is a set of finite observable and unobservable events. Q is the set of states, Y is the output space, δ: Σ^* x $Q \rightarrow Q$ is the state transition function and Σ^* is the set of all event sequences of the language $L(G)$. $\delta(\sigma, q)$ provides the set of possible next states if σ occurs at q. h: $Q \rightarrow Y$ is the output function and $h(q)$ is the observed output at q. q_0 is the initial state.

Let $\Sigma_{\Pi} = \{\Pi_{F1}, \Pi_{F2},\ldots, \Pi_{Fr}\}$ be the set of fault partitions. Each fault partition, Π_{Fj}, $j \in \{1, 2,\ldots, r\}$, corresponds to some kind of faults in an equipment element (sensor or actuator). We assume at most one fault may occur at a time. These faults must be considered when BDES models.

In (Balemi, 1993), Balemi *et al.* defined controllable events $\Sigma_c \subseteq \Sigma$ as controller's outputs sent to actuators, and uncontrollable events $\Sigma_u \subseteq \Sigma$ as the controller's inputs coming from sensors. ($\Sigma_o = \Sigma_c \cup \Sigma_u$) $\subset \Sigma$ is the set of observable events. The unobservable events are failure events or other events which cause changes not recorded by sensors.

Let G^i and its corresponding prefixed closed language, $L^i = L(G^i)$, be the local model of the restricted area of the system observed by this model. $G^i = (\Sigma^i, Q^i, Y^i, \delta^i, h^i, q_0{}^i)$ is represented as Moore automaton. $\Sigma_0{}^i = \Sigma_c{}^i \cup \Sigma_u{}^i$ is the set of local observable events by G^i and $\Sigma_0{}^i \subset \Sigma_0$. The other notations have the usual definition but for the restricted area observed by G^i.

G observes the system by one global projection function or mask, P_L: $\Sigma^* \cup \{\varepsilon\} \rightarrow \Sigma_0{}^*$, where $\Sigma_0{}^*$ is the set of all observable event sequences observed by G. The inverse projection function is defined as: $P_L^{-1}(u) = \{s \in L: P_L(s) = u\}$. Similarly, a local projection function can be defined for each local model G^i as: P^i: $\Sigma^{i*} \cup \{\varepsilon\} \rightarrow \Sigma_0{}^{i*}$.

Each state q_j of G is represented by an output vector h_j considered as a Boolean vector whose components are Boolean variables. Let d denote the number of state variables of G, the output vector h_j of each state q_j can be defined as:

$$\forall q_j \in Q, h(q_j) = h_j = (h_{j1},\ldots, h_{jp},\ldots, h_{jd}), h_{jp} \in \{0, 1\},$$
$$1 \le j \le 2^d, h_j \in Y \subseteq IB^d$$

A transition from one state to another is defined as a change of a state variable from 0 to 1, or from 1 to 0. Thus each transition produces an event α characterized by either rising, $\alpha = \uparrow h_{jp}$, or falling, $\alpha = \downarrow h_{jp}$, edges where $p \in \{1, 2,\ldots, d\}$.

To describe the effect of the occurrence of an event $\alpha \in \Sigma_0$, a displacement vector $E_\alpha = (e_{\alpha 1},\ldots, e_{\alpha p},\ldots, e_{\alpha d})$ is used. If $e_{\alpha p} = 1$, then the value of p^{th} state variable h_{jp} will be set or reset when α occurs. While if $e_{\alpha p} = 0$, the value of p^{th} state variable h_{jp} will remain unchanged:

$$\forall q_i, q_j \in Q, \forall \alpha \in \Sigma_o, q_j = \delta(\alpha, q_i) \Rightarrow h_j = h_i \oplus E_\alpha \quad (1)$$

The set of all the displacement vectors of all the events provides the displacement matrix E. For each event $\alpha \in \Sigma_0$, an enablement condition, $en_\alpha(q_i) \in \{0, 1\}$, is defined in order to indicate if the event α can occur at the state q_i, $en_\alpha(q_i) = 1$, or not:

$$\forall q_i, q_j \in Q, \forall \alpha \in \Sigma_o, q_j = \delta(\alpha, q_i) \Rightarrow h_j = h_i \oplus (E_\alpha . en_\alpha(q_i)) \quad (2)$$

2.2 Constrained-System Boolean Model

Let $S = (\Sigma, Q_S, Y, \delta_S, h, q_0)$ denote the constrained-system model, characterized as Moore automaton. It defines the global desired behavior of the system and it is represented by the prefixed closed specification language $K = L(S) \subseteq L(G)$. S can be obtained using different algorithms from the literature as the ones developed in (Philippot, 2005),

(Ramadge, 1987) and the references therein. To obtain the transition function δ_S, the enablement conditions for all the system events at each state must satisfy all the specifications K, representing the desired behavior:

$$\forall \alpha \in \Sigma_0, \forall q_i, q_j \in Q_S, q_j = \delta_S(\alpha, q_i) \Rightarrow \\ en_\alpha(q_i) = 1, h_j = h_i \oplus (E_\alpha . en_\alpha(q_i)) \quad (3)$$

Each local model G^i has a local constrained model S^i, which is a part of the global constrained model S. S^i is represented by the specification language $K^i = L(S^i)$, which is included in K. S^i is Moore automaton: $S^i = (\Sigma^i, Q^i_S, Y^i, \delta^i_S, h^i, q^i_0)$ and $Q^i_S \subset Q^i$. All these notations have the usual definition but for the local constrained-system model S^i.

2.3 Codiagnosability Notion

2.3.1 Basic Definitions

Let Ψ_{Fj} define the set of all the event sequences ending by a fault belonging to the fault partition Π_{Fj}. Thus $\Psi_F = \cup^r_{j=1}(\Psi_{F_j})$ denotes the set of all the event sequences ending by a fault belonging to one of fault partitions of Σ_Π. Consequently $\Psi_F \subseteq (L - K)$, i.e., all the faulty sequences are considered as violation of the specification language K. The set of faulty states is defined as $S_F : \cup^r_{j=1}(S_{F_j})$ where S_{Fj} is the set of states reached by the occurrence of a fault of F_j. Let H_{Fj} denote the set of all state output vectors of the faulty states belonging to S_{Fj}. Then the output partition H_{Fj} can be defined as:

$$\forall q' \in S_{Fj}, h' = h(q') \Rightarrow h' \in H_{Fj}.$$

The set of fault labels $\Lambda_F = \{F_1, F_2, ..., F_r\}$ indicates the occurrence of a fault belonging to one of the fault partitions Σ_Π. By adding the normal label N, we can obtain the set Λ of all the labels used by the diagnoser. We define the label function l: $Q \rightarrow \Delta$ to indicate the functional status of the system when it reaches a state $q \in Q$. Δ is the set of all possible subsets of the diagnoser labels:

$$\Delta = \left\{ \begin{matrix} \{N\}, \{F_1\}, \{F_2\}, ..., \{F_r\}, \{F_1, F_2\}, \{F_1, F_2, ..., F_r\}, \\ \{N, F_1\}, ..., \{N, F_r\}, \{N, F_1, F_2\}, ..., \{N, F_1, ..., F_r\} \end{matrix} \right\}.$$

Similarly, we can define Δ_F as the set of all the subsets of fault labels.

2.3.2 Events Timing Delays Modelling

The majority of sensors and actuators in manufacturing systems produce constrained events since state's changes are usually effected by a predictable flow of materials (Pandalai, 2000). Therefore, we define a set of expected consequents EC_β for each controllable event, $\beta \in \Sigma_c$, in order to predict uncontrollable but observable consequent events within pre-defined time periods. This EC_β describes the next events that should occur and the relative time periods in which they are expected.

These pre-defined time periods are determined by experts according to the system dynamic and to the desired behavior. If $u = \beta\alpha_1\alpha_2...\alpha_k$ is an observable event sequence starting by a controllable event β, and ending by the observable event sequence $\alpha_1\alpha_2...\alpha_k \subset \Sigma^*_{uo}$, then the set of expected consequents $EC_\beta(u)$ is created when the event β occurs. $EC_\beta(u)$ has the following form: $EC_\beta(u) = \{C^\beta_{\alpha_1}, C^\beta_{\alpha_2}, ..., C^\beta_{\alpha_i}, ..., C^\beta_{\alpha_k}\}$. $C^\beta_{\alpha i}$ is a consequent expected after the enablement of the controllable event β and it is defined as follows: $C^\beta_{\alpha i} = \{\alpha_j, \alpha_i, (q_{\alpha i}, [t^{\alpha i}_{\min}, t^{\alpha i}_{\max}], l^{\alpha i}_{q\alpha i})\}$. It means that when α_j occurs, the event α_i should happen at the state q_{α_i} and within the interval $[t^{\alpha_i}_{\min}, t^{\alpha_i}_{max}]$. If it is the case then the expected consequent is satisfied. If the event α_i has occurred before $t^{\alpha_i}_{min}$ or after $t^{\alpha_i}_{max}$ then the expected consequent is not satisfied and it provides the fault label $l^{\alpha_i}_{q\alpha i} \in \Delta_F$, as the cause of this non-satisfaction. This set of expected consequent $EC_\beta(u)$ is evaluated by a function $EF_\beta(u)$. $EF_\beta(u)$ is equal to 1 if one of its expected consequents is not satisfied while it is equal to zero if all its expected consequents are satisfied.

2.3.3 Codiagnosability Notion Formulation

If a system composed of n local diagnosers with a global closed prefixed language L, a global closed prefixed specification language K, a global projection function P, and a predefined set of fault partitions, $\Sigma_\Pi = \{\Pi_{F1}, \Pi_{F2}, ..., \Pi_{Fr}\}$, is diagnosable using a central diagnoser. Then this system is F-codiagnosable according to the projection functions, $P^i : i = 1 \ldots n$, if and only if :

$$\left. \begin{matrix} \exists k \in IN, \forall f \in \Pi_{F_j}, j \in \{1,2,...,r\}, \forall st \in (L-K) \cap \Psi_{F_j}, \\ \exists i \in \{1,2,...,n\}, |t| \geq k, \forall u \in P^{i^{-1}} P^i(st) \cap (L-K) \end{matrix} \right] \\ \Rightarrow u \in (L-K) \cap \Psi_{F_j} \quad (4)$$

$$\forall q \in Q, q' = \delta^i(u,q), h' = h(q') \Rightarrow h' \in H_{F_j}$$

$$\exists z \in \{1,2,...,m\} \Rightarrow EF_z(P^i(st)) = 1 \text{ and } l_z = \{F_j\}$$

The satisfaction of (4) means that the occurrence of a fault of the type F_j is diagnosable by at least one local diagnoser D^i, using the event-based, state-based or timed local models. Indeed if the faulty event sequence s, ending by a fault of the type F_j, is

distinguishable by the central diagnoser D after the execution of $k = |t|$ transitions, where t is a continuation of s. If u is any other event sequence belonging to $(L - K)$ and producing the same observable event sequence as st, $P^i(u) = P^i(st)$, according to the local diagnoser D^i. Then the system is F-codiagnosable if and only if:

- u contains in it a fault of the type F_j, (event-based model),
- u transits D^i to a state characterized by an output vector belonging to the output partition H_{Fj}, (state-based model),
- There is at least one expected consequent, defining a temporal constraint between the occurrence of the observable events $P^i(st)$ by the diagnoser D^i, not satisfied. This expected consequent is evaluated by an expected function which provides a fault label $l = \{F_j\}$ as the cause of this non-satisfaction, (timed-model).

2.3.4 Codiagnosability Checking

The set of local diagnosers are able to diagnose any fault belonging to one of the fault partitions of F and within a finite delay, if:

$$\forall \rho \in L, \rho \in K, \forall i \in \{1,2,...,n\}, \exists q \in Q \Rightarrow en^i_\rho(q) = 1 \quad (5)$$

$$\forall \rho \in L, \rho \in L-K, \forall j \in \{1,2,...,r\}, \rho \cap \psi_{F_j} \neq \varphi, \exists i \in \{1,2,...,n\}, \exists q \in Q \Rightarrow (en^i_\rho(q) = 0 \text{ or } EF_q(P^i(\rho)) = 1) \text{ and } l_q = \{F_j\} \quad (6)$$

$$|\rho| \leq k \in N \quad (7)$$

(5) means that all the enablement conditions of all the local diagnosers must be satisfied for any event of a sequence belonging to the global desired behavior. Thus this condition ensures that no conflict can occur between local diagnosers for the enablement of events at any state of the desired behavior. The satisfaction of (6) ensures that any event sequence violating the global desired behavior, due to the occurrence of a fault of the type F_j, must be diagnosed by at least one local diagnoser D^i when it reaches the state q. This detection and isolation are based on the non-satisfaction either of the enablement condition of the latest event in the event sequence ρ or of its expected function. In the both cases, this non-satisfaction should provide the fault label F_j. Finally (7) guarantees that this diagnosis decision will be realized in a finite delay equal to the cardinality of the event sequence ρ .

3 ILLUSTRATION EXAMPLE

3.1 Example Presentation

We monitor a wagon with an electric actuator with two senses of movement: right and left, obtained by two commands, R for the movement right and L for the movement left. Three sensors a, b and c are used to indicate the wagon location in, respectively, A, B or C, as it is illustrated in Figure 1. We have chosen this simple example for easy understanding. The same reasoning can be followed for the application of the approach on more complex examples.

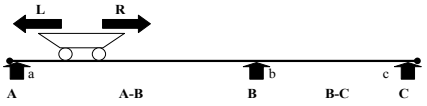


Figure 1: Illustration example.

The following hypotheses must hold:

- The wagon inertia is null,
- Actuator does not fail during operation, i.e., if it does fail, the fault is at the start of operation,
- There are no ambiguity or indecision cases between the local diagnosers.

The system is modelled with two sub models: G^1 and G^2. Their local observable events are respectively: $\Sigma_0{}^1 = \{\uparrow R, \downarrow R, \uparrow L, \downarrow L, \uparrow a, \downarrow a, \uparrow b, \downarrow b\}$ and $\Sigma_0{}^2 = \{\uparrow R, \downarrow R, \uparrow L, \downarrow L, \uparrow b, \downarrow b, \uparrow c, \downarrow c\}$. We use five Boolean state variables a, b, c, R and L to describe the overall wagon behavior G. a, b and c are true when the wagon is located respectively in A, B or C.

Each local model consists of two components: the wagon motor behavior and the change of the wagon location measured by the sensors a and b for G^1, and b and c for G^2. The set of fault partitions to be diagnosed is F = $\{F_1, F_2, F_3, F_4\}$. F_1, F_2, F_3 and F_4 indicate, respectively, sensor a, sensor b, sensor c and wagon motor stuck-on or stuck-off.

3.2 Constrained System Models

The constrained-system model S for the wagon example is depicted in Figure 2 and is provided by the user. S^1 and S^2 represent the local desired behaviors for the two sub models G^1 and G^2 according to their set of local observable events.

In BDES modelling, this desired behavior can be described using two tables; the first one explains the enablement conditions for the occurrence of each event and the second one is the displacement matrix for the estimation of the state output vector of each

next state. These tables are shown respectively in Table 1, Table 2 and Table 3 for S^1 and S^2.

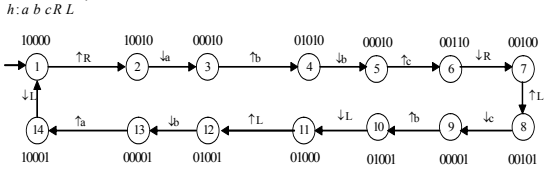


Figure 2: Global constrained-system model S .

Table 1: The enablement conditions for S^1 and S^2.

σ: S^1	en_σ	σ: S^2	en_σ
$\uparrow a$	$\bar{a}.\bar{b}.\bar{R}.L$	$\uparrow b$	$\bar{b}\bar{c}.R\bar{L}+\bar{b}\bar{c}.RL$
$\downarrow a$	$a.\bar{b}.R.\bar{L}$	$\downarrow b$	$b\bar{c}.R\bar{L}+b\bar{c}.RL$
$\uparrow b$	$\bar{a}\bar{b}.R\bar{L}+\bar{a}\bar{b}.RL$	$\uparrow c$	$\bar{b}.\bar{c}.R.\bar{L}$
$\downarrow b$	$\bar{a}b.R\bar{L}+\bar{a}b.RL$	$\downarrow c$	$\bar{b}.c.R.L$
$\uparrow R$	$a.\bar{b}.\bar{R}.\bar{L}$	$\uparrow R$	$\bar{b}\bar{c}.\bar{R}.\bar{L}$
$\downarrow R$	$\bar{a}.\bar{b}.R.\bar{L}$	$\downarrow R$	$\bar{b}.c.R.\bar{L}$
$\uparrow L$	$\bar{a}\bar{b}.\bar{R}\bar{L}+\bar{a}b.\bar{R}\bar{L}$	$\uparrow L$	$\bar{b}.c.\bar{R}.\bar{L}+b\bar{c}.\bar{R}.\bar{L}$
$\downarrow L$	$\bar{a}b.\bar{R}L+a\bar{b}.\bar{R}L$	$\downarrow L$	$b\bar{c}.\bar{R}L+\bar{b}\bar{c}.\bar{R}L$

Table 2: The displacement matrix E^1 for S^1.

State variable	$\uparrow a$	$\downarrow a$	$\uparrow b$	$\downarrow b$	$\uparrow R$	$\downarrow R$	$\uparrow L$	$\downarrow L$
a	1	1	0	0	0	0	0	0
b	0	0	1	1	0	0	0	0
R	0	0	0	0	1	1	0	0
L	0	0	0	0	0	0	1	1

Table 3: The displacement matrix E^2 for S^2.

State variable	$\uparrow b$	$\downarrow b$	$\uparrow c$	$\downarrow c$	$\uparrow R$	$\downarrow R$	$\uparrow L$	$\downarrow L$
b	1	1	0	0	0	0	0	0
c	0	0	1	1	0	0	0	0
R	0	0	0	0	1	1	0	0
L	0	0	0	0	0	0	1	1

3.3 Expected Consequents Definition

Two expected consequents are defined for G, one for each command enablement: $EC_{\uparrow R}$, $EC_{\uparrow L}$. The enablement of R, entails the events $\downarrow a$, $\uparrow b$, $\downarrow b$, and $\uparrow c$ to occur respectively at the states q_2, q_3, q_4, and q_5. $\downarrow a$ is expected to occur within the time period [1,2], after the enablement of R, $\uparrow b$ within the time period [3,5] after the occurrence of $\downarrow a$, $\downarrow$b inside the interval [1,2], and $\uparrow c$ inside [3,5] according to the system dynamic. If $\downarrow a$ does not occur at q_2 then the wagon motor has not responded. Thus the non-satisfaction of the corresponding expected consequent at this state indicates the occurrence of a fault belonging to Π_{F4}. If $\downarrow a$ has occurred, then S will transit to the state q_3. If $\uparrow b$ has not occurred, then the non-satisfaction of the corresponding expected consequent provides the label $l = \{F_2\}$ to indicate that the sensor b is faulty, stuck-off, since the wagon has responded. Similarly the non occurrence of $\downarrow$b at q_4 indicates that the sensor b is stuck-on. Consequently $EC_{\uparrow R}$ can be written:

$$EC_{\uparrow R} = \begin{Bmatrix} \{\uparrow R, \downarrow a, (q_2, [1,2], F_4)\}, \{\downarrow a, \uparrow b, (q_3, [3,5], F_2)\}, \\ \{\uparrow b, \downarrow b, (q_4, [1,2], F_2)\}, \{\downarrow b, \uparrow c, (q_5, [3,5], F_3)\} \end{Bmatrix}.$$

Similarly the expected consequent for the enablement of the command L can be written:

$$EC_{\uparrow L} = \begin{Bmatrix} \{\uparrow L, \downarrow c, (q_8, [1,2], F_4)\}, \{\uparrow L, \downarrow b, (q_{12}, [1,2], F_4)\}, \\ \{\downarrow b, \uparrow a, (q_{13}, [3,5], F_1\}, \{\downarrow c, \uparrow b, (q_9, [3,5], F_2)\} \end{Bmatrix}.$$

3.4 Local Diagnosers Construction

Two local diagnosers D^1 and D^2 are constructed for the sub models S^1 and S^2. Each local diagnoser contains, besides the states of the local desired behavior model, all the faulty states that can be reached by the occurrence of a fault belonging to one of the fault partitions. Each one of these faulty states is reached due to the non-satisfaction either of the enablement condition of an event or of an expected consequent. This makes the diagnoser declaring a fault. The diagnosers D^1 and D^2 are depicted respectively in Figure 3 and Figure 4. Each diagnoser state is determined by testing whether the enablement condition, or the expected consequent, is satisfied (the next state is a desire one) or not (the next state is faulty). The fault labels are calculated by determining the reason of the non-satisfaction.

The diagnoser can be initiated at any state distinguished by its output vector, i.e., the states with the dotted entrant arrows. If the diagnoser is initiated at any state distinguished by an event, the diagnoser cannot diagnose a past occurrence of a fault. As an example, the faulty states reached by an unsatisfied expected consequent cannot be distinguished from the ones of the desired behavior if the diagnoser was initiated at one of these states.

The system is F-codiagnosable if it satisfies the conditions (5), (6) and (7). The condition (5) is satisfied since the two diagnosers authorize both the events observable by them: $\forall q,\ en^1_{\uparrow b}.en^2_{\uparrow b} \neq 0$ and $en^1_{\downarrow b}.en^2_{\downarrow b} \neq 0$. The condition (6) is also verified since the local diagnosers can diagnose with certainty the occurrence of a fault belonging to one of the fault partitions of Σ_{Π}.

D^1 diagnoses with certainty the faults belonging to one of Π_{F1}, Π_{F2} and Π_{F4} while D^2 diagnoses with certainty the faults belonging to one of Π_{F2}, Π_{F3} and Π_{F4}. Finally (7) holds since the delay required to diagnose a fault belonging to one of the fault partitions, in the worst case and for any one of the two diagnosers, is finite and equal to 6 events. If we consider the non-satisfaction of an expected consequent as an event then starting from any diagnoser state of the desired behavior, the longest event sequence required to decide the occurrence of a fault is maximally equal to 6. As an example, starting from the state 7 of D^1, the detection of the occurrence of a fault belonging to one of Π_{F1}, Π_{F2} or Π_{F4} requires, respectively, 6 events (state 21), 5 events (state 20) and 5 events (state 19). Thus, the system is F-codiagnosable.

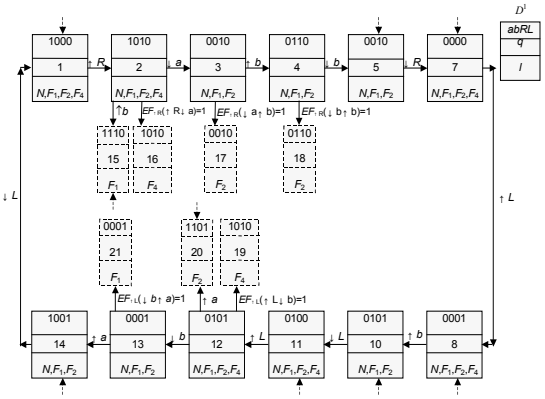


Figure 3: Local event-state-based diagnoser, D^1.

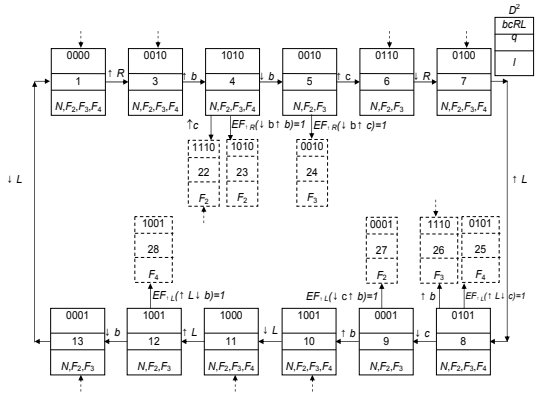


Figure 4: Local event-state-based diagnoser, D^2.

4 CONCLUSIONS

In this paper, a decentralized diagnosis approach is proposed to diagnose manufacturing systems. This approach is based on several local diagnosers. They diagnose together faults, which violate the specification language representing the desired behavior of the monitored system.

A simulation tool based on Stateflow of Matlab® is constructed in order to test and validate the proposed approach on application examples. This tool is based on a library of component models to design and to test the performances of diagnosis module for different applications.

We are developing a distributed diagnosis module to perform the diagnosis of manufacturing systems. This module uses the timed-event-state-based diagnoser, proposed in this paper, as a local diagnoser in a distributed structure.

REFERENCES

Balemi S, Hoffmann G.J., Gyugyi P, Wong-Toi H., Franklin G.F. Supervisory control of a rapid thermal multiprocessor, *IEEE Transactions on Automatic Control*, vol. 38, n°7, pp. 1040-105, 1993.

Debouk R., Lafortune S., and Teneketzis D. Coordinated decentralized protocols for failure diagnosis of DES, *Discrete Event Dynamic Systems: Theory and Applications*, 10(1-2):33–86, 2000.

Lin F., Diagnosability of Discrete Event Systems and its Applications, In *Discrete Event Dynamic Systems4*, Kluwer Academic Publishers, USA. 1994.

Pandalai D., L. E. N. Holloway, Template Languages for Fault Monitoring of Timed Discrete Event Processes, In *IEEE Transactions On Automatic Control 45(5),* 2000.

Philippot A., Sayed Mouchaweh M., Carré-Ménétrier V., Multi-models approach for the diagnosis of Discrete Events Systems, In *IMACS'05*, *International conference on Modelling, Analyse and Control of Dynamic Systems*, Paris-France, 2005.

Qiu W., Decentralized/distributed failure diagnosis and supervisory control of DES, *PhD Thesis,* the Iowa State University, USA, 2005.

Ramadge P., Wonham W., Supervisory control of a class of discrete event processes, In *SIAM J. Control Optim.* 25(1), 1987.

Sampath M., Segupta R., Lafortune S., Sinnamohideen K., Teneketzis D., Diagnosability of discrete event systems, In *11th Int. Conf. Analysis Optimization of Systems: DES*, France, 1994.

Tripakis S., Fault Diagnosis for Timed Automata, *7th International Symposium on Formal Techniques in Real Time and Fault Tolerant Systems* (FTRTFT'02), Oldenburg Germany, 2002.

Wang Y., Supervisory Control of Boolean Discrete-Event Systems, *Thesis of Master of Applied Sciences*, University of Toronto, Canada, 2000.

Zad S. H., Kwong R. H., Wonham W. M., Fault Diagnosis in DES: Framework and model reduction, *IEEE Transactions On Automatic Control 48(7),* 2003.

DUAL CONTROLLERS FOR DISCRETE-TIME STOCHASTIC AMPLITUDE-CONSTRAINED SYSTEMS

A. Królikowski and D. Horla
Poznań University of Technology
Institute of Control and Information Engineering
ul.Piotrowo 3A, 60-965 Poznań, Poland
Andrzej.Krolikowski@put.poznan.pl

Keywords: Input constraint. Suboptimal dual control.

Abstract: The paper considers a suboptimal solution to the dual control problem for discrete-time stochastic systems in the case of amplitude constraint imposed on the control signal. The objective of the control is to minimize the variance of the output around the given reference sequence. The presented approaches are based on: an MIDC (Modified Innovation Dual Controller) derived from an IDC (Innovation Dual Controller), a TSDSC (Two-stage Dual Suboptimal Control, and a PP (Pole Placement) controller. Finally, the certainty equivalence (CE) control method is included for comparative analysis. In all algorithms, the standard Kalman filter equations are applied for estimation of the unknown system parameters. Example of second order system is simulated in order to compare the performance of control methods. Conclusions yielded from simulation study are given.

1 INTRODUCTION

Much work has been done on the optimal control of stochastic systems which contain parametric uncertainty. The problem is inherently related with the dual control problem originally presented by Fel'dbaum who suggested that in the dual control, the problems of learning and control should be considered simultaneously in order to minimize the cost function. In general, learning and controlling have contradictory goals, particularly for the finite horizon control problems. The concept of duality has inspired the development of many control techniques which involve the dual effect of the control signal. They can be separated in two classes: explicit dual and implicit dual (Bayard and Eslami, 1985). Unfortunately, the dual approach does not result in computationally feasible optimal algorithms. A variety of suboptimal solutions has been proposed and many of them were heuristic identifier-controller structures. Other controllers like minimax controllers (Sebald, 1979), Bayes controllers (Sworder, 1966) or MRAC (Model Reference Adaptive Controller) (Åström and Wittenmark, 1989) are available.

The objective of this paper is to present and compare different approaches to suboptimal solution of the minimum variance control problem of discrete-time stochastic systems with unknown parameters. In this paper, an amplitude-constrained control input is considered which is an important practical case. A majority of proposed solutions in the literature does not include the input constraint into the design of control system. The saturation imposed on control signal deteriorates the probability density function (pdf) of the state from the Gaussian which makes finding an optimal control difficult even when system parameters are known. The dual methods described here are: the MIDC method which is the modification of the IDC (R. Milito and Cadorin, 1982) approach, the method based on the two-stage dual suboptimal control (TSDSC) approach (Maitelliand and Yoneyama, 1994) and the method based on the pole placement approach (Filatov and Unbehauen, 2004).

The Iteration in Policy Space (IPS) algorithm and its reduced complexity version were proposed by Bayard (Bayard, 1991) for a general nonlinear system. In this algorithm the stochastic dynamic programming equations are solved forward in time ,using a nested stochastic approximation technique. The method is based on a specific computational architecture denoted as a H block. The method needs a filter propagating the state and parameter estimates with as-

sociated covariance matrices.

In (Królikowski, 2000), some modifications including input constraint have been introduced into the original version of the IPS algorithm and its performance has been compared with MIDC algorithm.

This paper has a tutorial nature, and the possibility of incorporating the input constraint into the control algorithms was the motivation for a selection of the overviewed approaches.

Performance of the considered algorithms is illustrated by simulation study of second-order system with control signal constrained in amplitude.

2 CONTROL PROBLEM FORMULATION

Consider a discrete-time linear single-input single-output system described by ARX model

$$A(q^{-1})y_k = B(q^{-1})u_k + w_k, \tag{1}$$

where $A(q^{-1}) = 1 + a_{1,k}q^{-1} + \cdots + a_{na,k}q^{-na}$, $B(q^{-1}) = b_{1,k}q^{-1} + \cdots + b_{nb,k}q^{-nb}$, y_k is the output available for measurement, u_k is the control signal, $\{w_k\}$ is a sequence of independent identically distributed gaussian variables with zero mean and variance σ_w^2. Process noise w_k is statistically independent of the initial condition y_0. The system (1) is parametrized by a vector θ_k containing $na+nb$ unknown parameters $\{a_{i,k}\}$ and $\{b_{i,k}\}$ which in general can be assumed to vary according to the equation

$$\underline{\theta}_{k+1} = \Phi\underline{\theta}_k + e_k \tag{2}$$

where Φ is a known matrix and $\{e_k\}$ is a sequence of independent identically distributed gaussian variables with zero mean and variance matrix R_e. Particularly, for the constant parameters we have

$$\underline{\theta}_{k+1} = \underline{\theta}_k = \underline{\theta} = (b_1, \cdots, b_{nb}, a_1, \cdots a_{na})^T, \tag{3}$$

and then $\Phi = I$, $e_k = 0$ in (2).

The control signal is subjected to an amplitude constraint

$$|\, u_k \,| \leq \alpha \tag{4}$$

and the information state I_k at time k is defined by

$$I_k = [y_k, ..., y_1, u_{k-1}, ..., u_0, I_0] \tag{5}$$

where I_0 denotes the initial conditions.

An admissible control policy Π is defined by a sequence of controls $\Pi = [u_0, ..., u_{N-1}]$ where each control u_k is a function of I_k and satisfies the constraint (4). The control objective is to find a control policy Π which minimizes the following expected cost function

$$J = E[\sum_{k=0}^{N-1}(y_{k+1} - r_{k+1})^2] \tag{6}$$

where $\{r_k\}$ is a given reference sequence. An admissible control policy minimizing (6) can be labelled by CCLO (Constrained Closed-Loop Optimal) in keeping with the standard nomenclature, i.e. $\Pi^{\mathrm{CCLO}} = [u_0^{\mathrm{CCLO}}, ..., u_{N-1}^{\mathrm{CCLO}}]$. This control policy has no closed form, and control policies presented in the following section can be viewed as a suboptimal approach to the Π^{CCLO}.

3 SUBOPTIMAL DUAL CONTROL METHODS

In this section, we shall briefly describe three methods giving an approximate solution to the problem formulated in Section 2. The first one is the MIDC algorithm based on the IDC approach (R. Milito and Cadorin, 1982) which is an explicit dual control approach.

3.1 Method based on the Innovation Dual Control (IDC) Approach: Derivation of Π^{MIDC}

The IDC has been derived for system (1) with unconstrained control and constant parameters (3). The following cost function was considered

$$J = \frac{1}{2}E[(y_{k+1} - r_{k+1})^2 - \lambda_{k+1}\varepsilon_{k+1}^2 | I_k] \tag{7}$$

where $\lambda_{k+1} \geq 0$ is the learning weight, and ε_{k+1} is the innovation, see (16).

The modified IDC, u_k^{MIDC}, takes the constraint into account which results in the following closed-form expression

$$u_k^{\mathrm{MIDC}} = \\ = -\mathrm{sat}\left(\frac{[(1-\lambda_{k+1})\underline{p}_{b_1\underline{\theta}^*,k}^T + \hat{\underline{\theta}}_k^{*T}\hat{b}_{1,k}]\underline{s}_k^* - \hat{b}_{1,k}r_{k+1}}{(1-\lambda_{k+1})p_{b_1,k} + \hat{b}_{1,k}^2}; \alpha\right) \tag{8}$$

where

$$\begin{aligned}\underline{s}_k &= (u_k, u_{k-1}, \ldots, u_{k-nb+1}, -y_k, \ldots, -y_{k-na+1})^T = \\ &= (u_k, \underline{s}_k^{*T})^T,\end{aligned} \tag{9}$$

and following partitioning is introduced for parameter covariance matrix P_k

$$P_k = \begin{bmatrix} p_{b_1,k} & \underline{p}_{b_1\underline{\theta}^*,k}^T \\ \underline{p}_{b_1\underline{\theta}^*,k} & P_{\underline{\theta}^*,k} \end{bmatrix} \tag{10}$$

corresponding to the partition of $\underline{\theta}$

$$\underline{\theta} = (b_1, \underline{\theta}^{*T})^T \tag{11}$$

with

$$\underline{\theta}^* = (b_2, \ldots, b_{nb}, a_1, \ldots, a_{na})^T. \tag{12}$$

The estimates $\hat{\underline{\theta}}_k$ needed to calculate u_k^{MIDC} can be obtained in many ways. A common way is to use the standard Kalman filter in a form of suitable recursive procedure for parameter estimation, i.e.

$$\hat{\underline{\theta}}_{k+1} = \Phi\hat{\underline{\theta}}_k + \underline{k}_{k+1}\varepsilon_{k+1} \tag{13}$$

$$\underline{k}_{k+1} = \Phi P_k \underline{s}_k [\underline{s}_k^T P_k \underline{s}_k + \sigma_w^2]^{-1} \tag{14}$$

$$P_{k+1} = [\Phi - \underline{k}_{k+1}\underline{s}_k^T]P_k\Phi^T + R_e, \tag{15}$$

$$\varepsilon_{k+1} = y_{k+1} - \underline{s}_k^T\hat{\underline{\theta}}_k. \tag{16}$$

3.2 Method based on the Two-stage Dual Suboptimal Control (TSDSC) Approach: Derivation of Π^{TSDSC}

The TSDSC proposed in (Maitelliand and Yoneyama, 1994) has been derived for system (1) with stochastic parameters (2). Below this method is extended for the input-constrained case. The cost function considered for TSDSC is given by

$$J = \frac{1}{2}E[(y_{k+1} - r)^2 + (y_{k+2} - r)^2 | I_k] \tag{17}$$

and according to (Maitelliand and Yoneyama, 1994) can be obtained as a quadratic form in u_k and u_{k+1}, i.e.

$$J = \frac{1}{2}[au_k + bu_{k+1} + cu_ku_{k+1} + du_k^2 + eu_{k+1}^2] \tag{18}$$

where a, b, c, d, e are expressions depending on current data $\underline{s}_k^*$, reference signal r and parameter estimates $\hat{\underline{\theta}}_k$ (Maitelliand and Yoneyama, 1994). Solving a necessary optimality condition the unconstrained control signal is

$$u_k^{\mathrm{TSDSC,un}} = \frac{bc - 2ae}{4de - c^2}. \tag{19}$$

This control law has been taken for simulation analysis in (Maitelliand and Yoneyama, 1994). Imposing the cutoff the constrained control signal is

$$u_k^{\mathrm{TSDSC,co}} = sat(u_k^{\mathrm{TSDSC,un}}; \alpha). \tag{20}$$

The cost function (18) can be represented as a quadratic form

$$J = \frac{1}{2}[\underline{u}_k^T A \underline{u}_k + \underline{b}^T\underline{u}_k] \tag{21}$$

where $\underline{u}_k = (u_k, u_{k+1})^T$, and

$$A = \begin{bmatrix} d & \frac{1}{2}c \\ \frac{1}{2}c & e \end{bmatrix}, \underline{b} = \begin{bmatrix} a \\ b \end{bmatrix}. \tag{22}$$

The condition $4de - c^2 > 0$ together with $d > 0$ implies positive definitness and guarantees convexity. Minimization of (21) under constraint (4) is a standard QP problem resulting in $\underline{u}_k^{\mathrm{TSDSC,qp}}$. The constrained control $u_k^{\mathrm{TSDSC,qp}}$ is then applied to the system in receding horizon framework.

3.3 Method based on the Pole Placement (PP) Approach: Derivation of Π^{PP}

Let the desired stable closed-loop polynomial be described by $A^*(q^{-1}) = 1 + a_1^*q^{-1} + \cdots + a_{n^*}q^{-n^*}$. A dual version of a direct adaptive PP controller proposed in (N.M. Filatov and Keuchel, 1993; Filatov and Unbehauen, 2004) has been derived for system (1) where integral actions can be included. To this end, a bicriterial approach has been used to solve the synthesis problem. The two criteria correspond to the two goals of the dual adaptive control, namely to control the system output close to the reference signal, and to accelerate the parameter estimation process for future control improvment. Incorporating the amplitude constraint of the control input yields

$$u_k^{\mathrm{PP}} = sat\Big(u_k^{\mathrm{CAUT}} + \\ + \eta\, trP_k sign(p_{d_0,k}\bar{u}_k^{\mathrm{CAUT}} + \underline{p}_{d_0\underline{p}_1,k}^T\underline{m}_{1,k})\,;\,\alpha\Big) \tag{23}$$

where u_k^{CAUT} is the cautious action given by

$$u_k^{\mathrm{CAUT}} = -\frac{(\underline{p}_{r_0\underline{p}_0,k}^T + \hat{\underline{p}}_{0,k}^T\hat{r}_{0,k})\underline{m}_{0,k} - \hat{r}_{0,k}r_k}{p_{r_0,k} + \hat{r}_{0,k}^2}, \tag{24}$$

$\bar{u}_k^{\mathrm{CAUT}} = u_k^{\mathrm{CAUT}} + \sum_{i=1}^{n^*} a_i^*u_{k-i}$, $\underline{p}_0 = (s_0, \ldots, s_{ns}, r_1, \ldots, r_{nr})^T$, $\underline{m}_{0,k} = (y_k, \ldots, y_{k-ns}, u_{k-1}, \ldots, u_{k-nr})^T$, and $\eta \geq 0$ is the parameter responsible for probing. In this case the following partitioning is introduced for parameter covariance matrix P_k

$$P_k = \begin{bmatrix} p_{d_0,k} & \underline{p}_{d_0\underline{p}_1,k}^T \\ \underline{p}_{d_0\underline{p}_1,k} & P_{\underline{p}_1,k} \end{bmatrix} \tag{25}$$

corresponding to the partition of parameter vector $\underline{p}$

$$\underline{p} = (-d_0, \underline{p}_1^T)^T \tag{26}$$

where

$$\underline{p}_1 = (-d_1, \ldots, d_{nd}, -f_1, \ldots, -f_{nf}, r_0, \ldots, r_{nr}, s_0, \ldots, s_{ns})^T \tag{27}$$

and

$$\underline{m}_k = (\bar{u}_k, \underline{m}_{1,k}^T)^T \tag{28}$$

with $\underline{m}_{1,k} = (\bar{u}_{k-1}, \ldots, \bar{u}_{k-nd}, \bar{y}_k, \ldots, \bar{y}_{k-nf+1}, u_{k-l+1}, \ldots, u_{k-l-nr+2}, y_{k-l+2}, \ldots, y_{k-l+ns+2})^T$. The filtered output and input signals are obtained as $\bar{y}_k = A^*(q^{-1})y_k$, $\bar{u}_k = A^*(q^{-1})u_k$.

The corresponding diophantine equation and Bezout identity are

$$A(q^{-1})[r_0 + q^{-1}R(q^{-1})] + q^{-1}B(q^{-1})S(q^{-1}) = r_0 A^*(q^{-1}), \tag{29}$$

$$A(q^{-1})D(q^{-1}) + B(q^{-1})F(q^{-1}) = r_0 q^{-l+2}, \tag{30}$$

where the polynomial degrees are: $nr = na-1$, $ns = na-\kappa-1$, $l = na+nb$, $nd = nb-2$, $nf = na-1$, and κ is the number of possible integrators in the system.

It can be shown that the filtered output $\bar{y}_k$ can be represented in the following regressor form

$$\bar{y}_k = \underline{p}^T \underline{m}_{k-1} + v_k \tag{31}$$

For estimation of parameters $\underline{p}$ (note that parameters $\underline{p}_0$ are included into $\underline{p}$) the Kalman filter algorithm (13)-(16) can again be used where $\hat{\underline{\theta}}_k$ should be replaced by $\hat{\underline{p}}_k$, $\underline{s}_k$ should be replaced by $\underline{m}_k$, ε_{k+1} should be calculated as $\varepsilon_{k+1} = \bar{y}_{k+1} - \underline{m}_k^T \hat{\underline{p}}_k$, and the variance σ_w^2 should be replaced by the variance σ_v^2 which can be evaluated from (29), (30), (1).

4 SIMULATION TESTS

Performance of the described control methods is illustrated through the example of a second-order system with the following true values: $a_1 = -1.8$, $a_2 = 0.9$, $b_1 = 1.0$, $b_2 = 0.5$, where the Kalman filter algorithm (13)-(16) was applied for estimation. The initial parameter estimates were taken half their true values with $P_0 = 10I$. The reference signal was a square wave ± 3, and then the minimal value of constraint α ensuring the tracking is $\alpha_{min} = 3\frac{|A(1)|}{|B(1)|} = 0.2$. Fig. 1 shows the reference, output and input signals during tracking process under the constraint $\alpha = 1$ for all control policies.

For the control policy Π^{MIDC} the constant learning weight was $\lambda_k = \lambda = 0.98$. The policy Π^{PP} was simulated for third order polynomial $A^*(q^{-1})$ having poles at $0.2 \pm i0.1$, -0.1, and for the probing weight $\eta = 0.2$. The control policy Π^{CE} can easily be obtained from MIDC by taking $p_{b_1,k} = 0$, $\underline{p}_{b_1\underline{\theta}^*,k}^T = 0$.

Next, the simulated performance index

$$\bar{J} = \sum_{k=0}^{N-1} (y_{k+1} - r_{k+1})^2$$

was considered. The plots of $\bar{J}$ versus the constraint α are shown in Figs. 2, 3 for $\sigma_w^2 = 0.05, 0.1$, respectively, and $N = 1000$. The control $u_k^{\mathrm{TSDSC,qp}}$ was obtained solving the minimization of quadratic form (20) using MATLAB function *quadprog*. The performance of this control is not included in plots of Figs. 5, 6, because it performs surprisingly essentially inferior with respect to $u_k^{\mathrm{TSDSC,co}}$. In the latter case, a short-term behaviour phenomenon (G.P. Chen and Hope, 1993) can be observed in Figs. 2, 3. This means that when the cutoff method is used then the range of constraint α can be found where for increasing α the performance index is also increasing.

5 CONCLUSIONS

This paper presents various approaches toward a suboptimal solution to the discrete-time dual control problem under the amplitude-constrained control signal. A simulation example of second-order system is given and the performance of the presented control policies is compared by means of the simulated performance index.

The MIDC method seems to be a good suboptimal dual control approach, however it has been found that the MIDC control is quite sensitive to the value of the learning weight λ. In (Królikowski, 2000) it has been found that this method often performs very close to the IPS algorithm (Bayard, 1991).

Performance of all control policies except $\Pi^{\mathrm{TSDSC,co}}$ is comparable, however the differences between all methods are less noticeable when the constraint α gets tight, i.e. when $\alpha \to \alpha_{min}$. In all considered control policies except $u_k^{\mathrm{TSDSC,co}}$, the performance index increases when the input amplitude constraint gets more tight. This means that for $u_k^{\mathrm{TSDSC,co}}$ the effect of the short term behaviour phenomenon discussed in (G.P. Chen and Hope, 1993) could appear.

REFERENCES

Åström, H. and Wittenmark, B. (1989). *Adaptive Control*. Addison-Wesley.

Bayard, D. (1991). A forward method for optimal stochastic nonlinear and adaptive control. *IEEE Trans. Automat. Contr.*, 9:1046–1053.

Bayard, D. and Eslami, M. (1985). Implicit dual control for general stochastic systems. *Opt. Contr. Appl.& Methods*, 6:265–279.

Filatov, N. and Unbehauen, H. (2004). *Dual Control*. Springer.

G.P. Chen, O. M. and Hope, G. (1993). Control limits consideration in discrete control system design. *IEE Proc.-D*, 140(6):413–422.

Królikowski, A. (2000). Suboptimal lqg discrete-time control with amplitude-constrained input: dual versus non-dual approach. *European J. Control*, 6:68–76.

Maitelliand, A. and Yoneyama, T. (1994). A two-stage dual suboptimal controller for stochastic systems using approximate moments. *Automatica*, 30:1949–1954.

N.M. Filatov, H. U. and Keuchel, U. (1993). Dual pole-placement controller with direct adaptation. *Automatica*, 33(1):113–117.

R. Milito, C.S. Padilla, R. P. and Cadorin, D. (1982). An innovations approach to dual control. *IEEE Trans. Automat. Contr.*, 1:132–137.

Sebald, A. (1979). Toward a computationally efficient optimal solution to the lqg discrete-time dual control problem. *IEEE Trans. Automat. Contr.*, 4:535–540.

Sworder, D. (1966). *Optimal Adaptive Control Systems*. Academic Press, New York.

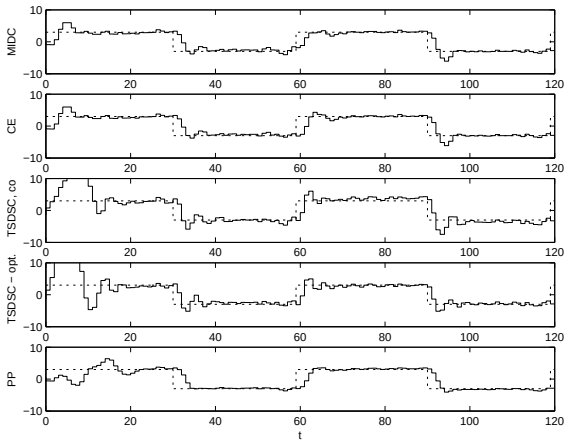


Figure 1: Reference, output and control signals for Π^{MIDC}, Π^{CE}, Π^{TSDSC}, Π^{PP} and $\alpha = 1$.

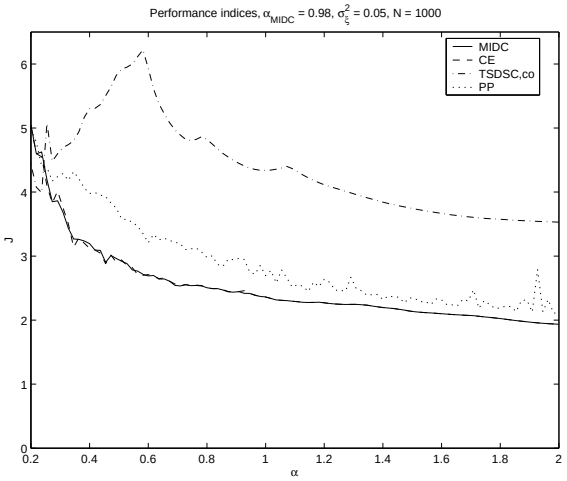


Figure 2: Plots of performance indices for $\sigma_w^2 = 0.05$.

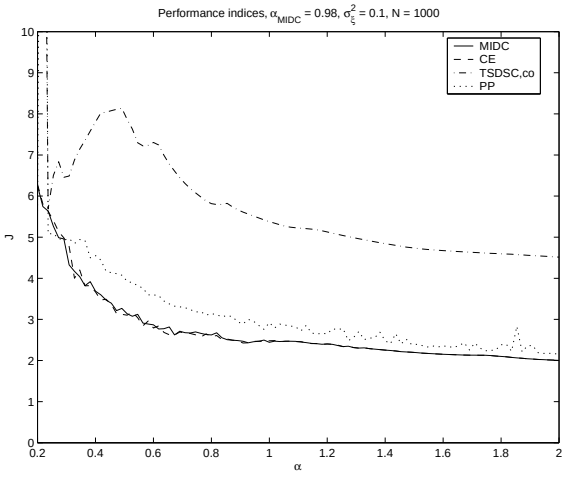


Figure 3: Plots of performance indices for $\sigma_w^2 = 0.1$.

TRANSFORMATION ANALYSIS METHODS FOR THE BDSPN MODEL

Karim Labadi
EPMI- ECS, 13 boulevard de l'Hautil 95092 Cergy Pontoise Cedex, France
k.labadi@epmi.fr

Haoxun Chen and Lionel Amodeo
LOSI-ICD (FRE CNRS 2848), 12 rue Marie Curie, BP 2060, 10010 Troyes Cedex, France
hoaxun.chen@utt.fr,lionel.amodeo@utt.fr

Keywords: Petri nets, BDSPN model, modelling, analysis, discrete event systems.

Abstract: The work of this paper contributes to the structural analysis of batch deterministic and stochastic Petri nets (BDSPNs). The BDSPN model is a class of Petri nets introduced for the modelling, analysis and performance evaluation of discrete event systems with batch behaviours. The model is particularly suitable for the modelling of flow evolution in discrete quantities (batches of variable sizes) in a system with activities performed in batch modes. In this paper, transformation procedures for some subclasses of BDSPN are developed and the necessity of the introduction of the new model is demonstrated.

1 INTRODUCTION

A Petri net model, called batch deterministic and stochastic Petri nets (BDSPN), was introduced for the modelling, and performance evaluation of discrete event systems with batch behaviours. As we know, industrial systems are often characterized as batch processes where materials are processed in batches and many operations are usually performed in batch modes to take advantages of the economies of scale or because of the batch nature of customer orders. It is shown in our previous papers that the model is a powerful tool for both analysis and simulation of those systems and its capability to meet real needs was demonstrated through applications to logistical systems (Labadi, et al. 2005, 2007; Chen, et al. 2005). The objective of this paper is to study the transformation of a BDSPN model into an equivalent classical Petri net model. Such a transformation is possible for some cases for which the corresponding transformation procedures are developed. We will also show that for the model with variable arc weights depending on its marking, the transformation is impossible. This study allows us to establish a relationship between BDSPNs and classical discrete Petri nets and to demonstrate the necessity of introducing the BDSPN model.

2 DESCRIPTION OF THE MODEL

BDSPN model is developed from deterministic and stochastic Petri nets (Marsan, et al. 1987; Lindemann, 1998) by introducing batch components (batch places, batch tokens, and batch transitions) and new transition enabling and firing rules. Firstly, we recall the basic definition and the dynamical behavior of the model (Labadi, et al. 2005, 2007; Chen, et al. 2005).

2.1 Definition of the Model

A BDSPN is a nine tuple (P, T, I, O, V, W, Π, D, μ_0) where:

$P = P_d \cup P_b$ is a finite set of places consisting of the discrete places in set P_d and the batch places in set P_b. Discrete places and batch places are represented by single circles and squares with an embedded circle, respectively. Each token in a discrete place is represented by a dot, whereas each batch token in a batch place is represented by an Arabic number that indicates its size.

$T = T_i \cup T_d \cup T_e$ is a set of transitions consisting of immediate transitions in set T_i, the deterministic timed transitions in set T_d, and exponentially distributed transitions in set T_e. T can also be

partitioned into $T_D \cup T_B$: a set of discrete transitions T_D and a set of batch transitions T_B. A transition is said to be a *batch transition* (respectively a *discrete transition*) if it has at least an input batch place (respectively if it has no input batch place).

$I \subseteq (P \times T)$, $O \subseteq (T \times P)$, and $V \subseteq (P \times T)$ define the input arcs, the output arcs and the inhibitor arcs of all transitions, respectively. It is assumed that only immediate transitions are associated with inhibitor arcs and that the inhibitor arcs and the input arcs are two disjoint sets.

$W: (I \cup O \cup V) \times IN^{|P|} \rightarrow IN$, where IN is the set of nonnegative integers, defines the weights for all ordinary arcs and inhibitor arcs. For any arc $(i, j) \in I \cup O \cup V$, its weight $W(i, j)$ is a linear function of the M-marking with integer coefficients α, β, i.e., $w(i, j) = \alpha_{ij} + \sum_{p \in P} \beta_{(i,j)p} \times M(p)$. The weight $w(i, j)$ is assumed to take a positive value.

$\Pi: T \rightarrow IN$ is a priority function assigning a priority to each transition. Timed transitions are assumed to have the lowest priority, i.e.; $\Pi(t) = 0$ if $t \in T_d \cup T_e$. For each immediate transition $t \in T_i$, $\Pi(t) \geq 1$.

$D: T \rightarrow [0, \infty)$ defines the firing times of all transitions. It specifies the mean firing delay for each exponential transition, a constant firing delay for each deterministic transition, and a zero firing delay for each immediate transition

μ_0: $P \rightarrow IN \cup 2^{IN}$ is the initial μ-marking of the net, where 2^{IN} consists of all subsets of IN, $\mu_0(p) \in IN$ if $p \in P_d$, and $\mu_0(p) \in 2^{IN}$ if $p \in P_b$.

The state of the net is represented by its μ-marking. We use two different ways to represent the μ-marking of a discrete place and the μ-marking of a batch place. The first marking is represented by a nonnegative integer, whereas the second marking is represented by a multiset of nonnegative positive integers. The multiset may contain identical elements and each integer in the multiset represents a batch token with a given size. Moreover, for defining the net, another type of marking, called M-marking, is also introduced. For each discrete place, its M-marking is the same as its μ-marking, whereas for each batch place its M-marking is defined as the total size of the batch tokens in the place.

2.2 Transition Enabling and Firing

The state or μ-marking of the net is changed with two types of transition firing called "*batch firing*" and "*discrete firing*". They depend on whether a transition has no batch input places. In the following, a place connected with a transition by an arc is referred to as input, output, and inhibitor place, depending on the type of the arc. The set of input places, the set of output places and the set of inhibitor places of transition t are denoted by ${}^\bullet t$, $t^\bullet$, and ${}^\circ t$, respectively, where ${}^\bullet t = \{ p \mid (p, t) \in I \}$, $t^\bullet = \{ p \mid (t, p) \in O \}$, and ${}^\circ t = \{ p \mid (p, t) \in V \}$. The weights of the input arc from a place p to a transition t, of the output arc from t to p are denoted by $w(p, t)$, $w(t, p)$ respectively.

2.2.1 Batch Enabling and Firing Rules

A batch transition t is said to be enabled at μ-marking μ if and only if there is a *batch firing index* (positive integer) $q \in IN$ ($q > 0$) such that:

$$\forall p \in {}^\bullet t \cap P_b, \exists b \in \mu(p): \quad q = b / w(p,t) \tag{1}$$

$$\forall p \in {}^\bullet t \cap P_d, \quad M(p) \geq q \times w(p,t) \tag{2}$$

$$\forall p \in {}^\circ t, \quad M(p) < w(p,t) \tag{3}$$

The batch firing of t leads to a new μ-marking μ':

$$\forall p \in {}^\bullet t \cap P_d : \mu'(p) = \mu(p) - q \times w(p,t) \tag{4}$$

$$\forall p \in {}^\bullet t \cap P_b : \mu'(p) = \mu(p) - \{q \times w(p,t)\} \tag{5}$$

$$\forall p \in t^\bullet \cap P_d : \mu'(p) = \mu(p) + q \times w(t,p) \tag{6}$$

$$\forall p \in t^\bullet \cap P_b : \mu'(p) = \mu(p) + \{q \times w(t,p)\} \tag{7}$$

2.2.2 Discrete Enabling and Firing Rules

A discrete transition t is said to be enabled at μ-marking μ (its corresponding M-marking M) if and only if:

$$\forall p \in {}^\bullet t, \quad M(p) \geq w(p,t) \tag{8}$$

$$\forall p \in {}^\circ t, \quad M(p) < w(p,t) \tag{9}$$

The discrete firing of t leads to a new μ-marking μ':

$$\forall p \in {}^\bullet t : \quad \mu'(p) = \mu(p) - w(p,t) \tag{10}$$

$$\forall p \in t^\bullet \cap P_d : \quad \mu'(p) = \mu(p) + w(t,p) \tag{11}$$

$$\forall p \in t^\bullet \cap P_b : \quad \mu'(p) = \mu(p) + \{w(t,p)\} \tag{12}$$

2.2.3 An Illustrative Example

We describe as an example the BDSPN model of a simple assembly-to-order system that requires two components shown in Fig. 1. In the model, discrete places p_1 and p_2 are used to represent the stock of component A and the stock of component B respectively. Batch place p_3 is used to represent batch customer orders with different and variable sizes. To fill a customer order of size b, we need $b \times w(p_1, t_1) = 2b$ units of component A from the stock

represented by p_1 and $b \times w(p_2, t_1) = b$ units of component B from the stock represented by p_2. These components will be assembled to b units of final product to fill the order. For instance, at the current µ-marking $\mu_0 = (4, 3, \{4, 2, 3\}, \varnothing, 0)^T$, it is possible to fill the batch customer order $b = 2$ in batch place p_3 since the batch transition t_1 is enabled with $q = b/ w(p_3, t_1) = 2$. After the batch firing of transition t_1 (start assembly), the corresponding batch token $b = 2$ will be removed from batch place p_3, $q \times w(p_1, t_1) = 4$ discrete tokens will be removed from discrete place p_1, and $q \times w(p_2, t_1) = 2$ discrete tokens will be removed from discrete place p_2. A batch token with size equal to $q \times w(t_1, p_4) = 2$ will be created in batch place p_4 and 2 discrete tokens will be created in discrete place p_5. Therefore, the new µ-marking of the net after the batch firing is: $\mu_1 = (0, 1, \{4, 3\}, \{2\}, 2)^T$ and its corresponding M-marking is $M_1 = (0, 1, 7, 2, 2)^T$.

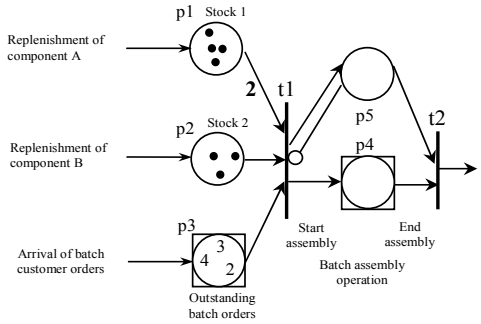


Figure 1: An assembly-to-order system.

2.3 Reachability Graph

For the analysis of the transformation procedures developed in the rest of this paper, we need to define in the following the concept of the *reachability graph* of the model.

A *µ-marking reachability graph* of a given BDSPN is a directed graph (V_μ, E_μ), where the set of vertices V_μ is given by the reachability set (μ_0^*): all µ-markings reachable from the initial marking μ_0 by firing a sequence of transitions and the initial marking), while the set of directed arcs E_μ is given by the feasible µ-marking changes in the BDSPN due to transition firing in all reachable µ-markings.

Similarly, we define *M-marking reachability graph* (V_M, E_M) which can be obtained from (V_μ, E_μ) by transforming each µ-marking in V_μ into its corresponding M-marking and by merging duplicated M-markings (and duplicated arcs).

3 TRANSFORMATION METHODS

The objective of this section is to study the transformation of a BDSPN model into an equivalent classical Petri net model.

3.1 Special Case

Firstly, we consider the case where all batch tokens in each batch place of the BDSPN are always identical. A batch place p_i is said to be *simple* if the sizes of its all batch tokens are the same for any µ-marking reachable from μ_0.

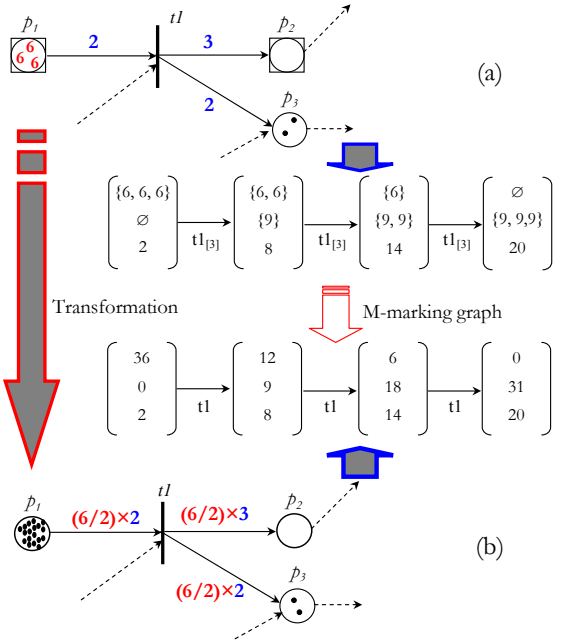


Figure 2: Transformation of a BDSPN (special case).

To illustrate the transformation method, we consider an example given in Fig. 2. The net (a) whose all batch places are simple can be easily transformed into an equivalent classical discrete Petri net (b). We observe that the two nets have the same M-marking reachability graph (the same dynamical behaviour). Indeed, the two properties, (*i*) all batch places of the net are simple and (*ii*) the net has no variable arc weight, lead to a *constant batch firing index* q_j for each batch transition $t_j \in T_b$ of the net. As formulated in the following procedure, the transformation method consists of (*i*) transforming each batch place into a discrete place and (*ii*) integrating the constant batch firing index of each batch transition in the weights of its input and output arcs in the resulting classical net in order to respect

the dynamic behaviour of the original batch net.

Transformation procedure (special case)

Given a BDSPN whose all batch places are *simple* and whose all arcs have a constant weight. This net can be transformed into an equivalent classical discrete Petri net, denoted by DPN by the following procedure:

Step1. The set of discrete places P_d of the BDSPN and their markings remain unchanged for the DPN.

$$\forall p_i \in P_d,\ M_0(p_i) = \mu_0(p_i) \tag{13}$$

Step2. Each batch place of the BDSPN is transformed into a discrete place M-marked in the DPN.

$$\forall p_i \in P_b,\ M_0(p_i) = \sum_{b \in \mu(p_i)} b \tag{14}$$

Step3. The set of transitions T of the BDSPN remains unchanged for the DPN.

Step4. The weight of each output arc of each batch place $p_i \in P_b$ of the BDSPN is set to the size of its batch tokens b_i.

$$\forall p_i \in P_b,\ \forall t_j \in p_i^{\bullet},$$

$$W^*(p_i,t_j) = W(p_i,t_j) \times \frac{b_i}{W(p_i,t_j)} = b_i \tag{15}$$

Step5. The weight of each output arc of each batch transition $t_j \in T_b$ of the BDSPN is set to its original weight multiplied by its batch firing index q_j.

$$\forall p_i \in P_b,\ \forall t_j \in p_i^{\bullet}, W^*(t_j,p_i)$$

$$= W(t_j,p_i) \times q_j = W(t_j,p_i) \times \frac{b_i}{W(p_i,t_j)}. \tag{16}$$

Step6. The weight of each output arc of each discrete transition $t_j \in T_d$ of the BDSPN remains unchanged for the DPN.

3.2 General Case

The proposed transformation procedure can be generalized to allow the transformation of a BDSPN containing batch places which are not simple into an equivalent classical Petri net. The transformation is feasible if we know in advance all possible batch firings of all batch transitions and all possible batch tokens which can appear in each batch place of the net during its evolution. In other words, the transformation can be performed when we well know the dynamic behaviour of the BDSPN for its given initial μ-markings μ_0.

(a) Let $D(t_j)$ denote the set of all *q-indexed transitions* $t_{j[q]}$ generated by the firings of the batch transition t_j with all possible batch firing indexes q during the evolution of the BDSPN starting from μ_0.

$$D(t_j) = \{t_{j[q]} \mid \exists \mu \in \mu_0^*, \mu[t_{j[q]} \rightarrow\} \tag{17}$$

where μ_0 denotes the set of reachable μ-markings from μ_0 and $\mu[t_{j[q]}\rightarrow$ denote that the batch transition t_j can be fired from μ with a batch firing index q.

(b) Let $D(p_i)$ denote the set of all possible batch tokens which can appear in the batch place p_i during the evolution the BDSPN starting from μ_0.

$$D(p_i) = \{b \mid \exists \mu \in \mu_0^*, b \in \mu(p_i)\} \tag{18}$$

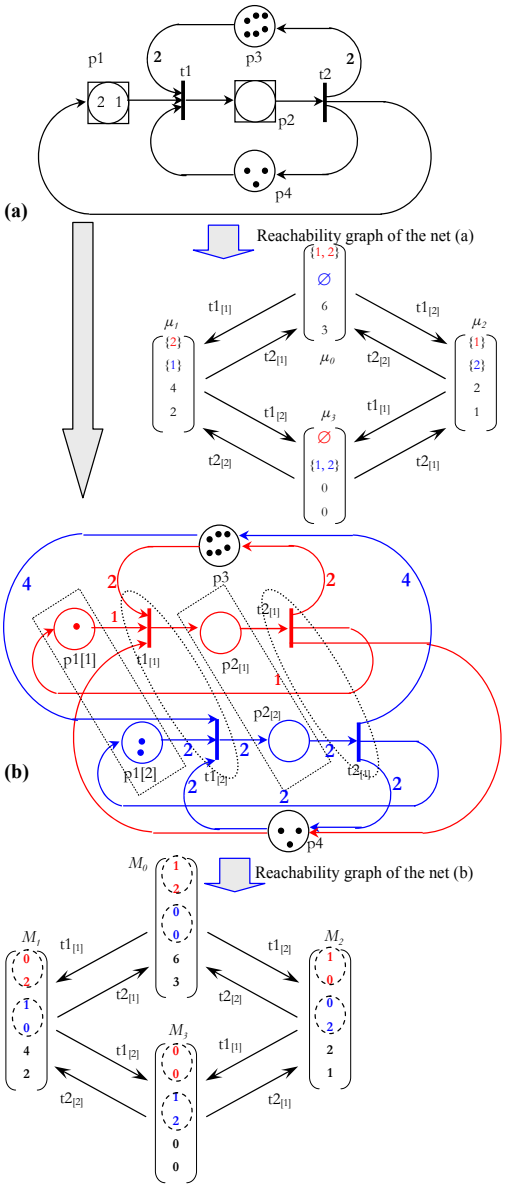


Figure 3: Transformation of a BDSPN (general case).

By analogy with the transformation procedure

for the special case, the transformation for the general case consists of the transformation of its each batch place p_i into a set of discrete places corresponding to $D(p_i)$ and the transformation of its each batch transition t_j into a set of discrete transitions corresponding to $D(t_j)$. For example, the transformation of the BDSPN given in Fig. 3 is realized by transforming the batch transition t_1 (resp. t_2) into a set of discrete transitions $\{t_{1[1]}, t_{1[2]}\}$ (resp $\{t_{2[1]}, t_{2[2]}\}$) and by transforming the batch place p_1 (resp. p_2) into a set of discrete places $\{p_{1[1]},p_{1[2]}\}$ (resp. $\{p_{2[1]}, p_{2[2]}\}$ as shown in Fig. 3b. Similar to the special case, to respect the dynamical behaviour of the BDSPN, each possible batch firing index of each batch transition is integrated in the weights of the input and output arcs of the corresponding transition in the resulting classical net. After a close look of the reachability graphs of the two nets, we find that the two nets have the same behaviour. As illustrated in the figure, each μ-marking μ_i of the BDSPN corresponds to the marking M_i of the resulting classical Petri net. The M-marking of each batch place p_i is expressed by its corresponding set of discrete places $D(p_i)$. The transformation procedure for the general case is outlined in the following.

Transformation procedure (general case)

Step1. The set of discrete places P_d of the BDSPN and their markings remain unchanged for the DPN.

$$p_i \in P_d,\ M_0(p_i) = \mu_0(p_i) \tag{19}$$

Step2. Each batch place p_i of the BDSPN is converted into a set of discrete places $D(p_i)$ in the DPN such as:

$$\begin{aligned} &D(p_i) = \{p_{i[b]} \,|\, b \in D(p_i)\} \text{ and} \\ &\forall p_{i[b]} \in D(p_i),\ M_0\left(p_{i[b]}\right) = \sum_{l \in \mu(p_i) \text{ and } l=b} l \end{aligned} \tag{20}$$

Step3. Each batch transition t_j of the BDSPN is converted into a set of discrete transitions $D(t_j)$ in the DPN such that:

$$D(t_j) = \{t_{j[q]} \,|\, t_{j[q]} \in D(t_j)\} \tag{21}$$

The set of discrete transitions T_b of the BDSPN remains unchanged for the DPN.

Step4. Each place $p_{i[b]} \in D(p_i)$ is connected to the output transitions $(p_{i[b]})^\bullet$ such that:

$$\begin{aligned} &\forall p_{i[b]} \in D(p_i), (p_{i[b]})^\bullet \\ &= \{t_{j[q]} \,|\, t_j \in p_i^\bullet \text{ and } q = b / W(p_i, t_j)\}. \end{aligned} \tag{22}$$

$$\begin{aligned} &\forall p_{i[b]} \in D(p_i),\ \forall t_{j[q]} \in (p_{i[b]})^\bullet \\ &W(p_{i[b]}, t_{j[q]}) = W(p_i, t_j) \times b. \end{aligned} \tag{23}$$

Step5. Each transition $t_{j[q]} \in D(t_j)$ is connected to the output places $(t_{j[q]})^\bullet$ such that:

$$\begin{aligned} &\forall t_{j[q]} \in D(t_j), (t_{j[q]})^\bullet = \\ &\{p_{i[b]} \,|\, (p_{i[b]} \in D(p_i)), (p_i \in t_j^\bullet \cap P_d) \\ &\text{and } (q = b / W(p_i, t_j))\} \\ &\cup \{p_i \,|\, p_i \in t_j^\bullet \cap P_d\}. \end{aligned} \tag{24}$$

The weights of the corresponding arcs are given by:

$$\begin{aligned} &\forall t_{j[q]} \in D(t_j), \forall (p_i \vee p_{i[b]}) \in (t_{j[q]})^\bullet, \\ &W(t_{j[q]}, p_{i[b]}) = q \times W(t_j, p_i). \end{aligned} \tag{25}$$

Step6. Each place $p_{i[b]} \in D(p_i)$ is connected to the input transitions $^\bullet\left(p_{i[b]}\right)$ such that:

$$\begin{aligned} &\forall p_{i[b]} \in D(p_i),\ {}^\bullet(p_{i[b]}) = \\ &\{t_{j[q]} \,|\, t_j \in {}^\bullet p_i \text{ and } q = b / W(t_j, p_i)\} \\ &\cup \{t_j \in ({}^\bullet p_i \cap P_d)\}. \end{aligned} \tag{26}$$

The weights of the corresponding arcs are given by:

$$\begin{aligned} &\forall p_{i[b]} \in D(p_i),\ \forall (t_j \vee t_{j[q]}) \in {}^\bullet(p_{i[b]}) \\ &W(t_{j[q]}, p_{i[b]}) = q \times W(t_j, p_i). \end{aligned} \tag{27}$$

Step7. Each transition $t_{j[q]} \in D(t_j)$ will be connected to the set $^\bullet(t_{j[q]})$ of input places such that:

$$\begin{aligned} &\forall t_{j[q]} \in D(t_j), {}^\bullet(t_{j[q]}) = \\ &\{p_{i[b]} \,|\, (p_i \in {}^\bullet t_j \cap P_d) \text{ and } (q = b / W(p_i, t_j))\} \\ &\cup \{p_i \,|\, p_i \in {}^\bullet t_j \cap P_d\}. \end{aligned} \tag{28}$$

The weights of the corresponding arcs are given by:

$$\begin{aligned} &\forall t_{j[q]} \in D(t_j), \forall (p_i \vee p_{i[b]}) \in {}^\bullet(t_{j[q]}), \\ &W(p_{i[b]}, t_{j[q]}) = q \times W(p_i, t_{j[q]}). \end{aligned} \tag{29}$$

Step8. The arcs which connect discrete places with discrete transitions in the BDSPN and their weights remain unchanged in the DPN.

3.3 Case with Inhibitor Arcs

The transformation is also possible for BDSPNs with inhibitor arcs whose weights are constant. We will illustrate it by using some examples.

Sub-case 1. As shown in the net depicted in Fig. 4a, in the case where there is an inhibitor arc connecting a discrete place p_i to a batch transition t_j, the corresponding inhibitor condition must be

reproduced in the resulting classical Petri net for all q-indexed transitions $t_{j[q]}$ generated by the batch transition t_j. Clearly, in this example, the batch transition t_1 can be fired with three possible batch firing indexes during the evolution of the net. In other words, the transition t_1 generates three possible q-indexed transitions $t_{1[1]}$, $t_{1[2]}$, $t_{1[3]}$. Thus, in the corresponding classical Petri net there are three inhibitor arcs which connect the discrete place p_2 to the three q-indexed transitions, respectively. It is easily to observe that the two nets are identical in terms of their dynamical behaviours.

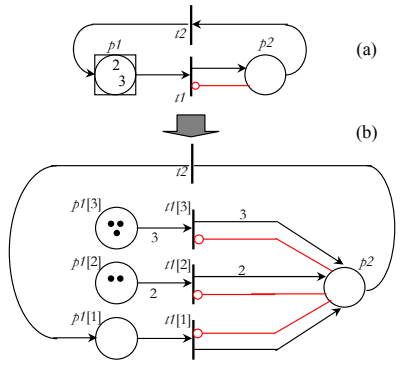


Figure 4: Transformation of a BDSPN with inhibitor arc.

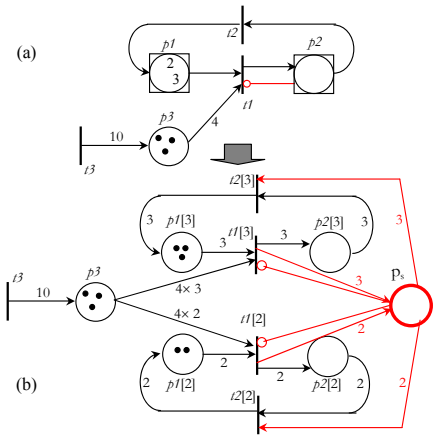


Figure 5: Transformation of a BDSPN with inhibitor arc.

Sub-case 2. We now consider the case as shown in Fig. 5.a where there is an inhibitor arc connecting a batch place to a transition. The enabling of the transition t_1 for a given batch firing index q in the net (a) must satisfy the condition $M(p_2) < w(t_1, p_2)$ imposed by the inhibitor arc. After the transformation of each batch place (resp. batch transition) into a set of discrete places (resp. a set of transitions), we observe that to respect the enabling condition imposed by the inhibitor arc in the net (a), it is necessary to capture the total marking of the discrete places generated by the batch place p_2 by using a supplementary place p_s in the classical Petri net.

3.4 Case of the Temporal Model

The transformation techniques discussed so far do not consider temporal and/or stochastic elements in a BDSPN, but they can be adapted for the BDSPN model with timed and/or stochastic transitions. The basic idea is as follows: Each discrete transition in the BDSPN model keeps its nature (immediate, deterministic, stochastic) in the resulting classical Petri net. The q-indexed transition $t_{j[q]}$ which may be generated by each batch transition t_j has the same nature as the transition t_j. Other elements of the BDSPN model may also be taken into account in the resulting classical model such as the execution policies; the priorities of some transitions; etc.

4 NECESSITY OF THE MODEL

In this section, the necessity of the introduction of the BDSPN model is demonstrated through the analysis of the transformation procedures presented in the previous section. The advantages of the model are discussed in two cases: the case where a BDSPN can be transformed into a classical Petri net and the case where the transformation is impossible.

Case 1. The BDSPN model is transformable: In the case where the transformation is possible, the advantages of the BDSPN model are outlined in the following: **(a)** As shown in the transformation procedures developed in the section 4, we note that the resulting classical Petri net depends on the initial μ-marking of the BDSPN. Obviously, if we change the initial μ-marking of the BDSPN given in Fig. 3.a, we will obtain another classical Petri net. For example, if there is another batch token of different size in the batch place p_1, all the structure of the corresponding classical Petri net must be changed. In fact, the batch places of the BDSPN may not generate the same set of q-indexed transitions $D(t_j)$ for each batch transition t_j and may not generate the same set of discrete places $D(p_i)$ for each batch place p_i during the evolution of the net. **(b)** The transformation of a given BDSPN model into an equivalent classical Petri net may lead to a very large and complex structure. According to the transformation procedure developed in subsection 3.2, the number of places $|P^*|$ and the number of transitions $|T^*|$ in the equivalent classical Petri net are given by:

$$\left|P^*\right| = \left|P_d\right| + \sum_{i=1}^{|P_b|} \left|D(p_i)\right| \text{ and } \left|T^*\right| = \left|T_d\right| + \sum_{j=1}^{|T_b|} \left|D(t_j)\right| \quad (30)$$

where $|P_b|$ is the number of the batch places; $|P_d|$ is the number of the discrete places; $|T_b|$ is the number of the batch transitions; $|T_d|$ is the number of the discrete transitions of the given BDSPN. $D(t_j)$ is the set of q-indexed transitions generated by each batch transition $t_j \in T_b$ and $D(p_i)$ is the set of all possible batch tokens which appear in each batch place $p_i \in P_b$ during the evolution of the BDSPN.

Case 2. The BDSPN is not transformable: The modelling of some discrete event systems such as inventory control systems and logistical systems, as shown in (Labadi, et al., 2005, 2007; Chen, et al. 2005), require the use of the BDSPN model with variables arc weights depending on its M-marking and possibly on some decision parameters of the systems. It is the case of the BDSPN model of an inventory control system whose inventory replenishment decision is based on the inventory position of the stock considered and the reorder and order-up-to-level parameters (see Fig. 6). The modelling of such a system is possible by using a BDSPN model with variables arc weights depending on its M-marking. The BDSPN model shown in Fig. 6 represents an inventory control system where its operations are modelled by using a set of transitions: generation of replenishment orders (*t3*); inventory replenishment (*t2*); and order delivery (*t1*) that are performed in a batch way because of the batch nature of customer orders represented by batch tokens in batch place *p4* and the batch nature of the outstanding orders represented by batch tokens in batch place *p3*. In the model, the weights of the arcs (*t3*, *p2), (t3, p3)* are variable and depend on the parameters s and S of the system and on the M-marking of the model (*S-M(p2)+M(p4)*; *s+M(p4)*). The model may be built for the optimization of the parameters *s* and *S*. In this case, the techniques for the transformation of the BDSPN model into an equivalent classical Petri net model proposed in the previous section is not applicable. In fact, contrary to the example given in Fig. 3, in this model, the sizes of the batch tokens that may be generated depend on both the initial μ-marking of the model and the parameters *s* and *S*. In other words, a change of the decision parameters *s* and *S* of the system or the initial μ-marking of the model will lead to another way of the evolution of the discrete quantities. Moreover, the appearance of stochastic transitions in the model makes more difficult to characterize all possible sizes of the batch tokens that are necessary to be known for the application of the transformation methods.

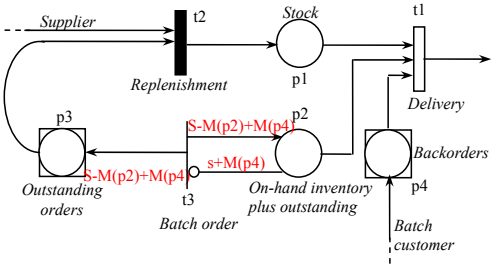


Figure 6: BDSPN model of an inventory control system.

5 CONCLUSION

The work of this paper has contributed to the structural analysis of batch deterministic and stochastic Petri nets (BDSPNs). Several procedures for the transformation of the model into an equivalent classical Petri net are developed. It is shown that such a transformation is possible for some cases but impossible for the model with variable arc weights depending on its marking. In this study, relationships between BDSPNs and classical discrete Petri nets are established and the advantages of introducing the BDSPN model are demonstrated. The capability of the BDSPN model to meet real needs is shown through industrial applications in our previous papers.

REFERENCES

Chen, H., Amodeo, L., Chu, F., and Labadi, K., "Performance evaluation and optimization of supply chains modelled by Batch deterministic and stochastic Petri net", *IEEE transactions on Automation Science and Engineering, pp. 132-144, 2005.*

Labadi, K., Chen, H., Amodeo, L., "Modeling and Performance Evaluation of Inventory Systems Using Batch Deterministic and Stochastic Petri Nets", *to appear in IEEE Transactions on Systems, Man, and Cybernetics – Part C, 2007.*

Labadi, K., Chen, H., Amodeo, L., "Application des BDSPNs à la Modélisation et à l'Evaluation de Performance des Chaînes Logistiques", *Journal Européen des Systèmes Automatisés, pp. 863-886, n° 7, 2005.*

Lindemann, C., "Performance Modelling with Deterministic and Stochastic Petri Nets", *John Wiley and Sons, 1998.*

Marsan A. M., and Chiola G., "On Petri nets with deterministic and exponentially distributed firing times", Lecture Notes in Computer Science, vol. 266, pp. 132-145, Springer-Verglag, 1987.

STATE ESTIMATION OF NONLINEAR DISCRETE-TIME SYSTEMS BASED ON THE DECOUPLED MULTIPLE MODEL APPROACH

Rodolfo Orjuela, Benoît Marx, José Ragot and Didier Maquin
Centre de Recherche en Automatique de Nancy, UMR 7039, Nancy-Université, CNRS
2, Avenue de la Forêt de Haye, 54 516 Vandœuvre-lès-Nancy, France
{*rodolfo.orjuela, benoit.marx, jose.ragot, didier.maquin*}*@ensem.inpl-nancy.fr*

Keywords: State estimation, nonlinear discrete-time systems, multiple model approach, decoupled multiple model.

Abstract: Multiple model approach is a powerful tool for modelling nonlinear systems. Two structures of multiple models can be distinguished. The first structure is characterised by decoupled submodels, i.e. with no common state (*decoupled multiple model*), in opposition to the second one where the submodels share the same state (*Takagi-Sugeno multiple model*). A wide number of research works investigate the state estimation of nonlinear systems represented by a classic Takagi-Sugeno multiple model. On the other hand, to our knowledge, the state estimation of the decoupled multiple model has not been investigated extensively. This paper deals with the state estimation of nonlinear systems represented by a decoupled multiple model. Conditions for ensuring the convergence of the estimation error are formulated in terms of a set of Linear Matrix Inequalities (LMIs) employing the Lyapunov direct method.

1 INTRODUCTION

Highly nonlinear processes are commonly encountered in practical engineering problems (chemistry, mechanic, hydraulic, electrotechnics, etc). An accurate model with a simple structure, preferably linear, is often necessary for designing a control law or setting up a diagnosis strategy using conventional control tools. Building only one model, valid in whole operating space of the system, is not always possible due, for example, to the change of the dynamic behaviour in the operating space. Hence, the operating space of the system is often limited before the identification stage (local modelling).

New techniques of identification have been developed for modelling the overall behaviour of the process (global modelling). One of these techniques is based on the decomposition of the operating space of the system into a finite number of operating zones. Each operating zone is characterised by a submodel that has a simple structure. According to the zone where the nonlinear system evolves, the output y_i of each submodel is more or less requested in order to describe the global behaviour y of the nonlinear system, that is to say:

$$y(k) = \sum_{i=1}^{L} \mu_i(k) y_i(k), \qquad (1)$$

where the i^{th} submodel contribution depends on the *weighting function* μ_i. A wide number of identification techniques based on this same principle can be distinguished: piecewise linear model, radial basis function networks, fuzzy models, multiple models, etc.

In this communication, we tackle the multiple model approach. Classically, the multiple model is built using linear submodels associated with weighting functions that ensure a smooth blend between the submodels. It is important to note that the multiple models are considered as an *universal approximation tool* of nonlinear systems (Johansen et al., 2000). Hence, it is possible to apply the available tools for linear systems to nonlinear systems represented by a multiple model.

In (Filev, 1991) two possible interpretations of equation (1) have been investigated in a fuzzy modelling framework (these interpretations will be directly related to multiple model). In the first interpretation, the submodels are decoupled and their state vector is different (*decoupled multiple model*); in the second one, the submodels have the same state vector (*Takagi-Sugeno multiple model*).

The second interpretation has been widely popularized and many works deal with the identification and analysis (control, state estimation, diagnosis, etc.) of nonlinear systems represented by this class of mul-

tiple model.

By comparison with the Takagi-Sugeno multiple model, the decoupled multiple model has been less investigated. Some works in control domain (Gawthrop, 1995; Gatzke and Doyle III, 1999; Gregorcic and Lightbody, 2000) and in identification (Venkat et al., 2003) of nonlinear systems have employed successfully this structure and shown its relevance. However, to our knowledge the state estimation problem has not been investigated.

In this paper, a new method for designing a state estimator of nonlinear discrete-time systems represented by a decoupled multiple model is presented. The paper starts with section 2 that introduces two multiple model structures according to the selected interpretation. Stability of decoupled multiple model is investigated in section 3. In section 4, sufficient conditions (in LMIs terms) are established in order to ensure the asymptotic convergence of the estimation error. Finally, section 5 presents an academic example of state estimation of a decoupled multiple model.

2 MULTIPLE MODEL STRUCTURES

The interconnection of the submodels can be performed with various structures in order to generate the global output of the multiple model. Two essential structures of multiple models can be distinguished whether the same state vector appears in all submodels or not.

Concerning the identification step, there exists different techniques (linearisation, parametric optimisation) for the parameter estimation of the submodels for a particular multiple model structure. See (Murray-Smith and Johansen, 1997; Gasso et al., 2001; Venkat et al., 2003) and the references therein for further information about these techniques.

2.1 Takagi-Sugeno Multiple Model

The Takagi-Sugeno multiple model structure is conventionally employed in multiple model analysis and synthesis (Murray-Smith and Johansen, 1997). This multiple model has the following structure:

$$\begin{aligned} x_i(k+1) &= A_i x(k) + B_i u(k), \\ x(k+1) &= \sum_{i=1}^{L} \mu_i(\xi(k)) x_i(k+1), \\ y(k) &= \sum_{i=1}^{L} \mu_i(\xi(k)) C_i x(k), \end{aligned} \tag{2}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ the input and $y \in \mathbb{R}^p$ the output vector. For the i^{th} submodel, $A_i \in \mathbb{R}^{n\times n}$ is the system matrix, $B_i \in \mathbb{R}^{n\times m}$ the input matrix and $C_i \in \mathbb{R}^{p\times n}$ the output matrix. The μ_i are the weighting functions with the following properties:

$$\sum_{i=1}^{L} \mu_i(\xi(k)) = 1, \qquad \forall k \tag{3a}$$

$$0 \le \mu_i(\xi(k)) \le 1 \quad \forall i = 1...L, \forall k \tag{3b}$$

ξ is the decision variable that depends, for example, on the measurable state variable and/or input or output of the system.

From equation (2), one can see that in the Takagi-Sugeno multiple model there is a common state x that couples all submodel states x_i. Therefore the dimension of the state vectors must be identical for all the submodels.

2.2 Decoupled Multiple Model

Another possible structure using a parallel interconnection of the submodels is proposed in (Filev, 1991). Here, this structure is slightly modified using a state representation as follows:

$$\begin{aligned} x_i(k+1) &= A_i x_i(k) + B_i u(k), \\ y_i(k) &= C_i x_i(k) \\ y(k) &= \sum_{i=1}^{L} \mu_i(\xi(k)) y_i(k), \end{aligned} \tag{4}$$

where $x_i \in \mathbb{R}^{n_i}$ and $y_i \in \mathbb{R}^p$ are, respectively, the state vector and the output vector for the i^{th} submodel and where u, y, ξ, $A_i \in \mathbb{R}^{n_i\times n_i}$, $B_i \in \mathbb{R}^{n_i\times m}$ et $C_i \in \mathbb{R}^{p\times n_i}$ have been defined in the previous section.

It should be noted that the global output of the multiple model is given by a weighted sum of the submodel outputs. The blending between the submodels is made through the static equation. Therefore each submodel evolves independently in its own state space according to the input control and its initial state.

It is obvious that the principal interest of this structure is the decoupling between the submodels. Indeed, in contrast to the Takagi-Sugeno multiple model, in the decoupled multiple model the dimension of the state vector x_i of each submodel can be different (of course the output vector dimension must be identical). Therefore, this structure is well adapted for modelling strongly nonlinear systems whose structure varies with the operating zone.

Notation: *The following notations will be used all along this paper. $P > 0$ ($P < 0$) means P is a positive (negative) definite matrix; P^T denotes the transpose of P. We shall simply write $\mu_i(\xi(k)) = \mu_i(k)$.*

3 STABILITY ANALYSIS

It is possible to rewrite the equations (4) using an augmented state vector as follows:

$$\begin{aligned} x(k+1) &= \tilde{A}x(k)+\tilde{B}u(k), \\ y(k) &= \tilde{C}(k)x(k), \end{aligned} \tag{5}$$

where:

$$\tilde{A}=\begin{bmatrix} A_1 & 0 & 0 & 0 & 0 \\ & \ddots & 0 & 0 & 0 \\ 0 & 0 & A_i & 0 & 0 \\ & & & \ddots & 0 \\ 0 & 0 & 0 & 0 & A_L \end{bmatrix}, \tilde{B}=\begin{bmatrix} B_1 \\ \vdots \\ B_i \\ \vdots \\ B_L \end{bmatrix}, \tilde{C}(k)=\begin{bmatrix} \mu_1(k)C_1 \\ \vdots \\ \mu_i(k)C_i \\ \vdots \\ \mu_L(k)C_L \end{bmatrix}^T$$

and $x(k)=\begin{bmatrix} x_1(k) & \cdots & x_i(k) & \cdots & x_L(k) \end{bmatrix}^T \in \mathbb{R}^n$, $n=\sum_{i=1}^{L} n_i$.

Comments

- The matrices $\tilde{A}$ and $\tilde{B}$ are partitioned block matrices.
- The output matrix $\tilde{C}(k)$ is a partitioned blocks matrix whose parameters vary with time. Indeed, the weighting functions $\mu_i(k)$ only affect the submodel outputs.

The stability of a decoupled multiple model can be easily established by analysing the eigenvalues of the matrix $\tilde{A}$. Notice that the matrix $\tilde{A}$ is a block diagonal matrix. Therefore, all eigenvalues of this matrix are inside the unit circle if and only if all eigenvalues of every matrices A_i are inside the unit circle.

To sum up, a decoupled multiple model is stable if and only if all submodels are stable, in contrast to Takagi-Sugeno multiple model where the stability depends not only on the stability of the submodels but also on the weighting function values. In the sequel, the multiple model is assumed to be stable.

4 STATE ESTIMATION

State estimation of Takagi-Sugeno multiple model has been widely investigated in a stabilisation law control design perspective (Tanaka and Sugeno, 1990; Feng et al., 1997; Chadli et al., 2003; Guerra and Vermeiren, 2004). Indeed, most of the used control techniques needs the state vector knowledge which is not in general fully measurable.

The classically used state estimator is an extension of the proportional (Luenberger) observer. However, some other classes of state estimators have been developed, for example, sliding mode observers (Palm and Bergstern, 2000) and unknown input observers (Akhenak et al., 2004).

The Lyapunov second method is typically employed in order to establish the sufficient convergence conditions of the estimation error in terms of a set of Linear Matrix Inequalities (LMIs) (Boyd et al., 1994) which are solved using standard convex optimization algorithms.

State estimation of decoupled multiple model has been partially explored in a self-tuning control law design perspective in (Gawthrop, 1995). Thanks to the decoupling between the submodels, the observer gains can be classically calculated under the assumption that the submodel outputs are known. However, in our case, it is assumed that this information is not available. Therefore, this approach cannot be used here.

The aim of this section is to investigate the state estimation of a decoupled multiple model, using only the measurable signals i.e. the input and the output of the system (the outputs of the submodels are not available). It is important to notice that the design of the observer must take into account the blending between the outputs of the submodels in order to guarantee the convergence of the estimation error.

4.1 Observer Structure

The proportional gain observer for the decoupled multiple model is given by:

$$\begin{aligned} \hat{x}_i(k+1) &= A_i\hat{x}_i(k)+B_iu(k)+K_i(y(k)-\hat{y}(k)), \\ \hat{y}_i(k) &= C_i\hat{x}_i(k), \\ \hat{y}(k) &= \sum_{i=1}^{L}\mu_i(k)\hat{y}_i(k), \end{aligned} \tag{6}$$

where $\hat{x}_i \in \mathbb{R}^{n_i}$ is the state estimation for the i^{th} submodel, $y(k)$ the output of the multiple model, $\hat{y}(k)$ the output estimation and $K_i \in \mathbb{R}^{n_i \times p}$ the gain of the i^{th} observer. Equation (6) can be written in a compact form using the partitioned matrices (5):

$$\begin{aligned} \hat{x}(k+1) &= A_{obs}(k)\hat{x}(k)+\tilde{B}u(k)+\tilde{K}y(k), \\ \hat{y}(k) &= \tilde{C}(k)\hat{x}(k), \end{aligned} \tag{7}$$

$$\text{where} \quad \tilde{K} = \begin{bmatrix} K_1 & \cdots & K_i & \cdots & K_L \end{bmatrix}^T, \tag{8}$$

$$A_{obs}(k) = \tilde{A}-\tilde{K}\tilde{C}(k). \tag{9}$$

Note that the matrix $A_{obs}(k)$ may be decomposed as follows:

$$A_{obs}(k) = \sum_{i=1}^{L}\mu_i(k)\Phi_i, \tag{10}$$

$$\Phi_i = \tilde{A}-\tilde{K}\tilde{C}_i, \tag{11}$$

where $\tilde{C}_i$ is the following partitioned block matrix:

$$\tilde{C}_i = \begin{bmatrix} 0 & \ldots & C_i & \ldots & 0 \end{bmatrix}. \tag{12}$$

The design of the observer consists in determining the gain $\tilde{K}$ such that the estimation error given by:

$$e(k) = x(k)-\hat{x}(k), \tag{13}$$

converges asymptotically to zero for an arbitrary blending between the submodel outputs.

4.2 Estimation Error Convergence

Here, the second Lyapunov method is used to investigate the estimation error convergence by means of a quadratic Lyapunov function. It is clear that other Lyapunov functions can be considered (see section 4.4). The following Theorem gives a sufficient condition for ensuring the estimation error convergence.

Theorem 1. *Consider the decoupled multiple model* (4) *and the observer* (6)*. The asymptotic convergence towards zero of the estimation error is guaranteed if there exists a symmetric and positive definite matrix P and a matrix G such that:*

$$\begin{bmatrix} P & \tilde{A}^T P - \tilde{C}_i^T G^T \\ P\tilde{A} - G\tilde{C}_i & P \end{bmatrix} > 0,\ i = 1...L, \quad (14)$$

where the observer gain is deduced from $\tilde{K} = P^{-1}G$.

Proof. Let us consider the following quadratic Lyapunov function:

$$V(e(k)) = e^T(k)Pe(k), \quad P = P^T \text{ and } P > 0. \quad (15)$$

The variation of the above function is given by:

$$\Delta V(e(k)) \quad = \quad V(e(k+1)) - V(e(k)), \quad (16)$$

$\Delta V(e(k))$ must be negative in order to ensure its decrease and the asymptotic error convergence also. Considering the dynamics of the estimation error given by:

$$e(k+1) \quad = \quad A_{obs}(k)e(k), \quad (17)$$

and substituting (15) and (17) into (16), then $\Delta V(e(k))$ becomes:

$$\Delta V(e(k)) = e^T(k)\{A_{obs}^T(k)PA_{obs}(k) - P\}e(k), \quad (18)$$

that is a quadratic form in $e(k)$. Therefore, a *necessary and sufficient condition* for ensuring $\Delta V(e(k)) < 0$ is:

$$A_{obs}^T(k)PA_{obs}(k) - P < 0, \quad \forall k. \quad (19)$$

By considering (10), the inequality (19) can be rewritten as:

$$\sum_{j=1}^{L} \mu_j(k)\Phi_j^T P P^{-1} P \sum_{i=1}^{L} \mu_i(k)\Phi_i - P < 0, \quad (20)$$

Combining the Schur complement with property (3a) of the weighting functions, it is possible to write:

$$\sum_{i=1}^{L} \mu_i(k) \begin{bmatrix} P & \Phi_i^T P \\ P\Phi_i & P \end{bmatrix} \quad > \quad 0. \quad (21)$$

The inequality (21) can be upper bounded using the property (3b) of the weighting functions. Finally, a *sufficient condition* that ensures the error convergence is given by:

$$\begin{bmatrix} P & \Phi_i^T P \\ P\Phi_i & P \end{bmatrix} \quad > \quad 0 \quad i = 1...L \quad (22)$$

and substituting (11) for Φ_i, we obtain:

$$\begin{bmatrix} P & (\tilde{A} - \tilde{K}\tilde{C}_i)^T P \\ P(\tilde{A} - \tilde{K}\tilde{C}_i) & P \end{bmatrix} > 0,\ i = 1...L. \quad (23)$$

These matrix inequalities are nonlinear in $\tilde{K}$ and P. Therefore, it is not possible to solve them directly using classical LMI tools. The following change of variables $G = P\tilde{K}$ allows the linearisation of this problem and ends the demonstration of Theorem 1. □

4.3 Eigenvalue Placement

In order to enforce dynamic performances of the observer (for example, the damping and the estimation error decay rate) the eigenvalue placement of the observer must be investigated. In (Chilali and Gahinet, 1996) a general characterization for eigenvalues clustering in subregions of the complex plan in terms of LMIs is proposed.

The eigenvalues of the matrix X are placed inside the circle with radius R and centred at $(q,0)$ in the z plan if the following LMI is feasible:

$$\begin{bmatrix} -RP & -qP + XP \\ -qP + (XP)^T & -RP \end{bmatrix} \quad < \quad 0, \quad (24)$$

where P is a symmetric and positive definite matrix. Let us notice that if $R = 1$ and $q = 0$ then we obtain the stability condition for linear discrete-time systems.

In order to place the eigenvalues of the observer, the LMIs of Theorem 1 are modified as follows.

Theorem 2. *Consider the decoupled multiple model* (4) *and the observer* (6)*. The eigenvalues of the observer are placed inside the circle with radius R and centred at* $(q,0)$ *if there exists a symmetric and positive definite matrix P and a matrix G such that:*

$$\begin{bmatrix} -RP & -qP + \tilde{A}^T P - \tilde{C}_i^T G^T \\ -qP + P\tilde{A} - G\tilde{C}_i & -RP \end{bmatrix} < 0, \quad (25)$$

for $i = 1...L$*, where the observer gain is given by* $\tilde{K} = P^{-1}G$.

It is clear that this Theorem coincides with Theorem 1 if $R = 1$ and $q = 0$. In order to avoid strong oscillations of the estimation error, the real part of the eigenvalues of the observer are placed in the positive zone of the unit circle and their imaginary part must be reduced. A judicious choice of the radius R and the centre $(q,0)$, for example $q = 0.5$ and $R = 0.45$, allows an appropriate placement of the eigenvalues of the observer.

4.4 Relaxed Convergence Conditions

The asymptotic convergence conditions of the estimation error, presented in the previous section, depend on the existence of the common matrix P which satisfies a set of LMIs. In general, when the multiple model has a large number of submodels, the matrix P cannot be found.

In order to reduce the conservatism of the conditions obtained with a quadratic Lyapunov function, new candidate Lyapunov functions called *nonquadratic functions* have been proposed. A wide number of published works show the efficient relaxation of the stability conditions provided by this class of functions for a continuous time Takagi-Sugeno multiple model (Jadbabaie, 1999; Rhee and Won, 2006) and also in the discrete time case (Guerra and Vermeiren, 2004). The following Theorem gives a sufficient condition for ensuring the estimation error convergence using a nonquadratic function.

Theorem 3. *Consider the decoupled multiple model* (4) *and the observer* (6)*. The asymptotic convergence towards zero of the estimation error is guaranteed if there exists symmetric and positive definite matrices P_i and P_j and a some matrix M and G such that:*

$$\begin{bmatrix} P_i & (M\tilde{A}-G\tilde{C}_i)^T \\ M\tilde{A}-G\tilde{C}_i & M+M^T-P_j \end{bmatrix} > 0 \; \forall i,j = 1...L, \quad (26)$$

where the observer gain is deduced from $\tilde{K} = M^{-1}G$.

Proof. The considered nonquadratic Lyapunov function is given by:

$$V(e(k)) = e^T(k)\sum_{i=1}^{L}\mu_i(k)P_i e(k) = e^T(k)P(k)e(k), \quad (27)$$

where $P_i = P_i^T$ and $P_i > 0$. The convergence error analysis is performed as in the previous case. A *necessary and sufficient condition* in order to ensure the error convergence is given by:

$$A_{obs}^T(k)P(k+1)A_{obs}(k) - P(k) \quad < \quad 0. \quad (28)$$

Introducing (10) and using the Schur complement, the above inequality becomes:

$$\sum_{j=1}^{L}\sum_{i=1}^{L}\mu_i(k)\mu_j(k+1)\begin{bmatrix} P_i & \Phi_i^T P_j \\ P_j\Phi_i & P_j \end{bmatrix} > 0. \quad (29)$$

Using property (3b) of the weighting functions and substituting (11) for Φ_i, one obtains the following *sufficient condition* that ensures the asymptotic convergence of the estimation error:

$$\begin{bmatrix} P_i & (\tilde{A}-\tilde{K}\tilde{C}_i)^T P_j \\ P_j(\tilde{A}-\tilde{K}\tilde{C}_i) & P_j \end{bmatrix} > 0, \quad i,j = 1...L. \quad (30)$$

Inequalities (30) are nonlinear matrix inequalities in $\tilde{K}$, P_i and P_j. In contrast to the quadratic case, there is not variable change that allows the direct linearisation of this problem. However, the results coming from (De Oliveira et al., 1999) (Theorem 2) help to rewrite the inequalities (30) as follows:

$$\begin{bmatrix} P_i & (M(\tilde{A}-\tilde{K}\tilde{C}_i))^T \\ M(\tilde{A}-\tilde{K}\tilde{C}_i) & M+M^T-P_j \end{bmatrix} > 0, \quad i,j = 1...L,$$

where M is not constrained to be symmetric ($M \neq M^T$). After this transformation, the linearisation of the above inequalities can be effectively yielded by using the change of variables $G = M\tilde{K}$. Hence, the proof of Theorem 3 is completed. □

Notice that Theorem 1 is encompassed by Theorem 3. Indeed, if one sets $P_i = P_j = M = P$ then the Theorem 3 coincides with the Theorem 1. Therefore, the previous result is less conservative than the condition obtained with a conventional quadratic function.

5 EXAMPLE

Let us consider the state estimation of the decoupled multiple model with $L = 3$ submodels. The numerical matrices A_i, B_i and C_i are:

$$A_1 = \begin{bmatrix} 0.8 & 0 \\ 0.4 & 0.1 \end{bmatrix}, A_2 = \begin{bmatrix} -0.3 & -0.5 & 0.2 \\ 0.7 & -0.8 & 0 \\ -2 & 0.1 & 0.7 \end{bmatrix}, A_3 = \begin{bmatrix} -0.5 & 0.1 \\ -0.6 & -0.5 \end{bmatrix},$$
$$B_1 = \begin{bmatrix} 0.2 & -0.4 \end{bmatrix}^T, B_2 = \begin{bmatrix} 0.7 & -0.5 & 0.3 \end{bmatrix}^T, B_3 = \begin{bmatrix} -0.2 & 0 \end{bmatrix}^T,$$
$$C_1 = \begin{bmatrix} 0.7 & 0 \\ 0.5 & 0.2 \end{bmatrix}, C_2 = \begin{bmatrix} 0.5 & 0 & 0.8 \\ 0.7 & 0.2 & 0.1 \end{bmatrix}, C_3 = \begin{bmatrix} 0.9 & 0.3 \\ -0.6 & 0 \end{bmatrix}.$$

Here, the decision variable ξ is the input signal $u(k) \in [0,1]$. The weighting functions are obtained from normalised Gaussian function:

$$\mu_i(u(k)) \quad = \quad \omega_i(u(k))/\sum_{j=1}^{L}\omega_j(\xi(k)), \quad (31)$$

$$\omega_i(u(k)) \quad = \quad \exp\left(-(u(k)-c_i)^2/\sigma^2\right), \quad (32)$$

with the standard deviation $\sigma = 0.4$ and the centre $c_i = [0.1, 0.5, 0.9]$. The eigenvalues of the matrix $\tilde{A}$ are inside the unit circle, thus the multiple model is stable. Using Theorem 3, we obtain the following observer gain:

$$\tilde{K} \quad = \quad \begin{bmatrix} 0.041 & 0.020 & 0.160 & 0.190 & 0.221 & -0.090 & -0.181 \\ 0.194 & 0.113 & -0.299 & -0.044 & -0.701 & 0.172 & 0.268 \end{bmatrix}^T.$$

As can be seen in figures 1 and 2, the suggested observer provides a good output estimation. The error around the origin time is due to the different initial conditions of the multiple model and the observer.

6 CONCLUSION

A decoupled discrete time multiple observer has been presented in order to proceed to the state estimation of a class of nonlinear systems. The proposed observer is an extension of the proportional observer used in the linear observer theory.

Sufficient conditions that guarantee the asymptotic convergence of the estimation error are given in terms of a set of LMIs using a quadratic Lyapunov function. Less conservative conditions are also proposed thanks to a nonquadratic Lyapunov function. In order to illustrate the performances of the proposed observer an academic example is presented.

There are interesting prospects in control and diagnosis of nonlinear systems using this class of multiple model and observer. In particular, this observer class may be useful for setting up a diagnosis strategy for example. This task can be done with a bank of the proposed observers that produce a set of residual signals useful for sensor fault detection and isolation. In future work, the proposed approach will be extended to other observer classes as proportional integral observer or unknown input observer.

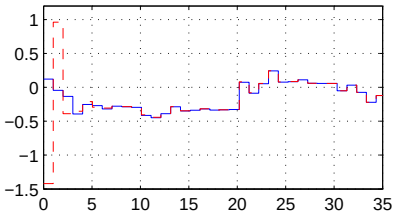


Figure 1: Output y_1 of the multiple model (solid line) and its estimated (dashed line).

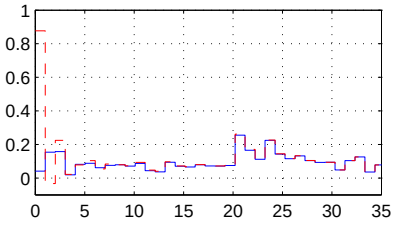


Figure 2: Output y_2 of the multiple model (solid line) and its estimated (dashed line).

REFERENCES

Akhenak, A., Chadli, M., Ragot, J., and Maquin, D. (2004). Estimation of state and unknown inputs of a nonlinear system represented by a multiple model. In *11h IFAC Symposium on Automation in Mineral and Metal processing*, Nancy, France.

Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear Matrix Inequalities in System and Control Theory*. Philadelphia, P.A., SIAM studies in applied mathematics edition.

Chadli, M., Maquin, D., and Ragot, J. (2003). Multiple observers for discrete-time multiple models. In *Safeprocess*, pages 801–806, Washington, D.C., USA.

Chilali, M. and Gahinet, P. (1996). H-infinity design with pole placement constraints: an LMI approach. *IEEE Transactions in Automatic and Control*, 41(3):358–367.

De Oliveira, M., Bernussou, J., and Geromel, J. (1999). A new discrete-time robust stability condition. *Systems Control Letters,*, 37(4):261–265.

Feng, G., Cao, S. G., Rees, N. W., and Chak, C. K. (1997). Design of fuzzy control systems with guaranteed stability. *Fuzzy Sets and Systems*, 85(1):1–10.

Filev, D. (1991). Fuzzy modeling of complex systems. *International Journal of Approximate Reasoning*, 5(3):281–290.

Gasso, K., Mourot, G., and Ragot, J. (2001). Structure identification in multiple model representation: elimination and merging of local models. In *40th IEEE Conference on Decision and Control*, pages 2992–2997, Orlando, USA.

Gatzke, E. P. and Doyle III, F. J. (1999). Multiple model approach for CSTR control. In *14 IFAC World Congress*, pages 343–348, Beijing, P. R. China.

Gawthrop, P. (1995). Continuous-time local state local model networks. In *IEEE Conference on Systems, Man & Cybernetics*, pages 852–857, Vancouver, Canada.

Gregorcic, G. and Lightbody, G. (2000). Control of highly nonlinear processes using self-tuning control and multiple/local model approaches. In *2000 IEEE International Conference on Intelligent Engineering Systems, INES 2000*, pages 167–171.

Guerra, T. M. and Vermeiren, L. (2004). LMI-based relaxed nonquadratic stabilization conditions for nonlinear systems in the Takagi-Sugeno's form. *Automatica*, 40(5):823–829.

Jadbabaie, A. (1999). A reduction in conservatism in stability and L2 gain analysis of Takagi-Sugeno fuzzy systems via linear matrix inequalites. In *14h IFAC World Congress*, pages 285–289, Beijing, P. R. China.

Johansen, T., Shorten, R., and Murray-Smith, R. (2000). On the interpretation and identification of dynamic Takagi-Sugeno fuzzy models. *IEEE Trans. on Fuzzy Systems*, 8(3):297–313.

Murray-Smith, R. and Johansen, T. (1997). *Multiple model approaches to modelling and control*. Taylor & Francis.

Palm, R. and Bergstern, P. (2000). Sliding mode observer for a Takagi-Sugeno fuzzy system. In *The Ninth IEEE International Conference on Fuzzy Systems*, volume 2, pages 665–670, San Antonio.

Rhee, B. and Won, S. (2006). A new fuzzy Lyapunov function approach for a Takagi-Sugeno fuzzy control system design. *Fuzzy Sets and Systems*, 157(9):1277–1228.

Tanaka, K. and Sugeno, M. (1990). Stability analysis of fuzzy systems using Lyapunov's direct method. In *NAFIPS*, pages 133–136, Toronto, Canada.

Venkat, A., Vijaysai, P., and Gudi, R. (2003). Identification of complex nonlinear processes based on fuzzy decomposition of the steady state space. *Journal of Process Control*, 13(6):473–488.

VERSATILE EVALUATION OF EFFECTS ON DCT-BASED LOSSY COMPRESSION OF EMG SIGNALS ON MEDICAL PARAMETERS

Tiia Siiskonen, Tapio Grönfors and Niina Päivinen
Department of Computer Science, University of Kuopio, Yliopistonranta 5, Kuopio, Finland
siiskone@hytti.uku.fi, tgronfor@messi.uku.fi, niina.paivinen@cs.uku.fi

Keywords: Lossy data compression, Electromyography, Discrete Cosine Transform and Medical parameters.

Abstract: Typically used simplified error measures, like mean-squared-error (MSE), do not reveal everything about the clinical quality of lossy compressed medical signals. Errors have to be interpreted via essential medical parameters. The medical parameters depend on the type of the signal and only the preservation of essential medical parameters can guarantee the correct clinical quality. In this study, short electromyography (EMG) signals are compressed with DCT transformation -based lossy compression method. The compression is gained with irreversible masking and scalar quantization of the DCT coefficients. The most prominent medical parameters of EMG signal are the mean frequency (MNF) and the median frequency (MDF). The behaviors of these parameters are studied both by fitting a regression line and by examining the mean absolute errors frequency-by-frequency over clinically interesting frequency range. This reveals the frequency dependency of errors of the medical parameters and inspires the idea that the generated linear model can be used for estimating the correct value of the processed medical parameter.

1 INTRODUCTION

The compression ratio, the computational efficiency of the method, and the quality of the result are the most essential features of lossy signal compression (Salomon, 2004). The quality of the result is typically characterized with mathematical, measurable error, or the distance between original and processed (compressed-decompressed) signal.

It has not been validated that simplified error, represented as mean-squared-error (MSE) (Carotti et al., 2006), signal-to-noise-ratio (SNR) (Cuerrero and Mailhes, 1997) or root-mean-squared difference (PRD) (Wellig et al., 1998), can establish the preservation of medical parameters. Only the preservation of essential medical parameters can guarantee the correct clinical quality. In spite of that fact, many medical signal compression studies rely only on simplified error measurements. However, some of the thorough studies have been concentrated on distinguishing proper medical parameters (Chan, Lovely and Hudgins, 1997; Carotti et al., 2006; Grönfors, Reinikainen and Sihvonen, 2006).

The lossy compression of electromyography (EMG) signals is not intensively studied, although the first methods have been published almost ten years ago (Cuerrero and Mailhes, 1997). Anyway, currently many EMG technologies, for example wireless measuring and archiving in patient recordings, need effective data compression. In this study, a DCT-based transformation approach has been used (Cuerrero and Mailhes, 1997; Berger et al., 2003), because of well-known algorithm with efficient implementation.

The most prominent spectral features of EMG signal are the mean frequency (MNF) and the median frequency (MDF) (Farina and Merletti, 2000; Filligoi and Felici, 1999), whose time evolution has been used for clinical assessment of EMG recordings. The simplified error represents a suggestive average estimate of the error value of the medical parameters, but it cannot be used to predict where in the dynamic range the error has been concentrated. In this study, we focus on versatile evaluation of compression effects on medical parameters. Both systematic and random errors on medical parameters are examined over these dynamic ranges.

2 MATERIALS AND METHODS

We have used real EMG recordings in this study. All the tests and simulations were done with Matlab (Versions 6.5.0.180913a Release 13 and 7.14 Release 14).

2.1 Test Signals

We have used EMG signals measured from paraspinal muscles of healthy young volunteers. The measurements and classification were done by an experienced clinical neurophysiologist. The duration of every signal was 20 seconds and they were sampled with 1 kHz sampling frequency, consisting of 20000 twelve-bit integer values measured with DCU-600 lightweight EMG system (Sihvonen et al., 2004). Each signal consists of several muscle activity periods.

We have randomly picked out five 20000 sample long EMG signals for training material and another five 20000 sample long EMG signals for testing material. On other words, we have used two independent materials for testing and training, both consisting of 100000 samples.

2.2 Spectral Features Mean Frequency and Median Frequency

The mean and median frequencies are calculated from the frequency spectrum of the segmented signal. Signal segments are sliding over the signal with one sample step (segments are heavily overlapping). The frequency spectrum is obtained by taking the FFT of the segment, using a Hanning window of length 1024. The frequency spectrum consists of 512 amplitude coefficients, A_i.

The mean frequency MNF is the amplitude-weighted average of the frequencies,

$$MNF = \frac{\sum_{i=1}^{M} f_i A_i}{\sum_{i=1}^{M} A_i} \quad (1)$$

Graphically, the median frequency is the frequency dividing the area of the amplitude spectrum into equal halves. The value can be computed using a cumulative function

$$c_{fk} = \frac{\sum_{m}^{f_k} A_m}{\sum_{m} A_m} \quad (2)$$

The median frequency MDF is the value of f_k for which the value of c_{fk} is as close to 1/2 as possible.

2.3 The DCT Method

The proposed compression technique is based on discrete cosine transformation which is a very popular transformation used in many compression schemes, especially in image compression standards such as JPEG. There are also applications for biomedical signal compression based on DCT (Cuerrero and Mailhes, 1997; Berger et al., 2003). The idea of transformation coding is that the sequence of *n* data samples of one domain is rotated to some other domain with equation

$$\mathbf{X} = \mathbf{TY} \quad (3)$$

where **X** is the vector of original signal coefficients, **Y** is the vector of transformed coefficients and **T** is the transform matrix. The DCT coefficients of *n* data samples in one-dimensional case is (Salomon, 2004) given by

$$G_f = \sqrt{\frac{2}{n}} C_f \sum_{t=0}^{n-1} p_t \cos\left[\frac{(2t+1) f\pi}{2n}\right] \quad (4)$$

where

$$C_f = \begin{cases} \frac{1}{\sqrt{2}}, & f = 0, \\ 1, & f > 0, \end{cases} \quad for \quad f, t = 0,1,...,n-1. \quad (5)$$

Input vector of *n* data values is p_t and the output vector is a set of *n* DCT coefficients G_f. The inverse DCT transformation is (Salomon, 2004) given by

$$p_t = \sqrt{\frac{2}{n}} \sum_{j=0}^{n-1} C_j G_j \cos\left[\frac{(2t+1) j\pi}{2n}\right], \quad for \quad t = 0,1,...,n-1. \quad (6)$$

DCT compression concentrates signal energy to a small number of DCT coefficients and the compression is usually achieved by eliminating the coefficients containing less information.

The DCT method applied here is based on three steps:

- DCT
- Eliminating some of DCT coefficients by using a masking vector
- Scalar quantization of the coefficients

First step was to calculate DCT from the original signal using blocks of 16, 24 or 32 signal coefficients. In these tests DCT was done by using MatLab's DCT-function. After that, some of the coefficients were eliminated by using binary maskvector. Maskvector is the same size as the used DCT block size. If maskvector's value in some index is zero, the value of corresponding index of DCT block will be eliminated. Otherwise maskvector's value is one and DCT coefficient in corresponding index will not be eliminated.

Maskvector is constant during the whole compression process and the same vector is used when compression is done and when signal is decompressed. Before IDCT, receiver adds zeros at those indexes of DCT block where coefficients have been eliminated to have correct number of reconstructed signal coefficients.

In this study, we have used masking to eliminate high end DCT coefficients. For block size 16 coefficients we masked out last 3, 5 and 7 DCT coefficients, for block size 24 respectively 4, 8 and 12 DCT coefficients, and for block size 32 respectively 5, 10 and 15.

After masking the selected coefficients, the rest of coefficients will be scalar quantized. Compression in this method comes from masking some DCT coefficients and from scalar quantization.

Decompression is done by finding the DCT values corresponding to indexes from codebook, adding zeros to those places of the DCT block where coefficients have been eliminated and making the IDCT.

2.4 Scalar Quantization of Coefficients

In this study, non-uniform scalar quantization method was used to quantize the DCT coefficients. In a uniform scalar quantization the difference between every value in codebook is the same, whereas in a non-uniform scalar quantization the difference between codebook values depends on the distribution of coefficients' probabilities. In the intervals where the probability of that the coefficient is placed on that interval is large, the difference between codebook values is short, and where the probability of coefficient is placed on some interval is small, the difference between codebook values is bigger.

Table 1: Raw remaining sizes and mean-squared-errors (MSE) of compressed signals in percentages by variations.

Codebook size 64 (6 bit)		
Segment length 16 samples		
Without mask	50%	25.6498
Masking last 3	41%	25.9935
Masking last 5	34%	27.6196
Masking last 7	28%	36.1951
Segment length 24 samples		
Without mask	50%	17.0290
Masking last 4	42%	17.2294
Masking last 8	33%	19.0580
Masking last 12	25%	36.3946
Segment length 32 samples		
Without mask	50%	19.5787
Masking last 5	42%	19.7208
Masking last 10	34%	20.8712
Masking last 15	27%	31.1169
Codebook size 256 (8 bit)		
Segment length 16 samples		
Without mask	67%	20.2835
Masking last 3	54%	20.6467
Masking last 5	46%	22.2934
Masking last 7	38%	30.9039
Segment length 24 samples		
Without mask	67%	13.6706
Masking last 4	56%	13.8864
Masking last 8	44%	15.7420
Masking last 12	33%	33.1424
Segment length 32 samples		
Without mask	67%	18.4853
Masking last 5	56%	18.6404
Masking last 10	46%	19.8118
Masking last 15	35%	30.1023

We constructed the codebooks by using Matlab's KMEANS function. Before using KMEANS function, the DCT of the training signal was calculated using the same DCT block size which will be used when compressing the test signal. KMEANS function was given the following parameters: training signal, which has 50000 samples, replicates 'rep' was 3, which made method more optimal, maximum number of iterations 'maxiter' was 800 and 'EmptyAction' was 'singleton', which creates a new cluster consisting of the one point furthest from its centroid. We tested codebook sizes 64 and 256. For codebook size 64, it is possible to present all codebook indexes with 6 bits and respectively for codebook size 256, indexes are presented with 8 bits.

3 RESULTS

The transformation itself has no compression effect; all the compression is gained with irreversible masking and scalar quantization of DCT coefficients.

The achieved compression rations and related MSE values by processing variations are listed in Table 1. The general observation is that the MSE increases when more coefficients are masked out and MSE decreases when codebook size increases.

3.1 The Parameter Model

The mean frequency and median frequency values are calculated from sliding segments for original testsignal and all compressed-decompressed signals. In every case we got 98974 MNF, MDF -pairs from every signal. These values are compared time synchronically against values of the original unprocessed test material. That way we got new set of value pairs:

$$\left(MNF_i^{original}, MNF_i^{processed}\right)$$
$$\left(MDF_i^{original}, MDF_i^{processed}\right) \quad (7)$$

where $i = 0,...,98973$ is the segment number.

The pairs of values make possible the evaluation of the effects of lossy compression to essential medical parameters from-frequency-to-frequency. In an ideal case, there are no differences.

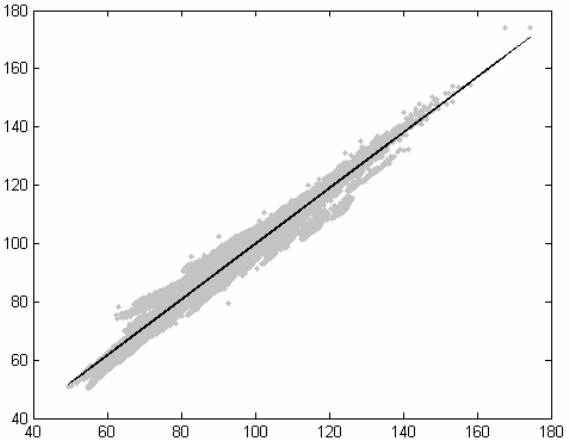


Figure 1: Idea of fitting the regression line.

To model the behaviour of original MNF and MDF values against the processed values, we fit the regression lines to all sets with Matlab's POLYFIT function.

$$MNF^{processed} = aMNF^{original} + b$$
$$MDF^{processed} = cMDF^{original} + d \quad (8)$$

In Figure 1, the best fit line can be seen inside the cloud of data points. Both axes are in frequency (Hz) and the points are presented as the original value on X-axis against the processed value on Y-axis. The line coefficients and the norm of residuals are listed in Table 2 - 5. If the line is exactly diagonal, there is no error between the medical parameters of original and processed signals.

The error of MNF value is typically positive in low frequencies (the MNF of processed signal is higher than the MNF of the original signal) and negative in high frequencies. Reversal point is around 80 Hz. The negative error in high frequencies is smaller on nonmasked cases and the masking increases it. The behaviour of the error of MDF value is similar to MNF value, but typically smaller in absolute value.

The line coefficients and the norm of residuals values not seems to be dependent on segment length. By comparing MSE values in Table 1 and norm of residual values in Tables 2-5, can be recognized that results are more or less correlated with each other.

Table 2: Line coefficients and the norm of the residuals of MNF values.

Codebook size 64 (6 bit)		
Segment length 16 samples		
Without mask	a=0.9611 b=5.0618	526.1498
Masking last 3	a=0.9479 b=5.6428	592.4270
Masking last 5	a=0.9401 b=6.1727	727.8094
Masking last 7	a=0.9425 b=5.9066	954.0903
Segment length 24 samples		
Without mask	a=0.9824 b=2.5583	280.5778
Masking last 4	a=0.9616 b=3.6733	486.0408
Masking last 8	a=0.9421 b=4.6243	833.8518
Masking last 12	a=0.9110 b=6.0588	1.1791e+003
Segment length 32 samples		
Without mask	a=0.9780 b=2.8869	301.1080
Masking last 5	a=0.9635 b=3.5139	394.1623
Masking last 10	a=0.9432 b=4.5146	626.5258
Masking last 15	a=0.9121 b=5.4824	972.4219

Table 3: Line coefficients and the norm of the residuals of MNF values.

Codebook size 256 (8 bit)			
Segment length 16 samples			
Without mask	a=0.9721	b=3.4008	412.3716
Masking last 3	a=0.9607	b=3.9323	544.9072
Masking last 5	a=0.9552	b=4.3098	711.5584
Masking last 7	a=0.9553	b=4.3155	946.9896
Segment length 24 samples			
Without mask	a=0.9865	b=1.7056	242.4786
Masking last 4	a=0.9660	b=2.8420	505.2355
Masking last 8	a=0.9469	b=3.8802	832.9791
Masking last 12	a=0.9160	b=5.3868	1.1716e+003
Segment length 32 samples			
Without mask	a=0.9752	b=2.8877	292.1546
Masking last 5	a=0.9605	b=3.5969	423.3512
Masking last 10	a=0.9412	b=4.5570	651.8788
Masking last 15	a=0.9119	b=5.4024	979.2064

Table 4: Line coefficients and the norm of the residuals of MDF values.

Codebook size 64 (6 bit)		
Segment length 16 samples		
Without mask	c=0.9943 d=0.9374	337.6673
Masking last 3	c=0.9895 d=1.0218	344.2408
Masking last 5	c=0.9837 d=1.1714	385.5636
Masking last 7	c=0.9689 d=1.4947	501.9240
Segment length 24 samples		
Without mask	c=0.9949 d=0.6864	291.5252
Masking last 4	c=0.9890 d=0.8036	355.9760
Masking last 8	c=0.9774 d=1.0882	477.4020
Masking last 12	c=0.9328 d=2.3365	708.1398
Segment length 32 samples		
Without mask	c=0.9960 d=0.6308	279.4163
Masking last 5	c=0.9910 d=0.7421	295.6508
Masking last 10	c=0.9807 d=1.0079	361.8950
Masking last 15	c=0.9445 d=1.9601	574.9671

Table 5: Line coefficients and the norm of the residuals of MDF values.

Codebook size 256 (8 bit)		
Segment length 16 samples		
Without mask	c=0.9955 d=0.5822	261.3096
Masking last 3	c=0.9914 d=0.6590	289.8141
Masking last 5	c=0.9863 d=0.7767	344.2559
Masking last 7	c=0.9717 d=1.1063	469.8654
Segment length 24 samples		
Without mask	c=0.9978 d=0.3327	243.8635
Masking last 4	c=0.9918 d=0.4807	296.6823
Masking last 8	c=0.9806 d=0.7793	422.6246
Masking last 12	c=0.9365 d=2.0190	672.4723
Segment length 32 samples		
Without mask	c=0.9932 d=0.7448	261.7802
Masking last 5	c=0.9884 d=0.8634	287.3707
Masking last 10	c=0.9779 d=1.1613	362.2217
Masking last 15	c=0.9426 d=2.0834	570.4948

3.2 Contemplation of Error

Examining the mean absolute error of MNF and MDF values frequency-by-frequency over clinically interesting frequency range from 40 Hz to 180 Hz is an entirely novel approach.

The mean absolute error (MAE) is calculated by sorting the value pairs (Equation 8) in increasing order and averaging the differences between original and processed value inside the pair. It must be noticed that the distribution of the value pairs is not uniform; on the contrary, the average value is in some cases coarse.

By examining Figures 2 – 4, it can be easily noticed that the mean absolute error of MNF and MDF get the least values between 80 and 120 Hz in all processing variations. Error is very moderate within this range, and the segment length itself doesn't dominate the error.

In the range less than 80 Hz, the error increases when more coefficients are masked out. However, behaviour is similar with MNF and MDF values and also with codebook size 64 (6 bit) and codebook size 256 (8 bit).

The most prominent differences can be seen in the range over 120 Hz. The error is multifold compared to other ranges and heavily increasing when more coefficients are masked out. At this range the errors are also more dependent on the codebook size.

Generally, the MNF error is larger than the MDF error. The segment lengths have not fundamental effect on error. Again, by comparing MSE values in Table 1 and peak level of the MAE in the range over 120 Hz in Figures 2-5, can be recognized that results are more or less correlated with each other, but not so evidently than in case of the norm of residual values.

4 CONCLUSIONS

The main value of this study was to reveal the complexity of error evaluation on EMG signal lossy compression studies. Guerrero and Mailhes (1997) have used standard deviation estimator -based SNR to evaluate the quality of the process. Wellig et al. (1998) have used both SNR and PRD on quality evaluation. Berger et al. (2003) use energy -based SNR as a tool for quality evaluation. None of these studies cover any medical parameters. Chan, Lovely and Hudgins (1997) were first ones to use medical parameters in performance evaluation. Carotti et al. (2006) have used both MSE and some medical

parameters, including MNF and MDF, for quality evaluation. Examination is made via four force levels and the results show a valid correlation between MSE, MNF, and MDF values. Grönfors, Reinikainen and Sihvonen (2006) have used PRD value and percentual differences of MNF and MDF values in quality evaluation. Also these values indicate correlative behaviour. The use of averaged values over signals is common for all the referred studies.

The averaged processing errors with standard deviations of medical parameters form the baseline for the evaluation of a lossy compression method. However, there are pitfalls in the use of averaged error values. Only the error examinations over the whole clinically interesting range of parameter values expose the fidelity.

In this study we have used frequency-by-frequency aspect and compared synchronically generated medical parameters of original and processed signals. We have found that there is more or less correlation between MSE values and errors in medical parameters. However, this interdependency can only reveal the coarse amount of error, not errors natural for a specific range of MNF or MDF values.

The contemplation of error approach (chapter 3.2) has strong analytic use in finding out the values for which the medical parameters are valid. The parameter model approach (chapter 3.1) has both theoretical, analytical, value and practical, predictive usage. The generated regression line can be used for estimating the true value of the processed parameter. Together both approaches can produce a tool for calculating the corrected MNF and MDF value and an index for their quality.

Some of the achieved results are hypothetical, such as the best achived compression ratio has the worst MSE and the effect of masking on error in high frequency range. With DCT-based method, the segment length seems not to have prominent effect on error as with direct vector quantization based method has (Grönfors and Päivinen, 2006).The method should be further tested with larger datasets and with larger quantity of different lossy compression methods.

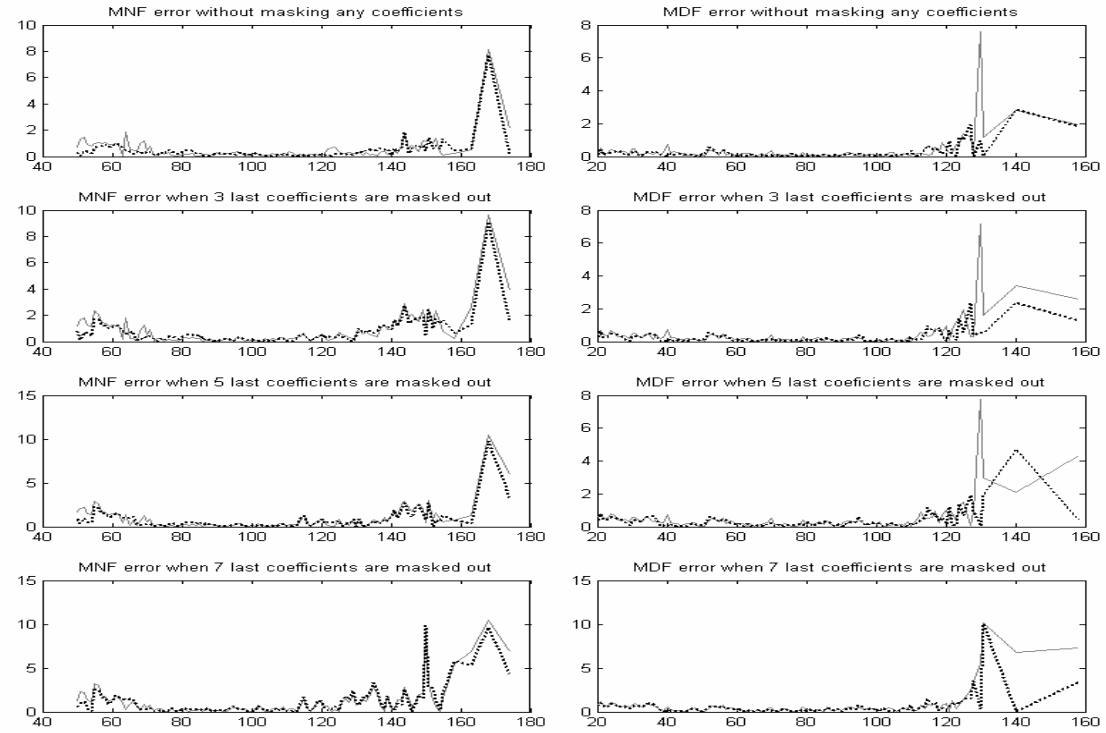


Figure 2: Mean absolute errors of MNF and MDF values for segment length 16. Solid line for codebook size 64 and dotted line for codebook size 256.

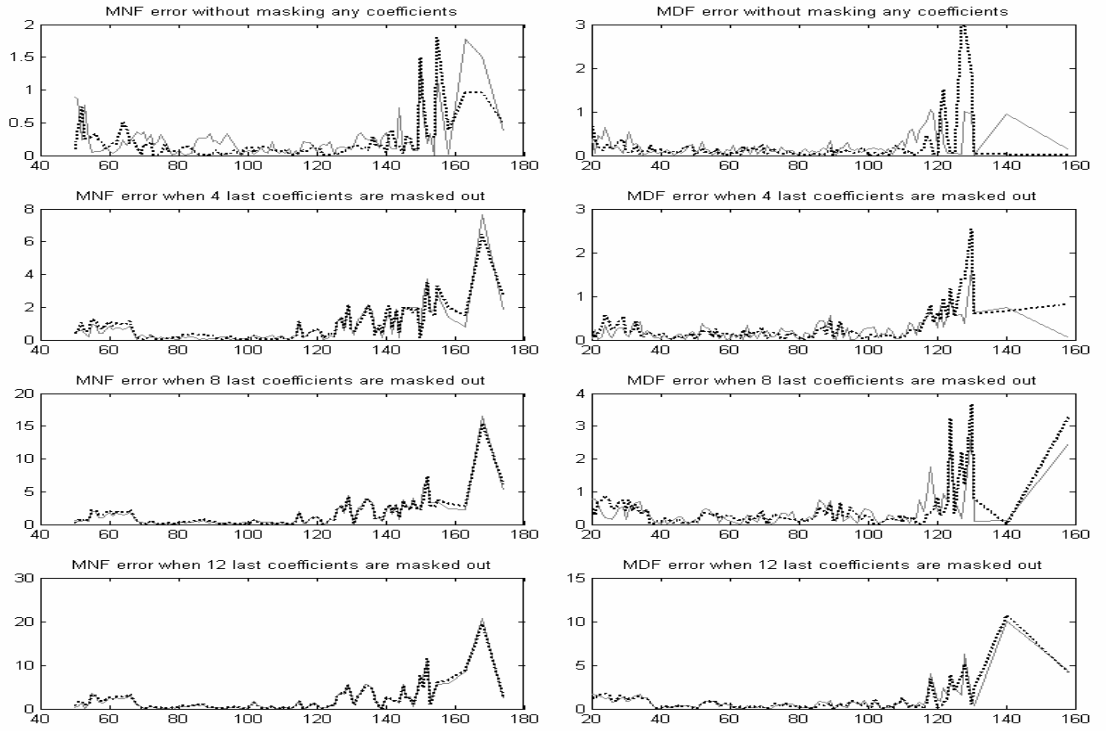


Figure 3: Mean absolute errors of MNF and MDF values for segment length 24. Solid line for codebook size 64 and dotted line for codebook size 256.

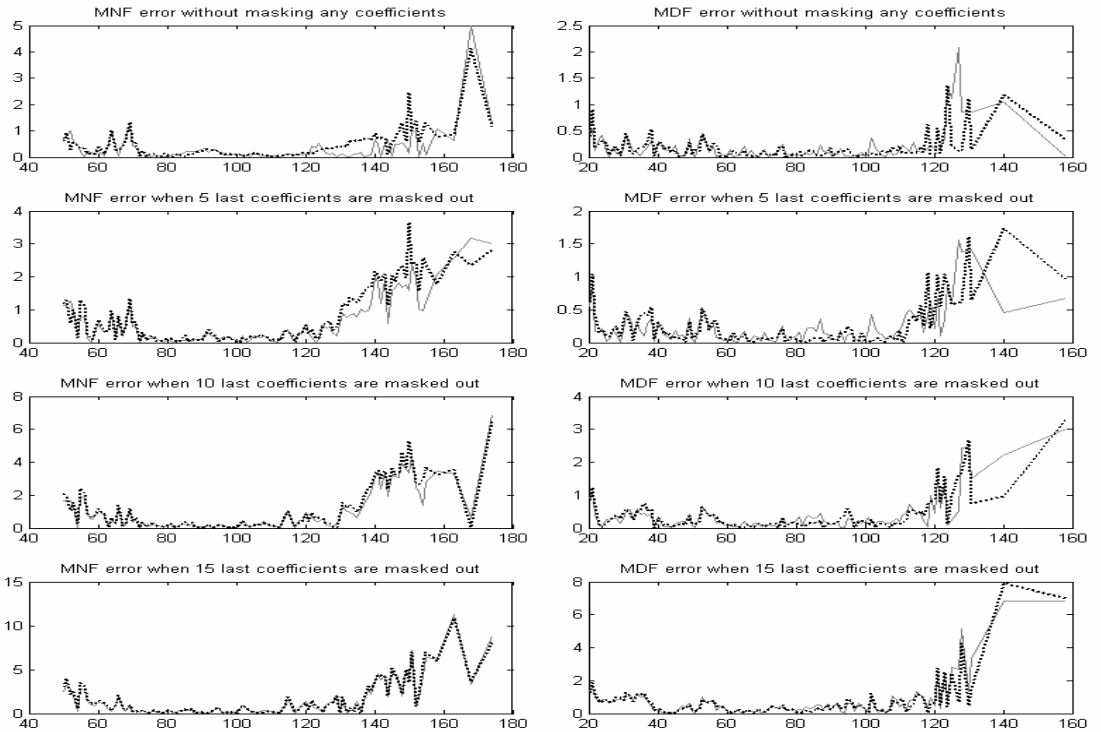


Figure 4: Mean absolute errors of MNF and MDF values for segment length 32. Solid line for codebook size 64 and dotted line for codebook size 256.

ACKNOWLEDGEMENTS

The authors thank MD PhD Teuvo Sihvonen for his valuable comments and support in EMG data collection.

REFERENCES

Berger, P., Nascimento, F., Carmo, J., Rocha, A., dos Santos, I., 2003. Algorithm for compression of EMG signals, In *Proc. 25th Annual International Conf IEEE. Engineering in Medicine and Biology society,* Cancun, Mexico, 1299-302

Carotti, E., De Martin, J., Merletti, R., Farina, D., 2006. Compression of surface EMG signals with algebraic code exited linear prediction, *Medical Engineering & Physics*, Article in press.

Chan, A., Lovely, D., Hudgins, B., 1997. Errors associated with the use of adaptive differential pulse code modulation in the compression of isometric and dynamic myo-electric signals, *Medical and Biological Engineering and Computing*, 36, 215-219

Cuerrero, A., Mailhes, C., 1997. On the choice of an electromyogram data compression method*, In Proc. 19th Annual International Conf IEEE. Engineering in Medicine and Biology society,* Chicago, IL, USA, 1558-61

Farina, D., Merletti, R., 2000. Comparison of algorithms for estimation of EMG variables during voluntary isometric contractions, *Journal of Electromyography and Kinesiology*, 10, 337-349

Filligoi. GD., Felici, F., 1999. Detection of hidden rhytms in surface EMG signals with a non-linear time-series tool, *Medical Engineering & Physics*, 21, 439-448

Grönfors, T., Päivinen, N., 2006, The effect of vector length and gain quantization level on medical parameters of EMG signals on lossy compression. *In Proc. 3th International Conference on Advances in Medical, Signal and Information Processing MEDSIP,* Glasgow, UK.

Grönfors, T., Reinikainen, M., Sihvonen, T., 2006. Vector quantization as a method for integer EMG signal compression, *Journal of Medical Engineering & Technology,* 30(1), 41-52

Salomon, D., 2004. *Data Compression* The Complete Reference, Springer-Verlag, New York

Sihvonen, T., Sihvonen, P., Kuusrainen, S., Grönfors, T., 2004. Lightweight embedded system for acquiring simultaneous electromyogenic activity and movement data, In *Proceedings of the 6th Nordic Signal Processing Symposium - NORSIG 2004*, June 9-11, 2004, Espoo, Finland, 177-179.

Wellig, P., Cheng, Z., Semling, M., Moschytz, G., 1998. Electromyogram data compression using single-tree and modified zero-tree wavelet encoding, *In Proc. 20th Annual International Conf IEEE. Engineering in Medicine and Biology society,* Hong Kong Sar, China, 1303-6.

FAST ESTIMATION FOR RANGE IDENTIFICATION IN THE PRESENCE OF UNKNOWN MOTION PARAMETERS

Lili Ma, Chengyu Cao, Naira Hovakimyan, Craig Woolsey
Department of Aerospace and Ocean Engineering, Virginia Tech, Blacksburg, VA 24061-0203
{*lma05, chengyu, nhovakim, cwoolsey*}*@vt.edu*

Warren E. Dixon
Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611-6250
wdixon@ufl.edu

Keywords: Fast adaptive estimator, Range identification.

Abstract: A fast adaptive estimator is applied to the problem of range identification in the presence of unknown motion parameters. Assuming a rigid-body motion with unknown constant rotational parameters but known translational parameters, extraction of the unknown rotational parameters is achieved by recursive least square method. Simulations demonstrate the superior performance of fast estimation in comparison to identifier based observers.

1 INTRODUCTION

A variety of 3D motion estimation algorithms have been developed since 1970's, inspired by such disparate applications as robot navigation, medical imaging, and video conferencing. Even though motion estimation from imagery is not a new topic, continual improvements in digital imaging, computer processing capabilities, and nonlinear estimation theory have helped to keep the topic current. Assuming that the motion of the moving object follows certain structure, which can have parametric uncertainties, extended Kalman filter (EKF) has been used to estimate the states and parameters of the nonlinear system associated with the moving object dynamics. Application of EKF assumes linearization about the estimated trajectory. However, for the motion estimation from imagery the geometric structure of the perspective system can be lost during the linearization(Ghosh et al., 1994; Dixon et al., 2003). Refs. (Jankovic and Ghosh, 1995; Chen and Kano, 2002; Dixon et al., 2003; Karagiannis and Astolfi, 2005; Ma et al., 2005) have considered nonlinear observers for perspective dynamic systems (PDS) arising in visual tracking problems. In general, a PDS is a linear system, whose output is observed up to a homogeneous line(Chen and Kano, 2002). This class of nonlinear observers is referred to as perspective nonlinear observers.

Perspective nonlinear observers (Jankovic and Ghosh, 1995; Chen and Kano, 2002; Dixon et al., 2003; Karagiannis and Astolfi, 2005; Ma et al., 2005) are used quite often for determining the unknown states (i.e., the 3D Euclidean coordinates) of a moving object with known motion parameters. For example, an identifier-based observer was proposed in(Jankovic and Ghosh, 1995) to estimate a stationary point's 3D position using a moving camera. Another discontinuous observer, motivated by sliding mode and adaptive methods, is developed in(Chen and Kano, 2002) that renders the state observation error uniformly ultimately bounded. A state estimation algorithm with a single homogeneous observation (i.e., a single image coordinate) is presented in(Ma et al., 2005). A reduced-order nonlinear observer is described in(Karagiannis and Astolfi, 2005) to provide asymptotic range estimation. All these results are based on the assumption that the object is following a known motion dynamics in the 3D space.

In this paper, we discuss a situation when some of the motion parameters, more specifically, the rotational parameters, are unknown constants. The objective is to achieve fast state estimation and parameter convergence.

One model for the relative motion of a point in the camera's field of view is the following linear system(Jankovic and Ghosh, 1995; Chen and Kano, 2002; Dixon et al., 2003; Karagiannis and Astolfi,

2005):

$$\begin{bmatrix}\dot{X}(t)\\ \dot{Y}(t)\\ \dot{Z}(t)\end{bmatrix} = \begin{bmatrix}0 & w_1 & w_2\\ -w_1 & 0 & w_3\\ -w_2 & -w_3 & 0\end{bmatrix}\begin{bmatrix}X(t)\\ Y(t)\\ Z(t)\end{bmatrix} + \begin{bmatrix}b_1\\ b_2\\ b_3\end{bmatrix}, \tag{1}$$

where the matrix $[w_i]$ presents the rotational dynamics, the vector $[b_i]$ corresponds to the translational motion, while $[X,\ Y,\ Z]^\top$ are the coordinates of the point in the camera frame. From the 2D image plane, the homogeneous output observations are given by

$$x_1(t) = X(t)/Z(t), \quad x_2(t) = Y(t)/Z(t). \tag{2}$$

These equations might model either a stationary point's 3D position as observed from a moving camera (assuming that the moving camera's velocities can be measured(Jankovic and Ghosh, 1995)) or a moving point's 3D position as observed from a stationary camera(Tsai and Huang, 1981). In general, w_i can be time-dependent, but in this paper we limit the discussion to constant w_i's.

Let

$$\begin{aligned} x(t) &= [x_1(t), x_2(t), x_3(t)]^\top \\ &= [X(t)/Z(t), Y(t)/Z(t), 1/Z(t)]^\top. \end{aligned} \tag{3}$$

The system (1) with output observations (3) is equivalent to the system

$$\begin{cases} \begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = \begin{bmatrix}b_1 - b_3x_1\\ b_2 - b_3x_2\end{bmatrix}x_3 + \begin{bmatrix}w_2 + w_1x_2 + w_2x_1^2 + w_3x_1x_2\\ w_3 - w_1x_1 + w_2x_1x_2 + w_3x_2^2\end{bmatrix}, \\ \dot{x}_3(t) = (w_2x_1 + w_3x_2)x_3 - b_3x_3^2, \end{cases} \tag{4}$$

with the output

$$y(t) = [x_1(t), x_2(t)]^\top. \tag{5}$$

Estimation of $x_3(t)$ from the measurements $(x_1(t), x_2(t))$ constitutes the range identification problem. Refs. (Jankovic and Ghosh, 1995; Chen and Kano, 2002; Dixon et al., 2003; Karagiannis and Astolfi, 2005; Ma et al., 2005) have solved this problem assuming that the motion parameters w_i and b_i in (1) are known (where $i \in \{1,2,3\}$). Here, we assume that the parameters w_i are unknown. The objective, then, is to estimate $x_3(t)$ as well as the unknown parameters w_i. This problem can be formulated in a way such that an existing identifier-based observer (IBO), described in(Jankovic and Ghosh, 1995), can be applied, such that under certain assumptions, the approach provides exponential convergence of both the range and the parameter estimates. A more general case of the problem is discussed in(Ma et al., 2007), where the rotational matrix is represented by a 3×3 matrix instead of the skew-symmetric matrix as in (1).

In this paper, a recently-developed novel adaptive estimator is applied for the estimation of $x_3(t)$ along with the unknown parameters w_i. A numerical comparison of the performance of this adaptive estimator with the IBO observer is provided.

The paper is organized as follows. Range identification in the presence of unknown parameters via the IBO is presented in Sec. 2. A brief review of the fast estimator is given in Sec. 3. In Sec. 4, fast estimation for the range identification problem with unknown motion parameters is presented. Section 5 presents the simulation results. Section 6 extends the analysis to general affine motion. Finally, section 7 concludes the paper.

2 RANGE IDENTIFICATION IN THE PRESENCE OF UNKNOWN PARAMETERS VIA IBO

Consider the state estimation problem for the perspective dynamic system (7), where the motion parameters w_i (for $i = 1,2,3$) are assumed to be unknown constants. Let θ be a vector of these unknown constants defined as

$$\theta = [w_1, w_2, w_3]^\top. \tag{6}$$

The system (4) can be rewritten as

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = w_s^\top(x_1,x_2)\begin{bmatrix}x_3\\ \theta\end{bmatrix}, \tag{7a}$$

$$\begin{bmatrix}\dot{x}_3(t)\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\underbrace{(w_2x_1 + w_3x_2)x_3 - b_3x_3^2}_{g_s(x_1,x_2,x_3,w_2,w_3)}\\ 0_{3\times 1}\end{bmatrix}, \tag{7b}$$

with

$$w_s^\top(x_1,x_2) = \begin{bmatrix}b_1 - b_3x_1 & x_2 & 1+x_1^2 & x_1x_2\\ b_2 - b_3x_2 & -x_1 & x_1x_2 & 1+x_2^2\end{bmatrix}, \tag{8}$$

which fits into the form of the general nonlinear system to which IBO might be applicable, by regarding $\mathbf{x}_1 = [x_1,x_2]^\top$, $\mathbf{x}_2 = [x_3,\theta^\top]^\top$, and $\phi(\mathbf{x}_1,\mathbf{u}) = 0$ (please refer to(Jankovic and Ghosh, 1995) for details of the IBO).

To apply the IBO observer, we need the following assumption for the system in (7):

Assumption 2.1

1. *Let* $\mathbf{x}(t) = \left[x_1(t),x_2(t),x_3(t),\theta^\top\right]^\top$ *be bounded* $\|\mathbf{x}(t)\| < M$, $M > 0$ *for every* $t \geq 0$. *Let* $\Omega = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}(t)\| < M\}$. *Further, for some fixed constant* $\gamma > 1$, *let* $\Omega_\gamma = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}(t)\| < \gamma M\}$.
2. *The function* $w_s(x_1,x_2)$ *and its first time derivative are piecewise smooth and uniformly bounded.*

Suppose that there exist positive constants L_1, L_2 such that

$$\|w_s^\top(x_1,x_2)\| < L_1, \quad \left\|\frac{\mathrm{d}w_s^\top(x_1,x_2)}{\mathrm{d}t}\right\| < L_2. \quad (9)$$

Further, there do not exist constants κ_i (for $i = 1,2,3,4$) with $\sum_{i=1}^4 \kappa_i^2 \neq 0$ such that

$$\kappa_1 v_1(\tau) + \kappa_2 v_2(\tau) + \kappa_3 v_3(\tau) + \kappa_4 v_4(\tau) = 0, \quad (10)$$

for all $\tau \in [t, t+\mu]$, where $\mu > 0$ is a sufficiently small constant, and $v_i(\tau)$ denotes the i^{th} column in w_s in (8).

It is straightforward to verify that, under Assumption 2.1, the system in (7) verifies the assumptions required for the application of IBO. Estimation of $x_3(t)$, along with the unknown motion parameters θ, can be obtained via direct application of the IBO, as given below.

Letting $e_1 = \hat{x}_1 - x_1$, $e_2 = \hat{x}_2 - x_2$, $e_3 = \hat{x}_3 - x_3$, the following observer can be designed for the system (7)

$$\begin{cases} \begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \end{bmatrix} = GA \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} + w_s^\top(x_1,x_2) \begin{bmatrix} \hat{x}_3 \\ \hat{\theta} \end{bmatrix}, \\ \begin{bmatrix} \dot{\hat{x}}_3 \\ \dot{\hat{\theta}} \end{bmatrix} = -G^2 w_s(x_1,x_2) P \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} + \begin{bmatrix} g_s(x_1,x_2,\hat{x}_3,\hat{w}_2,\hat{w}_3) \\ 0_{3\times 1} \end{bmatrix}, \\ \hat{\mathbf{x}}(t_i^+) = M \dfrac{\hat{\mathbf{x}}(t_i^-)}{\|\hat{\mathbf{x}}(t_i^-)\|}, \end{cases} \quad (11)$$

where $\mathbf{x}$ denotes $[x_1, x_2, x_3, \theta^\top]^\top$, $\hat{\theta}$ denotes the estimation of θ, and the sequence of t_i is defined as

$$t_i = \min\{t : t > t_{i-1} \text{ and } \|\hat{\mathbf{x}}(t)\| \geq \gamma M\}, \; t_0 = 0, \quad (12)$$

for some fixed constant $\gamma > 1$. The closed-loop error dynamics can be derived from (7) and (11) as

$$\begin{cases} \begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} = GA \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} + w_s^\top(x_1,x_2) \begin{bmatrix} e_3 \\ \tilde{\theta} \end{bmatrix}, \\ \begin{bmatrix} \dot{e}_3 \\ \dot{\tilde{\theta}} \end{bmatrix} = -G^2 w_s(x_1,x_2) P \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \\ \qquad + \begin{bmatrix} g_s(x_1,x_2,\hat{x}_3,\hat{w}_2,\hat{w}_3) - g_s(x_1,x_2,x_3,w_2,w_3) \\ 0_{3\times 1} \end{bmatrix}, \end{cases} \quad (13)$$

where $\tilde{\theta} = \hat{\theta} - \theta$ and $\dot{\tilde{\theta}} = \dot{\hat{\theta}}$, since θ is assumed to be a constant vector. The main claim is that there exists a positive constant G_0, such that the estimation errors $[e_1, e_2, e_3, \tilde{\theta}^\top]^\top$ converge to zero exponentially if the constant G in (11) is chosen to be larger than G_0(Jankovic and Ghosh, 1995).

3 FAST ESTIMATOR

Range identification in the presence of unknown motion parameters is further pursued using a recently-developed fast adaptive estimator. The adaptive estimator enables estimation of the unknown time-varying parameters in the system dynamics via fast adaptation (large adaptive gain) and a low-pass filter. If the time-varying unknown signal is linearly parameterized in unknown constant parameters, the adaptive estimator can be further augmented by a recursive least-square algorithm (RLS) to estimate the unknown constant parameters asymptotically(Cao and Hovakimyan, 2007).

In the following, main results of the the adaptive estimator are given for the purpose of completeness. More details are presented in(Cao and Hovakimyan, 2007).

3.1 Preliminaries

Some basic definitions from linear system theory are given in this section.

Definition 3.1 *For a signal $\xi(t),\ t \geq 0,\ \xi \in \mathbb{R}^n$, its $\mathcal{L}_\infty$ norm is defined as*

$$\|\xi\|_{\mathcal{L}_\infty} = \max_{i=1,\ldots,n} \left(\sup_{\tau \geq 0} |\xi_i(\tau)| \right), \quad (14)$$

where ξ_i is the i^{th} component of ξ.

Definition 3.2 *The $\mathcal{L}_1$ gain of a stable proper single–input single–output system $H(s)$ is defined as:*

$$\|H\|_{\mathcal{L}_1} = \int_0^\infty |h(t)|dt, \quad (15)$$

where $h(t)$ is the impulse response of $H(s)$.

Definition 3.3 *For a stable proper m input n output system $H(s)$ its $\mathcal{L}_1$ gain is defined as*

$$\|H\|_{\mathcal{L}_1} = \max_{i=1,\ldots,n} \left(\sum_{j=1}^m \|H_{ij}\|_{\mathcal{L}_1} \right), \quad (16)$$

where $H_{ij}(s)$ is the i^{th} row j^{th} column element of $H(s)$.

3.2 Problem Formulation

Consider the following system dynamics:

$$\dot{x}(t) = A_m x(t) + \omega(t), \qquad x(0) = x_0, \quad (17)$$

where $x \in \mathbb{R}^n$ is the system state vector (measurable), $\omega(t) \in \mathbb{R}^n$ is a vector of unknown time-varying signals or parameters, and A_m is a known $n \times n$ Hurwitz matrix. Let

$$\omega(t) \in \Omega, \quad (18)$$

where Ω is a known compact set. The signal $\omega(t)$ is further assumed to be continuously differentiable with uniformly bounded derivative

$$\|\dot{\omega}(t)\| \leq d_\omega < \infty, \quad \forall\, t \geq 0, \quad (19)$$

where d_ω can be arbitrarily large. The estimation objective is to design an adaptive estimator that provides fast estimation of $\omega(t)$.

3.3 Fast Adaptive Estimator

The adaptive estimator consists of the state predictor, the adaptive law and a low-pass filter, which extracts the estimation information.

State Predictor: We consider the following state predictor:

$$\dot{\hat{x}}(t) = A_m\hat{x}(t) + \hat{\omega}(t), \quad \hat{x}(0) = x_0, \tag{20}$$

which has the same structure as the system in (17). The only difference is that the unknown parameters $\omega(t)$ are replaced by their adaptive estimates $\hat{\omega}(t)$ that are governed by the following adaptation laws.

Adaptive Laws: Adaptive estimates are given by:

$$\dot{\hat{\omega}}(t) = \Gamma_c \text{Proj}(\hat{\omega}(t), -P\tilde{x}(t)), \quad \hat{\omega}(0) = \hat{\omega}_0, \tag{21}$$

where $\tilde{x}(t) = \hat{x}(t) - x(t)$ is the error signal between the state of the system and the state predictor, $\Gamma_c \in \mathbb{R}^+$ is the adaptation rate, chosen sufficiently large, and P is the solution of the algebraic equation $A_m^\top P + PA_m = -Q$, $Q > 0$.

Estimation: The estimation of the unknown signal is generated by:

$$\omega_e(s) = C(s)\hat{\omega}(s), \tag{22}$$

where $C(s)$ is a diagonal matrix with its i^{th} diagonal element $C_i(s)$ being a strictly proper stable transfer function with low-pass gain $C_i(0) = 1$. One simple choice is

$$C_i(s) = \frac{\theta_a}{s + \theta_a}. \tag{23}$$

3.4 Convergence Results

The fast adaptive estimator in Sec. 3.3 ensures that $\omega_e(t)$ estimates the unknown signal $\omega(t)$ with the final precision:

$$\|1 - C(s)\|_{\mathcal{L}_1}\|\omega\|_{\mathcal{L}_\infty} + \frac{\gamma_c}{\sqrt{\Gamma_c}}, \tag{24}$$

where $\|\cdot\|_{\mathcal{L}_1}$ denotes the $\mathcal{L}_1$ gain of the system.

To quantify this performance bound between $\omega_e(t)$ and $\omega(t)$, an intermediate signal $\omega_r(t)$ is introduced as:

$$\omega_r(s) = C(s)\omega(s). \tag{25}$$

The following theorem gives the performance bound between $\omega_e(t)$ and $\omega_r(t)$. Details of the proof can be found in(Cao and Hovakimyan, 2007).

Theorem 3.1 *For the system in (17) and the fast adaptive estimator in (20), (21) and (22), we have*

$$\|\omega_e - \omega_r\|_{\mathcal{L}_\infty} \leq \frac{\gamma_c}{\sqrt{\Gamma_c}}, \tag{26}$$

where

$$\gamma_c = \|C(s)H^{-1}(s)\|_{\mathcal{L}_1}\sqrt{\frac{\omega_m}{\lambda_{\min}(P)}}, \tag{27a}$$

$$H(s) = (sI - A_m)^{-1}, \tag{27b}$$

$$\omega_m = \max_{\omega\in\Omega} 4\|\omega\|^2 + 2\frac{\lambda_{\max}(P)}{\lambda_{\min}(Q)}\left(d_\omega \max_{\omega\in\Omega}\|\omega\|\right), \tag{27c}$$

and $\|\cdot\|_{\mathcal{L}_\infty}$ denotes the $\mathcal{L}_\infty$ norm of the signal.

Corollary 3.1 *For the system in (17) and the fast adaptive estimator in (20), (21) and (22), we have*

$$\lim_{\Gamma_c\to\infty}(\omega_e(t) - \omega_r(t)) = 0, \quad \forall\, t \geq 0. \tag{28}$$

We further characterize the performance bound between $\omega_r(t)$ and $\omega(t)$. For simplicity, we use a first order $C(s)$ as in (23). It follows from (25) that

$$\dot{\omega}_r(t) = -\theta_a\omega_r(t) + \theta_a\omega(t), \quad \omega_r(0) = 0. \tag{29}$$

We note that $\omega_r(t)$ can be decomposed into two components:

$$\omega_r(t) = \omega_{r_1}(t) + \omega_{r_2}(t), \tag{30}$$

where $\omega_{r_1}(t)$ and $\omega_{r_2}(t)$ are defined via:

$$\dot{\omega}_{r_1}(t) = -\theta_a\omega_{r_1}(t) + \theta_a\omega(t), \omega_{r_1}(0) = \omega(0) \tag{31a}$$

$$\dot{\omega}_{r_2}(t) = -\theta_a\omega_{r_2}(t), \omega_{r_2}(0) = -\omega(0). \tag{31b}$$

It follows from (31a) that

$$\|\omega_{r_1} - \omega\|_{\mathcal{L}_\infty} = \|1 - C(s)\|_{\mathcal{L}_1}\|\omega\|_{\mathcal{L}_\infty}. \tag{32}$$

Since

$$\lim_{\theta_a\to\infty}\|1 - C(s)\|_{\mathcal{L}_1} = 0, \tag{33}$$

the norm $\|\omega_{r_1} - \omega\|_{\mathcal{L}_\infty}$ can be rendered arbitrarily small by increasing the bandwidth of $C(s)$. Further, $\omega_{r_2}(t)$ decays to zero exponentially and the settling time is inverse proportional to the bandwidth of $C(s)$. Increasing the bandwidth of $C(s)$ implies that $\omega_{r_2}(t)$ decays to zero quickly.

From (26) and (32), when the transients of $C(s)$ due to the initial condition $-\omega(0)$ die out, $\omega_e(t)$ estimates $\omega(t)$ with the final precision given in (24). It is obvious that both the final estimation precision and the transient time can be arbitrarily reduced by increasing the bandwidth of $C(s)$, which leads to smaller $\mathcal{L}_1$ gain for $\|1 - C(s)\|_{\mathcal{L}_1}$. However, the large bandwidth of $C(s)$ leads to further increase of γ_c, which requires large Γ_c to keep the term $\frac{\gamma_c}{\sqrt{\Gamma_c}}$ small. We note that larger Γ_c implies faster computation and requires smaller integration step.

3.5 Extraction of Unknown Parameters

If the time-varying signal $\omega(t)$ can be linearly parameterized in unknown constant parameters and known

nonlinear functions, extraction of the unknown parameters can be achieved by recursive least-square (RLS) algorithm under certain persistent excitation type of condition. The RLS algorithm is reviewed below.

Consider a linear scalar regression model denoted as:

$$\omega_k = \theta^\top \phi_k + e_k, \tag{34}$$

where

$$\theta = [\theta_1, \theta_2, \cdots, \theta_n]^\top \tag{35}$$

is the $n \times 1$ vector of the plant parameters, and

$$\phi_k = [\phi_{k,1}, \phi_{k,2}, \cdots, \phi_{k,n}]^\top \tag{36}$$

is the $n \times 1$ regressor vector at time instant k, while e_k is a zero-mean discrete white noise sequence with variance σ_k^2. When the observation of (ω_k, ϕ_k) has been obtained for $k = 1, \cdots, N$ (with $N > n$), the RLS estimate for θ, denoted by $\hat{\theta}$, can be obtained in the following discrete form(Verhaegen, 1989):

$$\begin{aligned} L_k &= \frac{P_{k-1}\phi_k}{\lambda + \phi_k^\top P_{k-1}\phi_k}, \\ \hat{\theta}_k &= \hat{\theta}_{k-1} + L_k(\omega_k - \phi_k^\top \hat{\theta}_{k-1}), \\ P_k &= \frac{1}{\lambda}\left(P_{k-1} - \frac{P_{k-1}\phi_k\phi_k^\top P_{k-1}}{\lambda + \phi_k^\top P_{k-1}\phi_k}\right), \end{aligned} \tag{37}$$

where $P_0 = pI_{p\times p}$ and $\lambda \in (0,1]$. Coefficients p and λ are design gains and need to be chosen appropriately. When ϕ_k is persistently exciting during the observation period, RLS algorithm ensures the convergence of $\hat{\theta}$ to θ. The convergence rate of RLS can be increased by choosing large λ.

The PE condition of the regressor vector is defined as(Verhaegen, 1989):

Definition 3.4 *The regressor vector ϕ_k is persistently exciting over the observation interval $k_0 \le k \le k_N$ with an exponentially forgetting factor $\lambda \le 1$, if the following condition is fulfilled:*

$$\alpha I \le \sum_{k=k_0}^{k_N} \phi_k\phi_k^\top \lambda^{k_N-k} \le \beta I \tag{38}$$

for some positive $\alpha > 0$ and $\beta > 0$.

4 FAST ESTIMATION FOR RANGE IDENTIFICATION IN THE PRESENCE OF UNKNOWN PARAMETERS

Denote

$$\eta(t) = \begin{bmatrix}\eta_1(t)\\ \eta_2(t)\end{bmatrix}, \tag{39}$$

and write equation (7a) as

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = w_s^\top(x_1,x_2)\begin{bmatrix}x_3(t)\\ \theta\end{bmatrix} = \eta(t). \tag{40}$$

From equations (6), (8), and (40), we have

$$\begin{bmatrix} b_1 - b_3x_1 & x_2 & 1+x_1^2 & x_1x_2 \\ b_2 - b_3x_2 & -x_1 & x_1x_2 & 1+x_2^2 \end{bmatrix}\begin{bmatrix} x_3(t)\\ w_1\\ w_2\\ w_3\end{bmatrix} = \eta(t). \tag{41}$$

Multiplying the first equation in (41) by $T_2 = b_2 - b_3x_2(t)$ and subtracting the second equation from it pre-multiplying it by $T_1 = b_1 - b_3x_1(t)$, we arrive at:

$$\underbrace{[T_2x_2 + T_1x_1, T_2(1+x_1^2) - T_1x_1x_2, T_2x_1x_2 - T_1(1+x_2^2)]}_{\phi^\top(t)}\underbrace{\begin{bmatrix}w_1\\ w_2\\ w_3\end{bmatrix}}_{\theta(t)} = [T_2\eta_1 - T_1\eta_2]. \tag{42}$$

Recursive least squares method can be used to extract w_i's according to (37), with ω replaced by $T_2\eta_1 - T_1\eta_2$. Once w_i (for $i = 1,2,3$) are available, equation (41) takes the form:

$$\begin{bmatrix}b_1 - b_3x_1\\ b_2 - b_3x_2\end{bmatrix}x_3 = \begin{bmatrix}\eta_1\\ \eta_2\end{bmatrix} - \begin{bmatrix} x_2 & 1+x_1^2 & x_1x_2\\ -x_1 & x_1x_2 & 1+x_2^2\end{bmatrix}\begin{bmatrix}w_1\\ w_2\\ w_3\end{bmatrix}, \tag{43}$$

where $x_3(t)$ can be extracted using pseudo-inverse.

Using the fast adaptive estimator described in Sec. 3, estimation of $\eta(t)$, denoted by $\eta_e(t)$, can be obtained via the following steps:

- **State Estimator:**

$$\begin{bmatrix}\dot{\hat{x}}_1\\ \dot{\hat{x}}_2\end{bmatrix} = A_m\begin{bmatrix}\tilde{x}_1\\ \tilde{x}_2\end{bmatrix} + \hat{\eta}(t),\ \begin{bmatrix}\tilde{x}_1\\ \tilde{x}_2\end{bmatrix} = \begin{bmatrix}\hat{x}_1 - x_1\\ \hat{x}_2 - x_2\end{bmatrix}. \tag{44}$$

- **Adaptive Law (use large Γ_c):**

$$\dot{\hat{\eta}}(t) = -\Gamma_c P^\top \begin{bmatrix}\tilde{x}_1 & \tilde{x}_2\end{bmatrix}^\top. \tag{45}$$

- **Extraction:**

$$\eta_e(s) = C(s)\hat{\eta}(s), \quad C(s) = \frac{C}{s+C}. \tag{46}$$

According to Corollary 3.1, the final estimation precision $\eta_e(t) - \eta(t)$ and the transient time to achieve this can be arbitrarily reduced by increasing the bandwidth of $C(s)$. Increasing the bandwidth of $C(s)$ requires larger Γ_c.

The flow chart of state and parameter estimation of a rigid motion using the fast adaptive estimator is illustrated in Fig. 1. In the first step of estimating $\eta(t)$, both the estimation precision and transient time can be arbitrarily reduced by increasing the bandwidth of $C(s)$ and using larger Γ_c. In the second step of extracting $\hat{w}_i$'s from $\eta_e(t)$ using the recursive least square

method, fast speed can be achieved by properly tuning the RLS gains. Estimation of $x_3(t)$, denoted by $\hat{x}_3(t)$, can be obtained from $\eta_e(t)$ and $\hat{w}_i$'s via pseudo-inverse. Since the fast adaptive estimator assumes minimization of the $\mathcal{L}_1$ gain of $1-C(s)$ for performance improvement, it is referred to as $\mathcal{L}_1$ adaptive estimator.

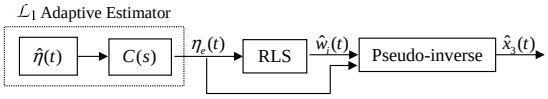


Figure 1: Flow chart of $\mathcal{L}_1$ adaptive estimator.

5 SIMULATION RESULTS

State estimation of $[x_3(t), \theta^\top]^\top$ using the IBO observer (11) and the fast adaptive estimator (44) (46) are implemented in Matlab, where the motion dynamics are selected to be

$$\begin{bmatrix}\dot{X}(t)\\ \dot{Y}(t)\\ \dot{Z}(t)\end{bmatrix} = \begin{bmatrix}0 & -4 & -0.8\\ 4 & 0 & -0.6\\ 0.8 & 0.6 & 0\end{bmatrix}\begin{bmatrix}X(t)\\ Y(t)\\ Z(t)\end{bmatrix} + \begin{bmatrix}10\\ 3\pi\sin(2\pi t)\\ 3\pi\sin(2\pi t+\pi/4)\end{bmatrix},$$
$$(X_0,Y_0,Z_0) = (1,1.5,2.5), \quad x_0 = (X_0/Z_0, Y_0/Z_0, 1/Z_0). \tag{47}$$

First, we present simulation results in the ideal case with no measurement noise. The parameters for the IBO observer and the fast adaptive estimator are chosen to be:

- IBO (referring to (11)):
  $G = 10$, $(\hat{x}_3(0), \hat{w}_1(0), \hat{w}_2(0), \hat{w}_3(0)) = (0,0,0,0)$.
- Fast adaptive estimator (referring to (37), (45), (46)):
  $p = 100$, $\lambda = 0.99999$, $A_m = -I_2$,
  $(\hat{\eta}_1(0), \hat{\eta}_2(0)) = (0,0)$, $\Gamma_c = 2\times 10^8$, $C = 200$.

In both cases, we set $(\hat{x}_1(0), \hat{x}_2(0)) = (x_1(0), x_2(0))$, $M = 30$, $A = I_2$, $P = -1/2 \times I_2$, where I_2 denotes the 2×2 identity matrix.

Estimation of w_i (for $i = 1,2,3$) with the use of the IBO and the fast adaptive estimator is shown in Figures 2 and 3, respectively. Figure 4 shows the zoomed version of Figure 3 for the steady state error. State estimation error of x_3 is plotted in Figure 5 for comparison of both methods.

From Figures 2 and 3, it can be observed that the fast adaptive estimator achieves faster estimation of the motion parameters. The same is true for x_3.

Simulation results are also presented in Figs. 6∼9 when the output is noise-corrupted with uniform bound $\pm 10^{-2}$. The simulation parameters are the same as above. In this case, when extracting $\hat{x}_3(t)$, the output from the pseudo-inverse is further processed

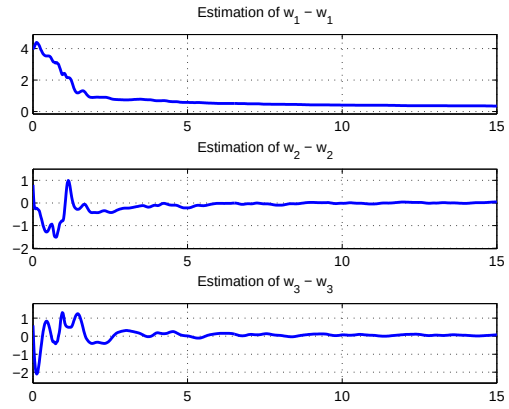


Figure 2: Estimation of motion parameters **using IBO** (without measurement noise).

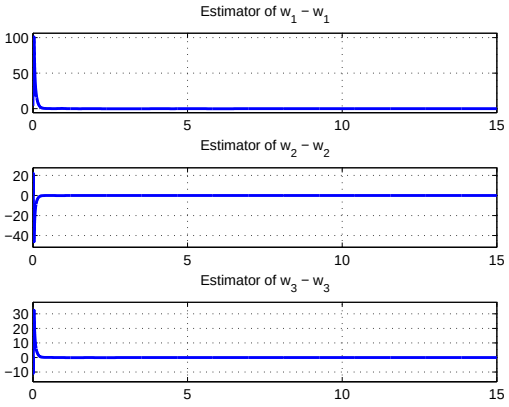


Figure 3: Estimation of motion parameters **using fast adaptive estimator** (without measurement noise).

using a low-pass filter $\frac{30}{s+30}$ to give the final state estimation. We observe that corresponding plots with or without measurement noise are very similar.

6 FURTHER EXTENSION

In this paper, rigid-body motion is considered that contains only three rotational parameters (w_1, w_2, w_3) as given in (1). For general affine motion described by

$$\begin{bmatrix}\dot{X}(t)\\ \dot{Y}(t)\\ \dot{Z}(t)\end{bmatrix} = \begin{bmatrix}a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ a_{31} & a_{32} & a_{33}\end{bmatrix}\begin{bmatrix}X(t)\\ Y(t)\\ Z(t)\end{bmatrix} + \begin{bmatrix}b_1\\ b_2\\ b_3\end{bmatrix}, \tag{48}$$

the rotational matrix contains nine parameters. Assuming that the $[a_{ij}]$ (for $i,j = 1,2,3$) are unknown

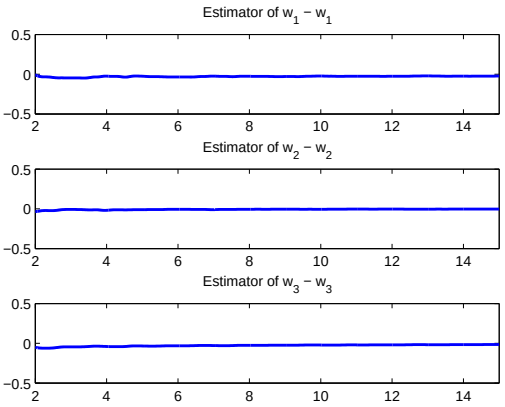


Figure 4: Enlarged view of Fig. 3 (without measurement noise).

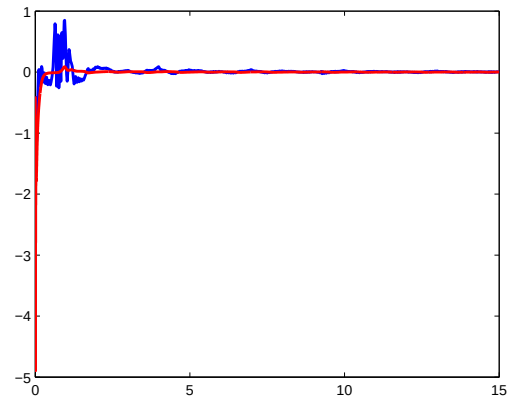


Figure 5: Comparison of state estimation errors (without measurement noise).

constants, the method described in Sec. 4 cannot lead to extraction of the nine unknown parameters in a straightforward way.

The system (48) with output observations (3) is equivalent to the system

$$\begin{cases} \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} b_1 - b_3 x_1 \\ b_2 - b_3 x_2 \end{bmatrix} x_3 + \begin{bmatrix} a_{13} + (a_{11} - a_{33}) x_1 \\ a_{23} + a_{21} x_1 \end{bmatrix} \\ \qquad + \begin{bmatrix} a_{12} x_2 - a_{31} x_1^2 - a_{32} x_1 x_2 \\ (a_{22} - a_{33}) x_2 - a_{31} x_1 x_2 - a_{32} x_2^2 \end{bmatrix}, \\ \dot{x}_3(t) = -(a_{31} x_1 + a_{32} x_2 + a_{33}) x_3 - b_3 x_3^2, \end{cases} \tag{49}$$

with the output (5). The above system can also be rewritten in the form of (7a), where θ and $w_s^\top(x_1, x_2)$ take the forms

$$\theta = [a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}, a_{31}, a_{32}, a_{33}]^\top, \tag{50}$$

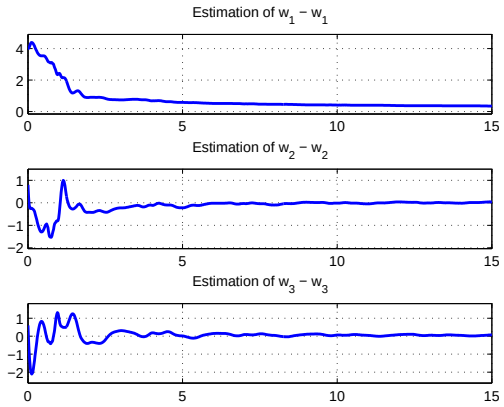


Figure 6: Estimation of motion parameters **using IBO** (with measurement noise).

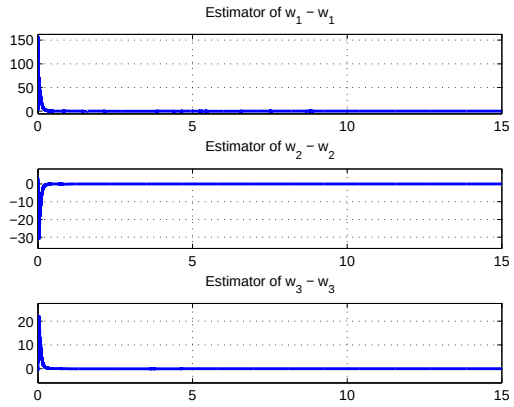


Figure 7: Estimation of motion parameters **using fast adaptive estimator** (with measurement noise).

and

$$w_s^\top(x_1, x_2) = \begin{bmatrix} b_1 - b_3 x_1 & x_1 & x_2 & 1 & 0 & 0 & 0 \\ b_2 - b_3 x_2 & 0 & 0 & 0 & x_1 & x_2 & 1 \end{bmatrix} \begin{bmatrix} -x_1^2 & -x_1 x_2 & -x_1 \\ -x_1 x_2 & -x_2^2 & -x_2 \end{bmatrix}, \tag{51}$$

respectively. Following the logic in Sec. 4, we can write the following system of algebraic equations

$$w_s^\top(x_1, x_2) \begin{bmatrix} x_3 & a_{11} & a_{12} & \cdots & a_{33} \end{bmatrix}^\top = \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix}, \tag{52}$$

with the $w_s^\top(x_1, x_2)$ given in (51). Again, multiplying the first equation in (52) by $T_2 = b_2 - b_3 x_2$ and subtracting the second equation from it pre-multiplying it

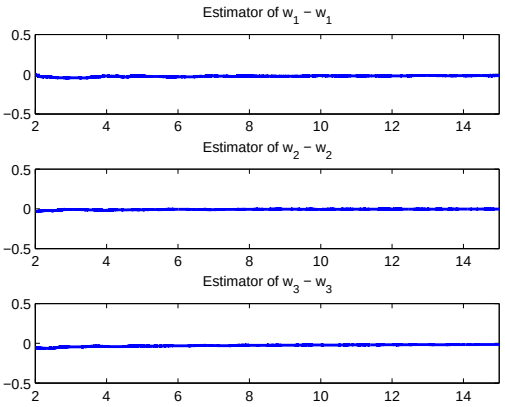


Figure 8: Enlarged view of Fig. 7 (with measurement noise).

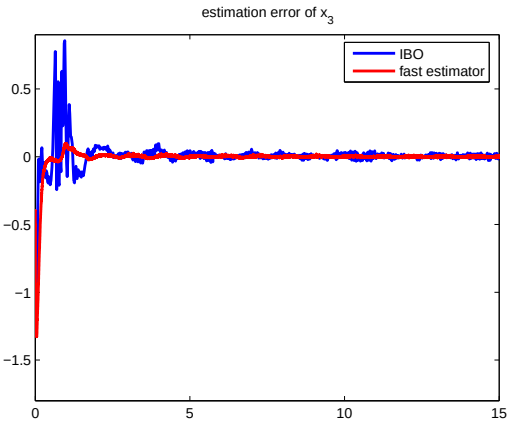


Figure 9: Comparison of state estimation errors (with measurement noise).

by $T_1 = b_1 - b_3 x_1$, we arrive at:

$$\underbrace{[T_2(x_1,x_2,1),\, T_1(x_1,x_2,1),\, (b_1x_2 - b_2x_1)(x_1,x_2,1)]}_{\phi_{\text{affine}}(t)} \begin{bmatrix} a_{11} \\ a_{12} \\ \vdots \\ a_{33} \end{bmatrix} = [T_2\eta_1 - T_1\eta_2]. \tag{53}$$

The nine columns in $\phi_{\text{affine}}(t)$ in (53) are linearly dependent. It is obvious that the $7^{\text{th}}, 8^{\text{th}}$, and 9^{th} columns can be presented as linear combinations of the first six columns. For example, column$_9$ can be written as column$_9$ = column$_5$ − column$_1$. Thus, extraction of the nine unknown parameters cannot be performed by the recursive least square method since it violates the PE condition in (38). Further research will explore the use of adaptive observers for general affine motion identification.

7 CONCLUSION

A recently developed fast adaptive estimator is applied to the range identification problem of a rigid motion in the presence of unknown motion parameters. Fast convergence speed is achieved compared to existing nonlinear perspective observers.

ACKNOWLEDGEMENTS

This work was sponsored in part by ONR Grant #N00014-06-1-0801 and AFOSR MURI subcontract F49620-03-1-0401.

REFERENCES

Cao, C. and Hovakimyan, N. (2007). Fast adaptive estimator for time-varying unknown parameters. To Appear in American Control Conference.

Chen, X. and Kano, H. (2002). A new state observer for perspective systems. *IEEE Trans. on Automatic Control*, 47(4):658–663.

Dixon, W., Fang, Y., Dawson, D., and Flynn, T. (2003). Range identification for perspective vision systems. *IEEE Trans. on Automatic Control*, 48(12):2232–2238.

Ghosh, B., Jankovic, M., and Wu, Y. (1994). Perspective problems in system theory and its application to machine vision. *Journal of Mathematical Systems, Estimation and Control*, 4(1):3–38.

Jankovic, M. and Ghosh, B. (1995). Visually guided ranging from observations of points, lines and curves via an identifier based nonlinear observer. *Systems and Control Letters*, 25:63–73.

Karagiannis, D. and Astolfi, A. (2005). A new solution to the problem of range identification in perspective vision systems. *IEEE Trans. on Automatic Control*, 50(12):2074–2077.

Ma, L., Cao, C., Hovakimyan, N., Dixon, W., and Woolsey, C. (2007). Range identification in the presence of unknown motion parameters for perspective vision systems. To Appear in American Control Conference.

Ma, L., Chen, Y., and Moore, K. (2005). Range identification for perspective dynamic system with a single homogeneous observation. *International Journal of Applied Mathematics and Computer Science*, 15(1):63–72.

Tsai, R. and Huang, T. (1981). Estimating three-dimensional motion parameters of a rigid planar patch. *IEEE Trans. on Acoustic, Speech, and Signal Processing*, ASSP-29(6):1147–1152.

Verhaegen, M. H. (1989). Round-off error propagation in four generally-applicable, recursive, least-squares estimation schemes. *Automatica*, 25(3):437–444.

ADVANCED CONTROL OF AEROBIC INDUSTRIAL WASTEWATER TREATMENT

Matei Vinatoru, Eugen Iancu, Gabriela Canureci and Camelia Maican
University of Craiova, Automation and Mechatronics Department
vinatoru@automation.ucv.ro, iancu@automation.ucv.ro

Keywords: Biological wastewater treatments, control systems, state estimators.

Abstract: The paper present the possibility of automatic control of the biological wastewater treatment station with applications in Romanian Chemical Companies. In this paper are developed a mathematical model for biological aeration basins and two automatic control systems (conventional control structure using three-positional controllers or PLC and advanced control structure using state estimators) for wastewater industrial purification stations.

1 INTRODUCTION

In the present world, environmental issues are a very important topic. More and more countries and international bodies are issuing stringent laws and standards for environment protection. A major field in environment protection is the industrial wastewater treatment, geared toward protecting world waters from pollution. Biological processes are the ones most used in wastewater treatment today. (Chen 2001, Peter, 2003). These processes, used to remove both inorganic and organic products, take place in wastewater treatment plants. The wastewater is treated using complex chemical and biological reactions, before being discharged in the environment. The schematic operational block diagram of such treatment plant is presented in figure 1. The residual wastewater discharged by industrial plants contains a lot of contamination substances (organic and inorganic matters, ammonium and nitric compounds), which shall be eliminated before the water is discharged in environment. Different treatment techniques are used to eliminate those substances, as physical/chemical treatment techniques and treatment by microorganisms called biological aerobe purification.

In the chemical treatment, certain chemicals are added to the wastewater. These chemicals are interacting with the contamination substances, changing there structure and allowing their elimination through mechanical processes (screen, grit, filtration). In the same time, the pH of the solution is brought to the neutral point. Most of fertilizers such as nitrates can be removed this way.

Biological treatment processes are used to remove the dissolved organic load from the water using microorganisms. They use aerobic bacteria for the decay of the organic matter. Aerobic bacteria must be present, in order to perform the chemical conversion of biological contaminants in other substances that can be easily eliminated trough simple mechanical processes.

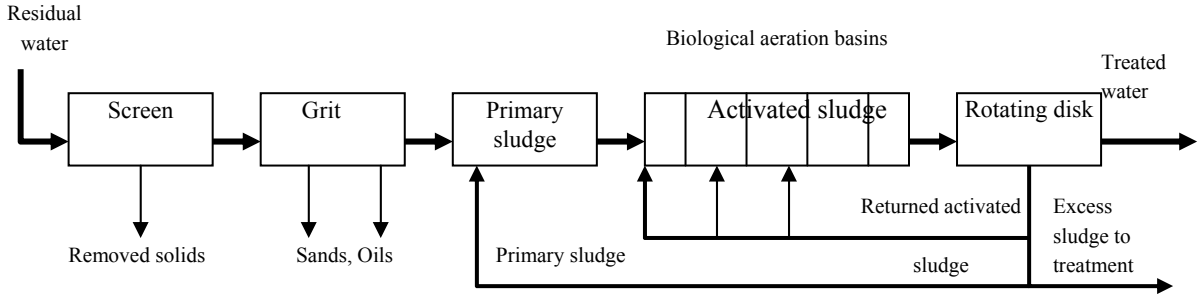


Figure 1: Wastewater industrial purification.

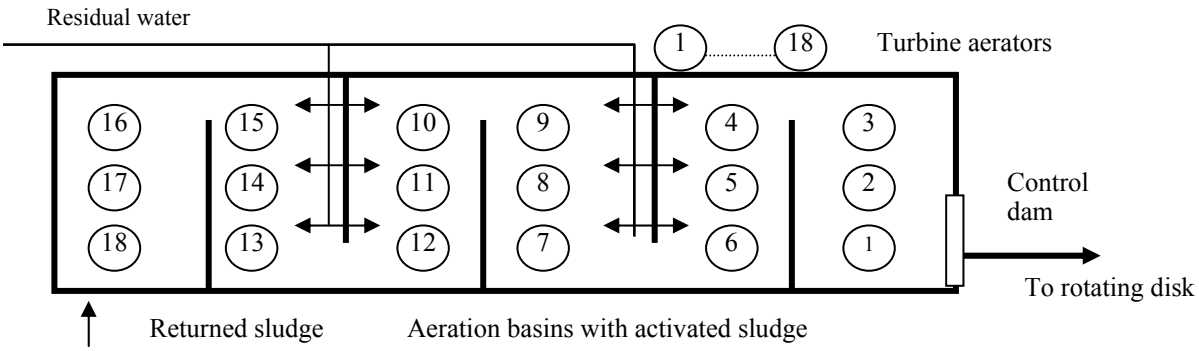


Figure 2: Aeration basins with activated sludge.

The contaminants are converted to carbon dioxide and water as a result of the following reaction.

$$\begin{bmatrix}\text{Organic}\\ \text{load}\end{bmatrix} + O_2 + [\text{Bacteria}] \Rightarrow CO_2 + H_2O + \begin{bmatrix}\text{moore}\\ \text{Bacteria}\end{bmatrix}. \quad (1)$$

In this case the Biological Oxygen Demand (BOD) defines the organic load. The biological treatment processes take place in biological aeration basins. Turbine aerators are used to aerate the water in order to maintain the optimal oxygen concentration for bacteria and also mix the water, keeping the media homogenous. The industrial wastewater and primary sludge (with bacteria) are mixed inside the aeration basins, and bacteria consume the organic matter resulting new bacteria, called activated sludge. The activated sludge exists normally in the form of flakes, which, besides live and dead biomass, contains absorbed and stored, both organic and mineral parts.

In order to control the content of the biomass, it is necessary to control the oxygen dissolved in the basin water, which can be achieved through the control of the turbine aerators (level, flow, speed, or number of running aerators). The turbine aerators inside the aeration basins can be considered as a distributed structure (see figure 2), and considering bacteria's growing and activity requirements, it is necessary to design an advanced control system to assure a high efficiency of the process (water quality, cost reduction). This paper presents an automatic control system for the biological wastewater treatment in Romanian treatment stations. The schematic diagram of the aeration basins is presented in figure 2.

It is necessary to control the aeration process of the wastewater in the biological treatment basins, in order to obtain the optimal conditions for aerobic bacteria growth and evolution (Peter, 2003). The problem of oxygen concentration measurement in solutions was already solved both in the country and abroad (Vinatoru,1979, Vanrolleghem, 2003).

The aeration process can be controlled as follows:

- through the control of the turbine immersion relative to the water level in the basin;
- through the control of transit time of wastewater in the aeration basin through the rising or lowering of the control dam;
- through the number of running aerators in each basin, which is the most efficient method for oxygen concentration control;
- through variation of the rotational speed of the aerators; this method cannot be applied in Romanian installations, since the rotational speed cannot be modified for the motors used.

2 THE MATHEMATICAL MODEL OF THE AERATION BASINS

Considering the oxygen concentration control possibilities, the aeration basins can be considered as an oriented object with three inputs (w-rotational speed of the motors driving the aerators, n-number of aerators running and h-level of water in the basin relative to the lowest position of the aerators) and one output (c-oxygen concentration in the basin). The oxygen concentration is a function of both time and spatial coordinate along the water path from the entrance in basin 6 to the exit over the dam in basin number 1.

We will consider one basin and the mathematical model for the dynamic regime is given by the following equations:

-The equation for the water flow over the dam in the aeration basin 1:

$$F_a = \alpha.b.\sqrt{2g}.(H-l)^{3/2} \text{ where } \alpha=(0{,}5-0{,}6)$$

-The equation for the aeration pump flow:

$$F_p = 738{,}432.\gamma.(n_s/n)^2.h^{3/2}$$

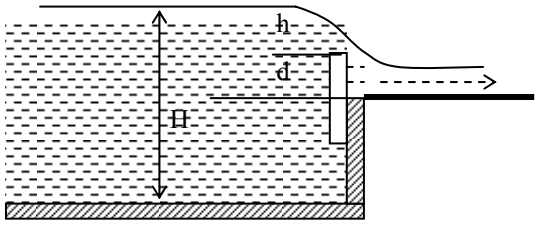


Figure 3: Dam measures.

-The quantity of oxygen used by microorganisms in the time unit and volume unit:

$$\frac{dC}{dt} = k_{au}(C_s - C) \tag{2}$$

-The global mass balance equation can be considered stationary since the water volume V variations can be neglected inside the basins:

$$F_{i1} + F_{i2} + F_{i3} - F_e = 0 \tag{3}$$

-The mass balance equation for the dissolved oxygen:

$$V\frac{dC_i}{dt} = F_{i1}.C_{i1} + F_{i2}.C_{i2} + F_{i3}.C_{i3} + \\ + H_i.F_p(C_r - C_i) - F_e.C_i - k_{au}V(C_s - C_i) \tag{4}$$

The liquid volume in the basin is given by the formula:

$$V = A\,(H_l + h)$$

For a particular case of treatment basins, the general mathematical model described by equations 1 to 6 can be linearized around the stationary values and simplified, resulting the schematic block diagram presented in figure 4.

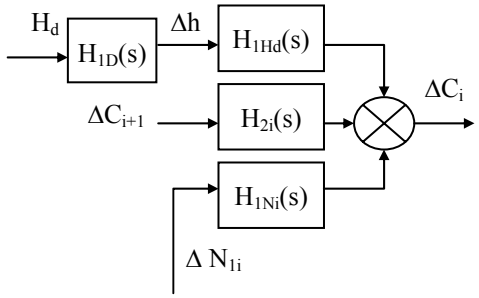


Figure 4: Block diagram of one basin.

1 to 6 can be linearized around the stationary values and simplified, resulting the schematic block diagram presented in figure 4.

The transfer functions are the following form:

$$H_{1Hd}(s) = \frac{b_{1i}s + b_{0i}}{a_{2i}s^2 + a_{1i}s + a_{0i}} \tag{5}$$

$$H_{2i}(s) = \frac{b_{1i}s + b_{2i}}{a_{2i}s^2 + a_{1i}s + a_{0i}} \tag{6}$$

$$H_{1Ni}(s) = \frac{b_{1i}s + b_{3i}}{a_{2i}s^2 + a_{1i}s + a_{0i}} \tag{7}$$

The interconnection between basins 1 to 6 is done through the transfer flow between basins:

$$F_{ei} = \mu_i.A_i\sqrt{2g(H_i - H_{i-1})} + k_{pi}.F_{pi}, i = 1,2,...5$$

where: $\mu_i = 1/\sqrt{1{,}5 + \lambda_i L_i / D_i}$,

$\lambda_i = 0{,}86\left(1 + 2/(H_i + H_{i-1})\right)^{1/3}$,

$A_i = 0{,}5D_i(H_i + H_{i-1}), k_{pi} = (1...3), V_i = S_i.H_i$

and the return flow is: $F_{ri} = N_i * F_{pi}$

Combining all 5 active basins, we obtain the block diagram in figure 5.

3 AUTOMATIC CONTROL OF THE AERATION BASINS

In order to determine the optimal structure of the automatic control system for the oxygen concentration in the biological treatment basins, the mathematical model of the basins was developed, using the technological diagram (figure 1) and the available controls for the oxygen concentration. We tried to get a better approximation of the real process than the one used in (Vinatoru, 1979); therefore it considered the influences at the border between two basins.

Analysing the existing conditions in the aeration basins, we can divide the installation in two big sectors:

- in the first sector, containing basins 1,2 and 3, the oxygen concentration control is done through the number of running aerators N_{1i} and through the height of the dam H_d (figure 4);
- in the second sector, containing basins 4 and 5, the oxygen concentration control is done through the number of running aerators N_{2i}.

The high cost of oxygen sensors and especially the high maintenance cost call for reduction of the number of sensors used. From the analysis of the biological water treatment basins at DOLJCHIM SA Craiova, where the experiments were also made, we determined that a minimum of two sensors are necessary, one being mounted at the exit (measuring concentration C_1) and one in basin 4 (measuring concentration C_4). These measurements will be used as output variables for the controlled process. The right control strategy and structure to be used depends of the financial capability of the company.
We studied three different control structures that can be used for similar basins.

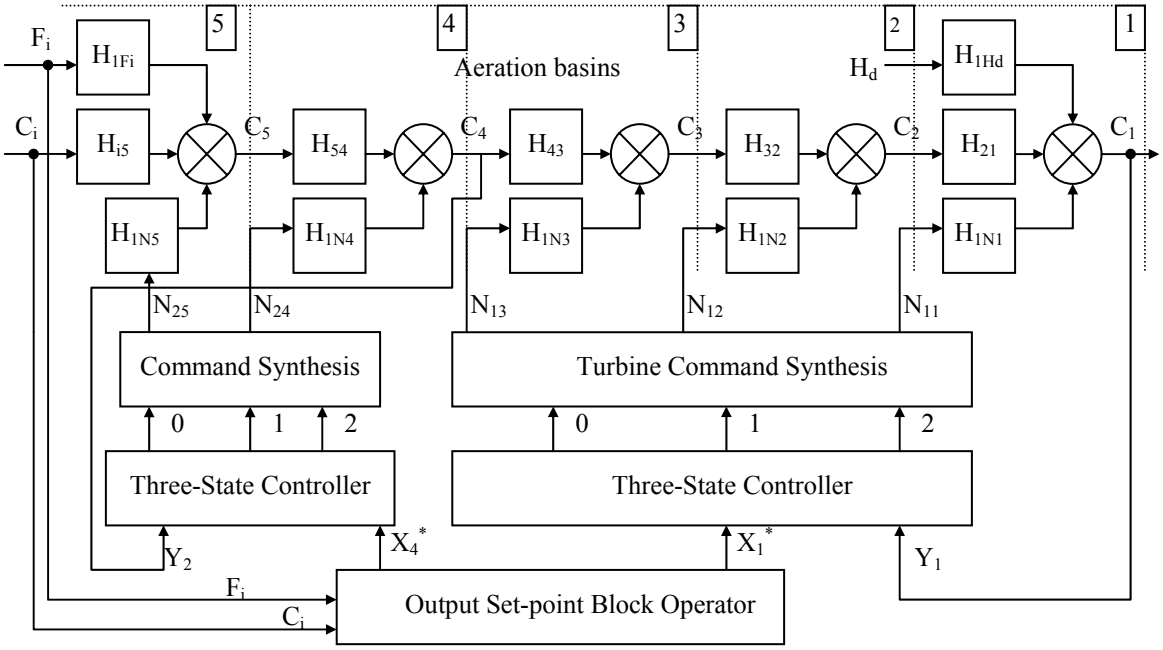


Figure 5: Conventional control diagram.

- Conventional control structure using tri-positional controllers, figure 5, which control the starting and stopping of the aeration turbines depending of the domain in which the oxygen concentrations in basins 1 and 4 are located, according with the algorithm presented in Table 1. This solution was implemented for the biological treatment basins. The control devices for the number of aerators and the actuator for the dam can be easily implemented and their control by the tri-positional controller is done based on the limits C_{min} and C_{max} according with table 1, imposed by the particular conditions inside the aeration basins.

Table 1: Control algorithm.

Control aerator basins 1,2 and 3			
Concentration C_1	0	C_{min}	C_{max}
Aerator running	All	1,3,5,7,9	3,5,7
Dam	Upper position	Upper position	Lower position
Control aerator basins 4 and 5			
Concentration C_4	0	C_{min}	C_{max}
Aerator running	All	11,13,15	11,15

From equations 1 to 6, considering the particular conditions for the aeration basins, it results that the entire process has a very slow dynamic regime, due to the high volume of wastewater compared with the transit time of the water in the basins. Therefore is not necessary a continuous control of the oxygen concentration. Moreover, the bacterial activity does not require an exact oxygen concentration but certain limits between which the activity is running normally.

- Fuzzy control structure, where the fuzzifier and the defuzzifier are following the control rules presented in table 1.

- Control structure using state estimators according with the diagram presented in figure 6. For this structure, the current state in each basin

$X1 = C_1$, $X_2 = C_2$, $X_3 = C_3$, $X_4 = C_4$, $X_5 = C_5$, is estimated based on the measured output values $Y_1 = C_1$, $Y_2 = C_4$.

Using the state variables and applying the command synthesis principles developed by the authors for the control of distributed parameter systems (Vinatoru 1979), the command for the aeration turbines will be generated by the Command Synthesis block in figure 6.

4 CASE STUDY

4.1 Control with Two Tri-positional Controllers

To show the performance and experimental results behaviour of the control structures some experiments have been carried out. The conventional control structure using tri-positional controllers has implemented to the DOLJ Chim SA Wastewater Treatment Plant Craiova.

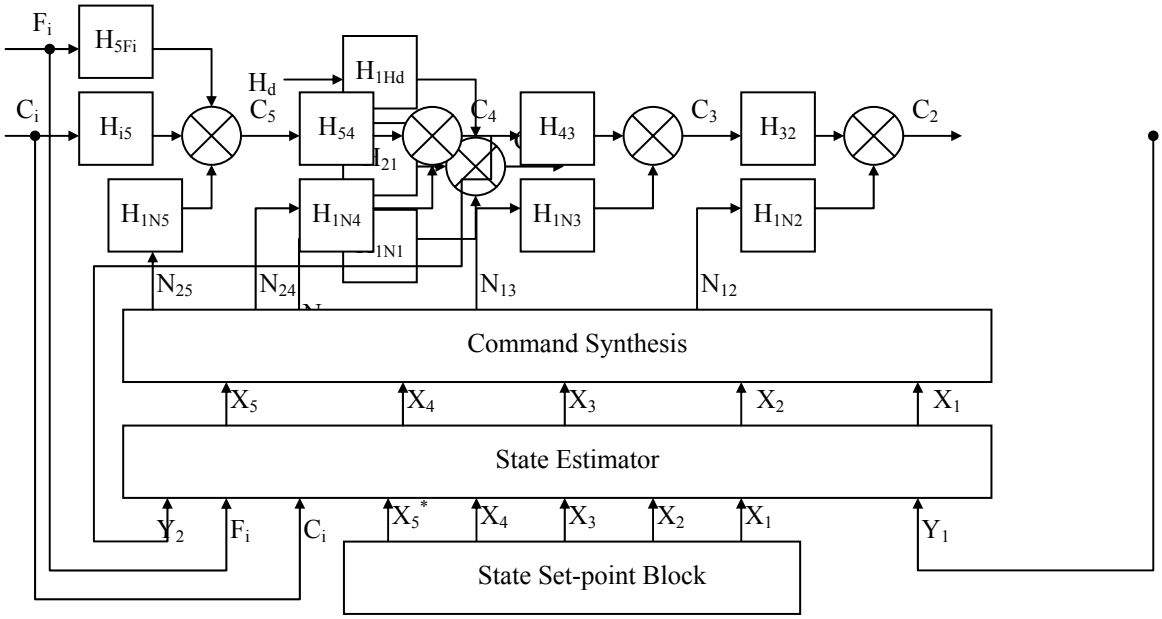


Figure 6: Advanced control diagram.

The experimental results in the various conditions of residual water flow are presented in figure 7. Aeration basins volume: 54.000 m^3 (9000 m^3/basin); Internal recycle flow rate 2700 m^3/hour; Dissolved Oxygen concentration (DO): 6 mg/l basin no. 1 and 4 mg/l in basin no. 4.

4.2 Control using a Sate Estimator

To sow the performance behaviour of advanced controller structure, the proposed control strategy has been implemented to the model of Wastewater purification Plant, presented in figure 6. The transfer functions from this structure are:

$$H_{21}(s) = \frac{0.66(80.4s+1)}{1397s^2+97.73s+1} = H_{32} = H_{43} = H_{54}$$

$$H_{1Ni}(s) = \frac{2.8(91.12s+1)}{1344s^2+96.72s+1}, i = 1...5$$

$$H_{i5}(s) = \frac{0.78(80.4s+1)}{1397s^2+97.73s+1}$$

According with the diagram presented in fig. 6 the state estimator has implemented in the form:

$$\frac{d\hat{x}(t)}{dt} = A_0(q)x(t) + C_0 w(t) \quad (8)$$

where q is elements of the unknown vector considered the tuning parameters of the observer. The input w(t) is:

$$w(t) = c.u(t) + g^T Q(\hat{y}(t) - y_m(t) \quad (9)$$

where g^T is the weighting functions and $Q \in R^2$ is introduced for a better tuning of the observer parameter in function of the difference between the estimated output $\hat{y}(t)$ and the real output $y_m(t)$ ($y_{m1}=C_1$, $y_{m2}=C_2$). We have implemented the command synthesis in function of each estimated steady state variables $x_i=C_i$ *(I=1…5),* controlled by inputs N_{ij} (see diagram from fig. 6).

The results are presented in figure 7.

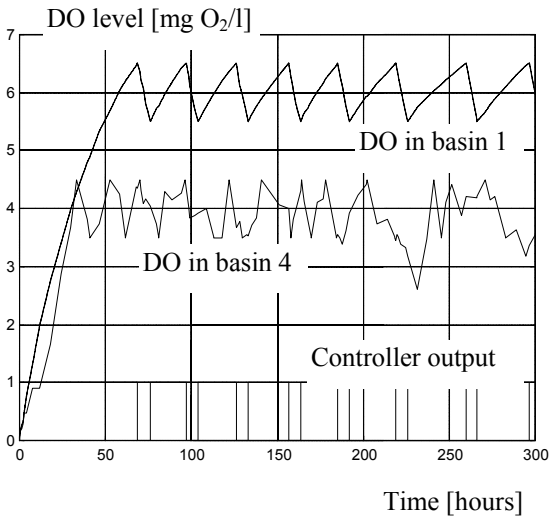


Figure 7: Experimental results I.

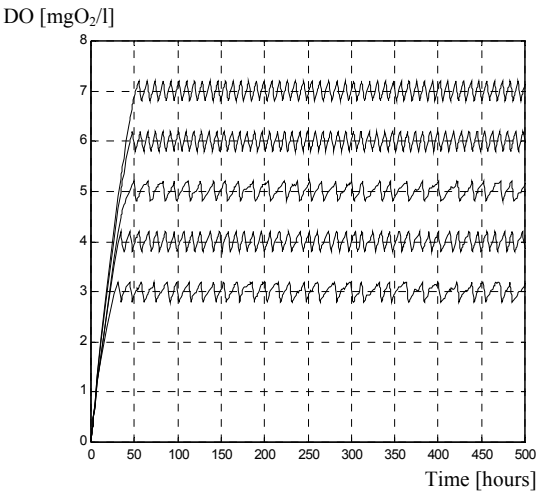


Figure 8: Experimental results II.

5 CONCLUSIONS

The results obtained using the proposed control structures to a Wastewater purification Plant are satisfactory. It causes a better performance of the plant because environmental law nearer to those requires the level of purification obtained. Also, the running costs have a notable reduction. The conventional tri-positional control structure is in implementation phase and we study the possibilities for advanced control structure. The results obtained till now establish the steps towards this objective.

REFERENCES

Chen W. C, Chang Ni-Bin, Shieh Wen K, Advanced hybrid fuzzy-neural controller for industrial wastewater treatment, Journal of environmental engineering , ISSN 0733-9372 CODEN JOEEDU , 2001, vol. 127, n°11, pp. 1048-1059

Chang Ni-Bin, Chen W. C, Shieh Wen K, Optimal control of wastewater treatment plants via integrated neural network and genetic algorithms, Civil engineering and environmental systems (Civ. eng. environ. syst.) ISSN 1028-6608, 2001, vol. 18, n°1, pp. 1-17

Demey D, Vanderhaegen V, Vanhooren H, Liessens J, Van Eyck L, Vanrolleghem P. A, Hopkins L., Validation and implementation of model based control strategies at an industrial wastewater treatment plant, Water science and technology (Water sci. technol.) ISSN 0273-1223, CODEN WSTED4 , 20011981, vol. 44, n° 2-3, pp. 145-153,

Henze M., Grady C.P.L., Gujer W., Marais G.V.R., (1987), *A general Model for Single-sludge Wastwater Treatment Systems,* Water Resourses, Vol. 20, pp. 505-515.

Ng Wun Jern, (2006), Industrial wastewater treatment, *National University of Singapore* , World Scientific Publishing, ISBN 978-1-86094-580-9

Peter A. Vanrolleghem, Models in Advanced Wastewater Treatment Plant Control (2003), Water Res. 2003 Jan;37(1):95-107, ELSEVIER

Urrutikoetxea A., Garcia de las Heras J.L., (1994), *Secondary Settling in Activated Sludge. Alab-scale Dynamic Model of Thickening,* Proceeding of the 8th Forum for Applied Biotechnology, Gent Coupure Links 653, B-9000, Gent Belgium.

Vinatoru M. , Niculescu E. ,Mihaiu M., (1979), The automatic compensation with the temperature of the oxyfen sensore –„3rd International Symposium on Control System and Computer Sciens – Bucureşti , , vol.III.p.654-660

Yusof R., Omatu S., (1993), *A multivariable self-tuning PID Controller,* Int. J. Control, Vol.57, No. 6, pp. 1387-1403.

APPENDIX

Notations and Symbols

V_i – volume of basin i (i=1-5) [m^3], F_{ei} - output flow from basin i (F_6=0) [m^3/h], F_{ia} – input wastewater flow in basin i [m^3/h] (F_{la}=0), F_{in} – input flow of activated sludge in basin i [m^3/h] ($F_{ln}=F_{2n}=F_{3n}=F_{nn}=0$), C_i – oxygen concentration in basin i [mgO_2/l], C_{in} - oxygen concentration in flow F_{in}, C_{ia} – oxygen concentration in flow F_{ia}, C_a– oxygen concentration at working temperature, K_{Au} – transfer coefficient in wastewater
F_{ri} – recirculation flow of aeration pumps [m^3/h]
C_{ri} – oxygen concentration in flow F_{ri}
γ -specific gravity of the medium inside the basin [N/m^3]
n - nominal rotational speed of the pumps, n_a – specific rotational speed of the pumps, h_i – immersion depth of the aerator [m], F_{pi} – pump flow [m^3/s], H_i –water level in the basin i [m], L_i –width of the separation wall [m], D_i – width of the transfer section between two basins, K_{pi} – ratio coefficient between pump flow F_{pi} and exit flow
S_i –area of horizontal section through the basin [m^2], N_i – umber of running pumps in basin i, m_c –shape coefficient, depending of the geometric shape of the dam, b –dam width, H –the liquid level above the dam, M_l –coefficient of velocity, depending of the access speed upstream of the dam, l_d –dam height [m].

MULTIPLE-MODEL DEAD-BEAT CONTROLLER IN CASE OF CONTROL SIGNAL CONSTRAINTS

Emil Garipov, Teodor Stoilkov
Technical University of Sofia, 1000 Sofia, Bulgaria
emgar@tu-sofia.bg, teodor.stoilkov@syscont.com

Ivan Kalaykov
Örebro University, 70182 Örebro, Sweden
ivan.kalaykov@tech.oru.se

Keywords: Dead-beat controller, multiple-model control, single-input single-output systems.

Abstract: The task of achieving a dead-beat control by a linear DB controller under control constraints is presented in this paper. Two algorithms using the concept of multiple-model systems are proposed and demonstrated - a multiple-model dead-beat (MMDB) controller with varying order using one sampling period and a MMDB controller with fixed order using several sampling periods. The advantages and disadvantages of these controllers are summarized.

1 INTRODUCTION

The Dead-Beat (DB) control problem in discrete time control theory consists of finding an input signal, which provides a transient response in a minimum number of sampling time steps. It has been studied by many researchers, e.g. (Jury, 1958), (Kucera, 1980), (Kaczorek, 1980), (Isermann, 1981), etc. If an n^{th} order linear system is null controllable, this minimum number of steps is n, as the applied feedback provides all poles of the closed-loop transfer function at the z-plane origin. The linear case is easy to solve, but DB control for non-linear systems is an open research problem (Nesic et al., 1998).

The DB controller of normal order (Isermann, 1981), denoted as DB(n,d), provides a constant control action after $n_s = (n+d)$ sampling steps, where d is the plant delay. For small sampling period the linear DB(n,d) controller forms extremely high control values at the first and second sampling steps after a step change of the system reference signal. In general, the control valve constrains the control signal, so these high amplitudes cannot be passed to the plant, thus making the system to be non-linear.

One way to solve the problem of constrained control signal, and still keeping the system as linear, is to prolong the transient response by increasing the controller order n_s. Isermann (1981) suggested increased by one order DB(n,d,1) controller, so the transient response takes $n_s = (n+d+1)$ sampling steps with decreased control value compared to the DB(n,d). This approach did not have essential practical application, but suggested two ideas:

- a higher controller order reduces the maximal amplitude of the control action;

- linear dead-beat control can be achieved by flexible tuning of the controller numerator coefficients.

In (Garipov and Kalaykov, 1991) an approach for design of adaptive DB(n,d,m) controller is presented, where the order increment m is sequentially changed until the control signal fits the control constraints. The reduction of the control magnitude pays off the prolongation of the transient response, as the signal energy distributes in more sampling time steps. Another approach is to increase the system sampling period without losing information. A control system with two sampling periods is proposed in (Garipov and Stoilkov, 2004) as a compromise solution.

These last two above mentioned approaches are useful for generalizing them by merging and involving various aspects of the multiple-model concept, as presented in (Murray-Smith and Johansen, 1997). In the present paper the task is solved by multiple-model dead-beat controller (MMDB) for one fixed and several sampling periods of the control system.

In Section 2 we present the theoretical base for design of DB controller of increased order. In Section 3 we describe the operation principle of DB control based on two sampling periods. In Section 4 the MMDB controller concept is developed in two variants. The first is based on a set of DB controllers of increased order in a system with one sampling period.

The second is utilizing a set of normal order DB controllers designed for several sampling periods. The concluding section summarizes the main properties of the proposed DB controllers.

2 DESIGN OF DB CONTROLLER OF INCREASED ORDER

Let the control plant description be:

$$W_o(z) = \frac{B(z)}{A(z)} z^{-d} =$$

$$= \frac{b_1 z^{-1} + b_2 z^{-2} + ... + b_n z^{-n}}{1 + a_1 z^{-1} + a_2 z^{-2} + ... + a_n z^{-n}} z^{-d} \qquad (1)$$

According (Garipov and Kalaykov, 1991) the designed DB(n,d,m) controller is

$$W_p(z) = \frac{Q(z)}{1 - z^{-d} P(z)} =$$

$$= \frac{q_0 + q_1 z^{-1} + ... + q_{n+m} z^{-(n+m)}}{1 - z^{-d}(p_1 z^{-1} + p_2 z^{-2} + ... + p_{n+m} z^{-(n+m)})} \qquad (2)$$

The vector θ of (2n+2m+1) unknown coefficients of the DB controller can be determined from the following matrix equation

$$X\,\theta = Y, \qquad (3)$$

$$X = \begin{bmatrix} & X^* & \\ \cdots & \cdots & \cdots \\ D_z & \vdots & Z \end{bmatrix}, Y = \begin{bmatrix} Y^* \\ \cdots \\ D_y \end{bmatrix}, \theta = \begin{bmatrix} p^{(1)} \\ p^{(2)} \\ q^{(1)} \\ q^{(2)} \end{bmatrix},$$

$$X^* = \begin{bmatrix} & E_1 & & \vdots & & D_e & \\ \cdots & \cdots & \cdots & \vdots & \cdots & \cdots & \cdots \\ & A_1 & & \vdots & & -B_1 & \\ \cdots & \cdots & \cdots & \vdots & \cdots & \cdots & \cdots \\ D_a & \vdots & A_2 & \vdots & D_b & \vdots & -B_2 \end{bmatrix},$$

$$\dim X^* = (2n+m+1) \times (2n+2m+1),$$

$$Y^* = \begin{bmatrix} 1 \\ 0 \\ . \\ 0 \end{bmatrix}, p^{(1)} = \begin{bmatrix} p_1 \\ p_2 \\ . \\ p_m \end{bmatrix}, p^{(2)} = \begin{bmatrix} p_{1+m} \\ p_{2+m} \\ . \\ p_{n+m} \end{bmatrix},$$

$$q^{(1)} = \begin{bmatrix} q_0 \\ q_1 \\ . \\ q_m \end{bmatrix}, q^{(2)} = \begin{bmatrix} q_{1+m} \\ q_{2+m} \\ . \\ q_{n+m} \end{bmatrix}.$$

$$\dim Y^* = (2n+m+1) \times 1,$$

$$A_1 = \begin{bmatrix} a_0 & 0 & . & . & . & . & 0 \\ a_1 & a_0 & 0 & . & . & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ a_n & a_{n-1} & . & a_0 & 0 & . & 0 \\ 0 & a_n & . & . & a_0 & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & 0 & . & a_n & a_{n-1} & . & a_0 \end{bmatrix},$$

$$\dim A_1 = (n+m) \times (n+m),$$

$$B_1 = \begin{bmatrix} b_1 & 0 & . & . & . & . & 0 & 0 \\ b_2 & b_1 & 0 & . & . & . & 0 & 0 \\ . & . & . & . & . & . & . & . \\ b_n & b_{n-1} & . & b_1 & . & . & 0 & 0 \\ 0 & b_n & . & . & b_1 & 0 & . & 0 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ 0 & . & 0 & b_n & . & . & b_1 & 0 \end{bmatrix},$$

$$\dim B_1 = (n+m) \times (n+m+1),$$

$$A_2 = \begin{bmatrix} a_n & a_{n-1} & . & a_1 \\ 0 & a_n & a_{n-1} & a_2 \\ . & . & . & . \\ 0 & 0 & . & a_n \end{bmatrix},$$

$$B_2 = \begin{bmatrix} b_n & b_{n-1} & . & b_1 \\ 0 & b_n & . & b_2 \\ 0 & 0 & . & . \\ 0 & 0 & . & b_n \end{bmatrix}$$

$$\dim A_2 = n \times n, \dim B_2 = n \times n,$$

$$E_1 = [\,1\ 1\ 1\ \ldots\ 1],$$

D_e, D_a, D_z, D_y are matrices with zero elements, $\dim D_e = 1 \times (n+m+1)$, $\dim D_a = n \times m$, $\dim D_b = n \times (m+1)$, $\dim D_z = m \times (n+m)$, $\dim D_y = m \times 1$.

The only solution of (3), which is the goal of dead-beat controller design task, is achieved when the rank of the linear system (3) is full. In fact this depends on the initially undetermined block matrix **Z,** $\dim Z = m \times (n+m+1)$. The z_{ij} values can be chosen in accordance with intention of the designer to guarantee desired control $u(k)$ such that additional m behavior conditions based on the following dependencies between parameters and signals:

a) When step change of the reference signal takes place at the k^{th} sampling step, the DB controller normally produces the largest positive amplitude $u(k)$ at k^{th} sampling step, followed by a smaller and negative value $u(k+1)$ at $(k+1)^{th}$ sampling step. Therefore, if the signal energy after the k^{th} sampling step is distributed over two or more sampling steps, holding the control signal, the large control magnitudes will be reduced (Isermann, 1981). This can be described by the inequality

$$Mod\{u(k+i)\}|_{u(k+i+1)=u(k+i)} < Mod\{u(k+i)\}|_{u(k+i+1)\neq u(k+i)}$$

$i = 0, 1, ...$, which should be related to the initially determined physical constraints on the control $u(k)$.

b) The matrix **Z** is needed only for dead-beat controllers of increased order, i.e. only when $m > 1$. Each row of it consists of one additional simple condition based on Isermann's idea for holding the previous value of the control signal

$$u(k+i+1) = u(k+i), \quad i = 0,\ 1,\ ..., \qquad (4)$$

for certain number of time steps. According (Garipov and Kalaykov, 1991), such behavior can be obtained by properly setting the coefficients of the polynomial $Q(z)$ of $(n+m)^{th}$ order. As always $q_0 \neq 0$ and $q_{n+m} \neq 0$, if we set $q_{i+1} = 0$ we obtain the desired condition $u(k+i+1) = u(k+i)$. Therefore the values z_{ij} play a special role of pointing which coefficient q_{i+1} is selected to be zero. When all values z_{ij} =0, it is assumed all coefficients $q_{i+1}$are nonzero. Therefore, first we have to zero the matrix **Z** and then set one unit value in the rows of **Z**. More details for how to select the values are given in (Garipov and Kalaykov, 1991).

c) If we want to hold the control signal longer time according condition (4), we have to zero more neighbor coefficients in $Q(z)$ by manipulating two or more neighbor rows of **Z**.

As an illustrative example let us take a plant with a continuous transfer function

$$W_o(s) = \frac{2s+1}{(10s+1)(7s+1)(3s+1)} e^{-4s}$$

For a sampling period T_o= 4 sec. we get

$$W_o(z) = \frac{0.06525z^{-1} + 0.04793z^{-2} - 0.00750z^{-3}}{1 - 1.49863z^{-1} + 0.70409z^{-2} - 0.09978z^{-3}} z^{-1},$$

$n_a = n_b = n = 3,\ d = 1$

Three dead-beat controllers with different structures: DB(3,1,0), DB(3,1,1) – three variants and DB(3,1,2) – six variants are designed according to the approach (Garipov and Kalaykov, 1991). It these variants some of the $Q(z)$ coefficients were zeroed. Obviously, the bigger is *m* the more variants of zeroing exist. Table 1 represents the maximum and minimum control values of the control signal during the transient response. The normal order DB controller (*m*=0) provides the largest values, while *variant1* when $m = 1$ and $m = 2$ provide significantly smaller values, which could fit to the control signal constraints.

Table 1: Max and min control values for the example.

m	Variant #	$u_{\max}$	$u_{\min}$
0		***9.46***	-4.71
1	variant1	***3.78***	-2.05
1	variant2	6.43	-0.18
1	variant3	8.28	-2.95
2	variant1	***2.34***	-0.83
2	variant2	3.01	0.28
2	variant3	3.49	-0.14
2	variant4	5.13	0.62
2	variant5	5.94	0.12
2	variant6	5.94	-2.27

3 DEAD-BEAT CONTROLLER IN A SYSTEM WITH TWO DIFFERENT SAMPLING PERIODS

The concept of DB controller of increased order, as described in the previous section, is one way of holding the control signal during more sampling steps of the transient response and consequently redistributing the signal energy in time. In this section we present an alternative approach employing nearly the same idea for redistributing the signal energy in time. To prolong the transient response and still keep the system null controllable, we can increase the sampling period for which we design a DB controller of normal order DB(*n*,*d*,0), but implement this controller in a system operating at smaller sampling rate. The concept (Garipov and Stoilkov, 2004) can be demonstrated by the discrete-continuous control system with two different sampling periods as shown on Fig.1. In fact this is a kind of internal model control (IMC) scheme, the inner loop of which is designed for a large sampling interval, and the outer loop is operating a small sampling interval. The main idea is that the main controller should work at the large sampling interval, thus redistributing the control signal energy in time and providing smaller control signal magnitude. But at the same time the entire system should operate at smaller sampling interval, therefore a correction signal from the plant-model difference should close the system.

The "Discrete Controller" block provides the control *u* to the "Continuous Plant" block (assumed to be linear with known time delay). Two different sampling periods are introduced:

- ***small sampling period*** T_0^{CS}, which is fundamental for the entire system, meaning that all signals are sampled and propagate at this period;
- ***large sampling period*** $T_0^{Reg} = l.T_0^{CS},\ l > 1$,used

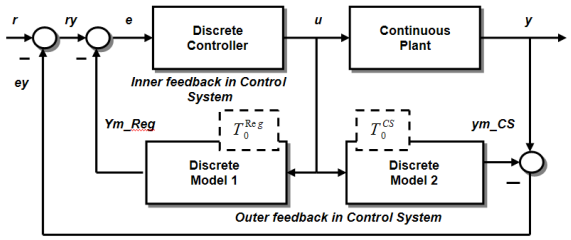


Figure 1: Discrete-continuous control system operating with two different sampling periods.

to define "Discrete Model 1" and respectively in the design of the "Discrete Controller" block.

In fact, the system contains two feedback loops:

- ***outer loop***, which forms corrected reference signal $ry = r - ey$ by the error $ey = y - ym\_CS$ between the measured output y of the "Continuous Plant" and the calculated output ym_CS of the "Discrete Model 2";
- ***inner loop***, forming the error $e = ry - ym\_Reg$ in the system between corrected reference ry and calculated output ym_Reg of "Discrete Model 1".

As an illustrative example let us take the same system given in Section 2. If we select a small sampling period T_o=0.1 sec, the normal order DB(n,d,0) controller produces extremely high control signal amplitude $u(0) = 216130$ after the unit step change of the reference signal. Obviously this value will be "clipped" by the control valve and the system performance will deteriorate. We decide to keep $T_0^{CS} = 0.1$ sec as a fundamental sampling period for the entire system, but introduce a second large sampling period $T_0^{Reg} = 8$ sec for which a DB controller is designed. Even $T_0^{Reg} = 8$ sec does not seem to be good choice, we intentionally use here for illustration. Hence, in the inner loop we have to use the "Discrete Model1", which is sampled at $T_0^{Reg} = 8$ sec, for providing proper control signal behavior. The outer loop is to correct the reference signal depending on the "Discrete Model2" operating at $T_0^{CS} = 0.1$ sec (nearly continuous-time control). The designed DB Controller for T_0^{Reg}=8 sec is:

$$W_o(z) = \frac{2.8653 - 2.4004z^{-1} + 0.5635 - 2 - 0.0285z^{-3}}{1 - 0.6045z^{-1} - 0.3991z^{-2} + 0.0036z^{-3}},$$

The first numerator coefficient q_o=2.8653 is equal theoretically to the control value $u(0)$. Fig. 2 demonstrates the controlled output (top) and the control signal (bottom), which has acceptable amplitude $u(0)$=2.8653 exactly as expected. The finite transient response takes 24 sec that is exactly three times T_0^{Reg}, as the system is of third order.

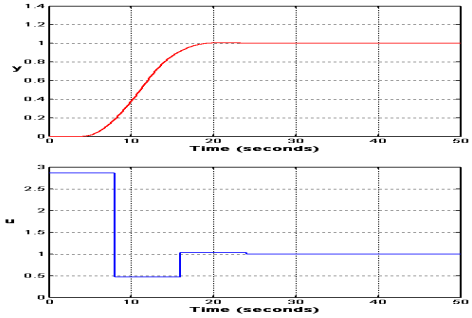


Figure 2: System with sampling period $T_0^{CS.} = 0,1$ sec and DB controller, designed for $T_0^{Reg} = 8$ sec.

4 MULTIPLE-MODEL DEADBEAT CONTROLLER

4.1 MMDB Controller with Varying Order using One Sampling Period

The existence of control signal constraints by the control valve clearly indicates the needs to guarantee a control magnitude that always fits within the control constraints for all operating regime of the system. The closer is the operating point to the constraints the bigger should be the DB controller order, as already clarified in Section 2. Obviously increasing the order the transient response becomes longer, but it is more important to keep the control signal within the constraints paying with the longer finite time of the response. As the plant operating point continuously changes, we should select the minimal order of the DB controller that satisfies the control signal constraints. So we came to the idea of building a MMDB controller that combines several DB controllers of different order running in parallel. The MMDB consists of two major parts:

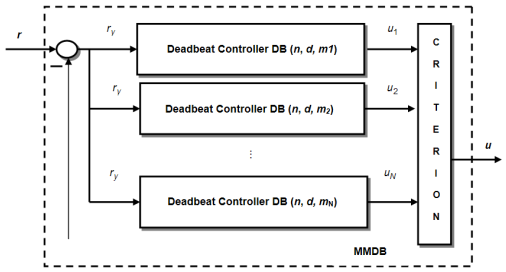


Figure 3: Structure of the MMDB.

- a set of N DB(n,d,m) controllers for the given model of the controlled plant, each of which is designed for different values of m, namely m_1, m_2, ..., m_N, such that all they provide constrained control signal within

the constraints of the control valve $[u_{\min}, u_{\max}]$for all possible variations of the reference signal; one sampling interval is assumed;

- a "criterion" block that switches the input of the plant to the output of one of the DB(n,d,m) controllers depending on a predefined set of conditions, in this case checking the output of which of the DB(n.d,m) controllers is within the constraints $[u_{\min}, u_{\max}]$. Additional criterion is to select the individual DB controller having the minimal value of m_i, because then the transient response is of minimum duration. As

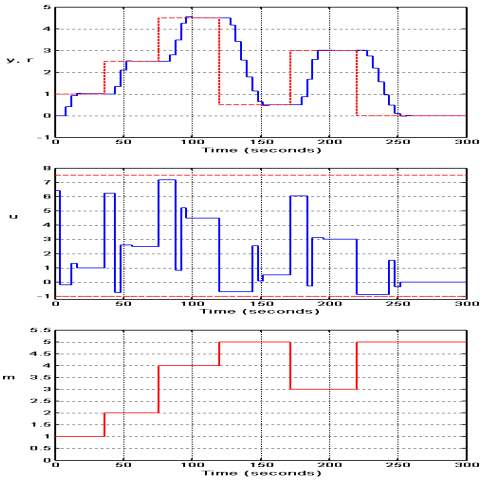


Figure 4: Reference signal and plant output (top); control signal within the constraints (middle); increment of the DB controller order (bottom).

an example we designed a MMDB controller for the plant described in Section 2 with sampling period $T_0^{Reg} = 4$sec. A set of DB controllers is included, namely DB(3,1,m), m = 0, 1, 2, 3, 4 and 5. On Fig. 4 the transient response of the plant follows the reference signal, but is stepwise as the sampling period is big. The control signal lies within the constraints. The "criterion" block decides to switch the appropriate DB(n,d,m_j) controller such that the constraints are satisfied, as seen on the bottom picture on Fig. 4. The "criterion" block is selecting an individual controller with higher or smaller order depending on the distance of the plant operating regime to the control constraints and the step change magnitude of the reference.

The important property of the proposed MMDB controller is the embedded flexibility to select the appropriate order of the DB controller. For comparison on Fig. 5 we present the performance of fixed DB(3,1,0) and DB(3,1,1) controllers at the same operating conditions. Obviously the transient response does not represent a deadbeat behavior as a result of applying too low DB controller order, which cannot bring the control signal within the constraints.

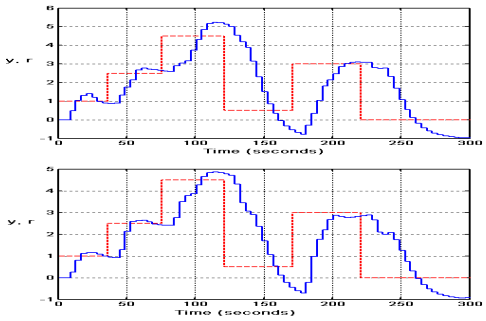


Figure 5: Plant output and reference signal for DB(3,1,0) (top) and DB(3,1,1) (bottom) controller.

4.2 MMDB Controller with Fixed Order using Several Sampling Periods

Contrary to the concept presented in Section 4, here we suggest a MMDB controller that contains a number of controllers, each of which is designed for different sampling periods $T_0^{Reg_i}$, i=1, 2, ..., N, assuming that the entire control system operates with a sampling period $T_0^{CS} << T_0^{Reg_i}$, as shown on Fig. 6.

The difference between this MMDB and the MMDB on Fig. 3 is the content of the individual DB controllers. Here they are assumed of DB(n,d,0) type (normal order DB controller), but they differ due to the different sampling period used for their design. Generally, there is no limitation to use DB(n,d,m) type controllers as well, but for simplicity m is not considered to be a parameter of choice. As an exam-

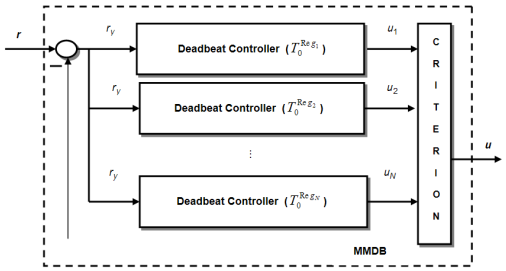


Figure 6: Structure of the MMDB.

ple we demonstrate a MMDB controller for the plant described in Section 2 with sampling period T_0^{CS}= 0.1 sec. A set of DB controllers is designed for T_0^{REG}= 4, 6, 8, 10, 12, 14 and 18 sec. The performance of the system is shown on Fig. 7. One can see that the transient response of the plant follows the reference signal and is rather smooth due to the small sampling period of the entire system. The control signal lies within the constraints. On the bottom picture on Fig.

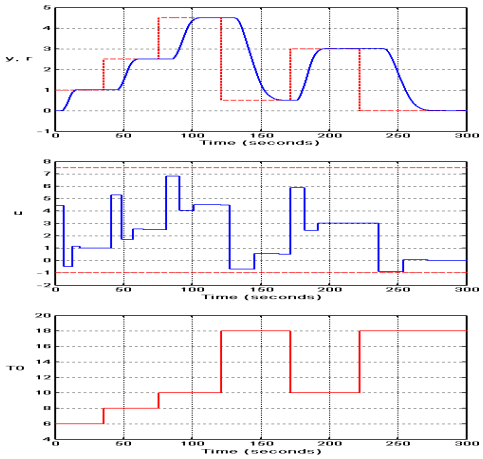


Figure 7: Reference signal and plant output (top); control signal within the constraints (middle); sampling period of the DB controller (bottom).

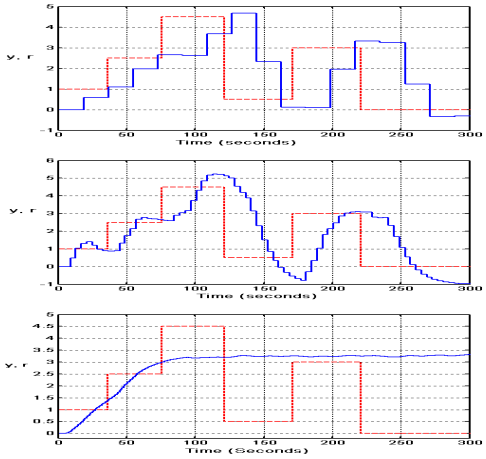


Figure 8: Plant output and reference signal for: $T_0^{Reg} = T_0^{CS}$=18 sec (top); $T_0^{Reg} = T_0^{CS}$=4 sec (middle); $T_0^{Reg} = T_0^{CS}$= 0.1 sec (bottom).

7 it can be seen that the "criterion" block is selecting an individual controller designed for bigger higher or smaller sampling period depending on the distance of the plant operating regime to the control constraints and the magnitude of the step change of the reference signal.

The important property of the proposed MMDB controller with fixed order is the possibility to select the appropriate sampling period of the DB controller that keeps the control signal within the constraints. For comparison on Fig. 8 we present the performance of fixed DB(3,1,0) controller designed and implemented at the same sampling period $T_0^{Reg} = T_0^{CS}$ and the same operating conditions. Obviously the transient response does not represent a deadbeat behavior as a result of applying too low DB controller order, which cannot bring the control signal within the constraints.

5 CONCLUSION

Two original ideas for solving the task of achieving a dead-beat control by a linear DB controller under control constraints were presented in this paper: for design of DB controllers of increased order and for implementation of a discrete-continuous control system, which operates with two different sampling periods. Two algorithms using the concept of multiple-model systems were proposed and demonstrated – a MMDB controller with varying order using one sampling period and a MMDB controller with fixed order using several sampling periods. Both algorithms provide normal operating of the control system and control signal does not leave the predefined constrains. Numerical simulations confirm the performance of the proposed algorithms.

The advantages and disadvantages of these controllers are summarized in Table 2, which can be a useful tool for selection of DB controllers in practical applications.

ACKNOWLEDGEMENTS

The third author acknowledges the support of the Swedish KKS Foundation for part of this research.

REFERENCES

Garipov, E. and Kalaykov, I. (1991). Design of a class robust self-tuning controllers. In *Prepr. of IFAC Symp. on Design Methods*.

Garipov, E. and Stoilkov, T. (2004). Multiple-model deadbeat controller in control systems with variable sampling period. In *Annual Proc. of the Technical University Sofia*.

Isermann, R. (1981). *Digital Control Systems*. Springer Verlag, Berlin.

Jury, E. (1958). *Sampled-Data Control Systems*. Wiley, New York.

Kaczorek, T. (1980). Deadbeat control of single-input single-output linear time-invariant systems. *Int. J. Syst. Sci.*, 11:411–421.

Kucera, V. (1980). A dead-beat servo problem. *International Journal of Control*, 32:107–113.

Murray-Smith, R. and Johansen, T. A. (1997). *Multiple Model Approaches to Modeling and Control*. Taylor and Francis, London.

Nesic, D., Mareels, M., Bastin, G., and Mahony, R. (1998). Output dead beat control for a class of planar polynomial systems. *SIAM J. Control Optim.*, 36:253–272.

Table 2: Basic properties of the Dead-beat controllers.

Controller	Advantages	Disadvantages
• DB controller of ***normal order***, system with one model and ***one sampling period***	• Easy tuning of the controller with small design efforts.	• Large control amplitudes for models of low order and small time delays for small sampling period. • Rough response to the reference signal when big sampling period is used. • No adaptive properties when changing the operating regimes of the control system.
• DB controller of ***increased order***, system with one model and ***one sampling period***	• Possibility of multi variant tuning. • Good control and significant reduction of the large control amplitudes at the first few sampling steps. • Smoother response even for small sampling period, due to the increased controller order.	• Relatively complex design algorithm. • Higher order of the controller needed to reduce the large control amplitudes. • No adaptive properties when changing the operating regimes of the control system.
• DB controller of ***normal order***, system with one model and ***two different sampling periods***	• Simple controller design algorithm. • Good control and significant reduction of the large control amplitudes at the first few sampling steps. • Smoother response to the reference signal even for small sampling period, due to the increased controller order.	• Complicated scheme of the control system. • No adaptive properties when changing the operating regimes of the control system.
• MMDB controller using ***increased order DB blocks***, system with ***one sampling period***	• Adaptation to changes in operating regimes of the control system in case of complex profile of the reference signal and controller output constraints. • Good control and significant reduction of the large control amplitudes at the first few sampling steps. • Smoother response even for small sampling period, due to the increased controller order.	• Relatively complex design algorithm. • Complicated scheme of the control system, as several DB controllers with different fixed structures but with one sampling period function at different operating points of the control system. • Need of supervisor for switching between various controllers.
• MMDB controller using ***normal order DB blocks***, system with ***several sampling periods***	• Adaptation to changes in operating regimes of the control system in case of complex profile of the reference signal and controller output constraints. • Good control and significant reduction of the large control amplitudes at the first few sampling steps. • Smoother response even for small sampling period, due to the increased controller order. • Simple algorithm for designing DB controller of normal order.	• Complicated scheme of the control system, as several DB controllers with different fixed structures but with one sampling period function at different operating points of the control system. • Need of supervisor for switching between various controllers.

WEBMATHEMATICA BASED TOOLS FOR NONLINEAR CONTROL SYSTEMS

Heli Rennik, Maris Tõnso and Ülle Kotta
Institute of Cybernetics, Tallinn University of Technology, Akadeemia tee 21, Tallinn, 12818, Estonia
heli.rennik@mail.ee, mtonso@staff.ttu.ee, kotta@cc.ioc.ee

Keywords: Web-based education, webmathematica, nonlinear control, linear algebraic framework.

Abstract: Algebraic approach of differential one-forms provides simple theoretical framework for several typical problems of nonlinear control theory that makes it useful for educational purposes. Additional assistance is provided by Mathematica functions, developed by us and made available by creating a web-based application using web-Mathematica. These symbolic computation tools provide solutions for several modelling and anlysis problems like accessibility, identifiability, realizability and realization and require no other software except for an internet browser.

1 INTRODUCTION

Over the many years we have developed a software package NLControl in Mathematica for nonlinear control systems (Kotta and Tõnso, 1999; Kotta and Tõnso, 2003), based on algebraic approach of differential one-forms (Aranda-Bricaire et al., 1996; Kotta et al., 2001; Conte et al., 1999). This package provides basic tools for modelling, analysis and synthesis both of discrete- and continuous-time systems.

The Mathematica functions developed by us and described partly in (Kotta and Tõnso, 1999; Kotta and Tõnso, 2003) cannot be used outside of Mathematica environment. Our task was to make our tools available via the world-wide-web, in such a way that no other software except for an internet browser needs to be installed in a computer to use these tools. Recently, we developed a first set of web-based tools. The reason for this was twofold. First, the computer has become an integral part of the educational process (Moog et al., 2003a). Second, we want to make the tools, developed by us, available to a larger control community. The other web-based tool for nonlinear control system is described in (Ondera and Huba, 2005). Since the intention was to create a web-based application that would re-use the original program code and whose outputs would mimic the original tools as much as possible, we used webMathematica.

The paper is organized as follows. Section 2 describes webMathematica technology both the frontend, which is available to everyone over the web, and web server technology, which includes description of kernels, kernel pools and configuration options. Section 3 provides a description of nonlinear control systems, both discrete- and continuous-time, together with the necessary assumptions and our basic tool to handle the considered problems - the sequence of subspace H_k of one forms, associated to the nonlinear control system. Next, a brief description of modeling and analysis problems, implemented in our webMathematica website is given together with some simple examples and program codes. Finally, the last section concludes the paper by discussing our contribution and providing some future perspectives.

2 WEB-BASED IMPLEMENTATION

This section describes a webMathematica website for nonlinear control systems, both discrete- and continuous-time. We have implemented several Mathematica functions programmed by us to web-Mathematica for public use. WebMathematica is an option to make our functions available for students

and science community without revealing our programming code. Our webMathematica website is simple, including at moment five different function pages for discrete- and continuous-time systems described either by input-output equations or by state space equations.

2.1 About Webmathematica Frontend

WebMathematica frontend is a web based user interface for Mathematica. WebMathematica adds interactive calculations and visualizations to a website by integrating Mathematica with web server technology. WebMathematica makes all Mathematica functionality available over the web and is easy to use even for non-professional programmer. For using webMathematica one has to know HTML basics and use a web browser.

Websites (or in other words user interfaces) can use standard web graphical user interface elements, such as text fields, check boxes, and drop-down lists. The major browsers, like Internet Explorer, Mozilla and Netscape Navigator support webMathematica. WebMathematica allows a site to deliver HTML pages that are enhanced by the addition of Mathematica commands and uses the request/response standard followed by web servers. The process how webMahtematica works is following:

1. Browser sends a request to webMathematica server.
2. WebMathematica server acquires Mathematica kernel from the pool.
3. Mathematica kernel is initialized with input parameters, it carries out calculations, and returns result to server.
4. WebMathematica server returns Mathematica kernel to the pool.
5. WebMathematica server returns result to Browser.

Requests are sent to the server with webMathematica web pages that are based on two standard Java technologies: Java Servlet and JavaServer Pages (JSP) technologies. JavaServer Pages (JSPs) use a special library of tags that work with Mathematica. These tags have the form $\langle msp : tag \rangle$. The $\langle msp : allocate \rangle$ tag causes a Mathematica kernel to be allocated to use for computations. The contents of the $\langle msp : evaluate \rangle$ tags are sent to Mathematica for computation with the result inserted into the final page. The $\langle /msp : allocate \rangle$ tag frees the Mathematica kernel to be used for another computation.The library of tags is called the MSP Taglib. In our site we use also JavaScript and Java. We use JavaScript for opening and closing windows and communicating between windows and Java for calculating random inputs.

2.2 Webmathematica Web Server Technology

There are many different combinations of hardware and operating systems that support webMathematica components, for example Windows, Linux, Solaris, Mac OS X. Before one starts to install webMathematica, one has to install Java and a servlet container. WebMathematica is tested with Apache Tomcat and JRun. We are using Linux operating system and Tomcat as a web container.

WebMathematica server can be configured as necessary. The number of pools of Mathematica kernels can be specified. Also one can set the number of kernels that are launched when the system starts. The parameters, as the number of times that each kernel can be taken from the kernel pool before being shut down, can be specified in order to clean it up from unnecessary processes. It is a good idea to shut down each kernel at a regular time-period. Another parameter specifies the maximum number of milliseconds for processing a page. When this time is exceeded, the kernel is shut down and restarted. This is useful in case one is computing large operations.

We are using only one kernel and one kernel pool and we have got problems with functions that do not work properly. They intend to let Tomcat ending up crashing. We have specified the time 30 seconds in which all responses must be got from the server. When the web site is made available for the students, we will increase the number of kernels.

3 IMPLEMENTED FUNCTIONS

Several different functions programmed by us are gathered into one Mathematica package called NLControl for solving different control problems. At moment we have implemented five functions from this package into webMathematica website. These functions are `Submersivity`, `SequenceHk`, `Realization`, `Accessibility` and `Identifiability`. We have chosen functions that are based on subspaces H_k and cover different modeling and analysis problems.

Consider the continuous-time

$$\dot{x} = f(x,u) \tag{1}$$

or the discrete-time nonlinear control system

$$x^+ = f(x,u), \tag{2}$$

respectively, where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, f is an analytic function and x^+ denotes the forward shift of x. In order to make computations with these equations in Mathematica, we have created two special objects for NLControl package: `CStateSpace[f,x,u,t]` and `DStateSpace[f,x,u,t]`, that represent systems (1) and (2), respectively. Objects `CIO[f,u,y,t]` and `DIO[f,u,y,t]` are used for representing continuous- and discrete-time input/output equations, respectively.`DStateSpace, CStateSpace, DIO` and `CIO` are used by all the implemented functions `Submersivity, SequenceHk, Realization, Accessibility` and `Indentifiability`.

3.1 Submersivity

The function f in (2) is said to be a submersion, if generically (i.e. everywhere, except on a set of measure zero)

$$\operatorname{rank}\frac{\partial f}{\partial(x,u)} = n. \tag{3}$$

Submersivity assumption has to be satisfied for discrete-time control systems for the other functions to be applicable. `Submersivity` gives `True` if discrete-time state equations are submersive. If the result is `False` then the other Mathematica functions cannot be used since they may yield wrong results.

For checking `Submersivity` assumption the system has to be given in the state space form with the list of state and input variables.

The Mathematica block sent to the server looks as follows:

```
<msp:evaluate>
   MSPBlock[ {$$f, $$Xt, $$Ut},
      MSPFormat[ Submersivity[
         DStateSpace[$$f, $$Xt, $$Ut, t]],
         OutputForm
      ]
   ]
</msp:evaluate>
```

Example: Consider the bioreactor model, where cells are being grown through the consumption of a substrate (Kazantzis and Kravaris, 2001):

$$\begin{aligned} x_1(t+1) &= x_1(t) + [\tfrac{x_1(t)x_2(t)}{x_1(t)+x_2(t)} - 0.08x_1(t)]T \\ x_2(t+1) &= x_2(t) + [\tfrac{-x_1(t)x_2(t)}{x_1(t)+x_2(t)} - 0.08x_2(t) \\ &\quad +0.008]T \end{aligned} \tag{4}$$

The result is `True`.

3.2 SequenceHk

Let $\mathcal{K}$ be the field of meromorphic functions in a finite number of system variables $\{x, u^{(k)}, k \geq 0\}$ or $\{x(0), u(k), k \geq 0\}$ for the continuous- and discrete-time system, respectively. Over the field $\mathcal{K}$ one can define a vector space $\mathcal{E} := \operatorname{span}_{\mathcal{K}}\{\mathrm{d}\varphi \mid \varphi \in \mathcal{K}\}$. The elements of $\mathcal{E}$ are called one-forms. The relative degree r of one-form ω in $\mathcal{X} = \operatorname{span}_{\mathcal{K}}\{\mathrm{d}x\}$ is given by $r = \min\{k \in \mathbb{N} \mid \operatorname{span}_{\mathcal{K}}\{\omega, \ldots, \omega^{(k)}\} \not\subset \mathcal{X}\}$ or $r = \min\{k \in \mathbb{N} \mid \operatorname{span}_{\mathcal{K}}\{\omega(0), \ldots, \omega(k)\} \not\subset \mathcal{X}\}$.

Let us define a decreasing sequence of subspaces $\mathcal{H}_0 \supset \mathcal{H}_1 \supset \mathcal{H}_2 \supset \ldots$ such that each $\mathcal{H}_k$, for $k > 0$, is the set of all one-forms with relative degree at least k(Aranda-Bricaire et al., 1996; Conte et al., 1999):

$$\begin{aligned} \mathcal{H}_0 &= \operatorname{span}_{\mathcal{K}}\{\mathrm{d}x, \mathrm{d}u\} \\ \mathcal{H}_k &= \{\omega \in \mathcal{H}_{k-1} \mid \dot{\omega} \in \mathcal{H}_{k-1}\} \end{aligned}$$

or

$$\mathcal{H}_k = \{\omega \in \mathcal{H}_{k-1} \mid \omega^+ \in \mathcal{H}_{k-1}\}.$$

There exists an integer $k^* > 0$ such that $\mathcal{H}_k \supset \mathcal{H}_{k+1}$, for $k \leq k^*$, and $\mathcal{H}_{k^*+1} = \mathcal{H}_{k^*+2} = \ldots \mathcal{H}_\infty$, $\mathcal{H}_{k^*} \not\supseteq \mathcal{H}_\infty$. The existence of k^* comes from the fact that each $\mathcal{H}_k$ is a finite dimensional vector space, so that at each step either the dimension decreases by at least one or $\mathcal{H}_{k+1} = \mathcal{H}_k$.

$\mathcal{H}_\infty$ contains the one-forms with infinite relative degree so that these one-forms will never be influenced by the control.

An one-form $\omega \in \mathcal{E}$ is exact if $\mathrm{d}\omega = 0$, and closed if $\mathrm{d}\omega \bigwedge \omega = 0$ where by $\bigwedge$ is denoted the wedge product. We say that the subspace is completely integrable if it admits the basis which consists only of closed one-forms.

The function `SequenceHk` computes first N elements in the sequence of subspaces Hk associated to state equations, where N is an positive integer specified by us. For running `SequenceHk`, the system has to be given in the state space form with the list of state and input variables.

The Mathematica block sent to the server looks as follows:

```
<msp:evaluate>
   MSPBlock[ {$$f, $$Xt, $$Ut},
      MSPFormat[ SequenceHk[
         DStateSpace[$$f, $$Xt, $$Ut, t], All],
         OutputForm, TimeArgument-> True
      ]
   ]
</msp:evaluate>
```

Example 1: Consider another bioreactor model, where the microorganisms grow by consuming the substrate (Benamor et al., 1997):

$$\begin{aligned} x_1(t+1) &= x_1(t) + T\tfrac{a_1x_1(t)x_2(t)}{a_2x_1(t)+x_2(t)} - Tu(t)x_1(t) \\ x_2(t+1) &= x_2(t) - T\tfrac{a_3a_1x_1(t)x_2(t)}{a_2x_1(t)+x_2(t)} - Tu(t)x_2(t) \\ &\quad +Ta_4u(t) \\ y(t) &= x_1(t) \end{aligned} \tag{5}$$

The result is the following:

$$\begin{aligned} \mathcal{H}_1 &= Span[dx_1(t), dx_2(t)] \\ \mathcal{H}_2 &= Span[dx_1(t)] \\ \mathcal{H}_3 &= \{0\} \end{aligned} \tag{6}$$

Example 2: The dynamic model of a current-driven induction motor expressing the rotor flux and the stator currents in a reference frame rotating at synchronous speed is given by the continuous-time state equations(Bazanella and Reginatto, 2000):

$$\begin{aligned} \dot{x}_1(t) &= -c_1 x_2(t) - u_1(t)x_2(t) + c_2 u_3(t) \\ \dot{x}_2(t) &= -c_1 x_2(t) + u_1(t)x_1(t) + c_2 u_2(t) \\ \dot{x}_3(t) &= -c_3 x_3(t) + c_4(c_5(x_2(t)u_3(t) - x_1(t)u_2(t)) \\ &\quad -Tm) \end{aligned} \tag{7}$$

The result is the following:

$$\begin{aligned} \mathcal{H}_1 &= Span[dx_1(t), dx_2(t), dx_3(t)] \\ \mathcal{H}_2 &= \{0\} \end{aligned} \tag{8}$$

3.3 Realization

Consider a higher order i/o differential

$$y^{(n)} = \Phi(y, \ldots, y^{(n-1)}, u, \ldots, u^{(s)}) \tag{9}$$

or difference equation

$$\begin{aligned} y(t+n) &= \Phi(y(t), \ldots, y(t+n-1), \\ &\quad u(t), \ldots, u(t+s)), \end{aligned} \tag{10}$$

where n and s are nonnegative integers, $s < n$ and Φ is a real analytic function. The realization problem is to construct the state equations of order n,

$$\dot{x} = f(x,u) \quad x^+ = f(x,u)$$

or

$$y = h(x) \qquad y = h(x)$$

for the i/o equation (9) and (10), respectively. It is important to stress that the state-space realization does not exist for every i/o equation. In order to find state equations, one has to compute the sequence $\mathcal{H}_k$ for the i/o equations (9) or (10). The detailed procedures can be found in (Kotta et al., 2001) and (Moog et al., 2003b), respectively.

Theorem 1. The i/o equation has a state-space realization iff for $1 \leq k \leq s+2$ the subspaces $\mathcal{H}_k$ are completely integrable. The state coordinates can be found by integrating the basis functions of $\mathcal{H}_{s+2}$.

Function `Realization` determines whether the nonlinear higher order input-output difference equation can be realized in the classical state-space form and the if the i/o equation is realizable, finds the state equations. For running `Realization` the system has to be given by the input-output equations and the list of input and output variables.

The Mathematica block sent to the server looks as follows:

```
<msp:evaluate>
   MSPBlock[ {$$eqs, $$Ut, $$Yt},
      MSPFormat[ Realization[
         DIO[$$eqs, $$Ut, $$Yt, t], [#][t]&],
         OutputForm, TimeArgument-> True
      ]
   ]
</msp:evaluate>
```

Example 1: The fed-batch bakers' yeast fermentation process 1 is described by the following i/o equations (Keulers et al., 1993).

$$\begin{aligned} y(t+1) = 0.9106y(t) + 2.072u(t+1) - 1.903u(t) \\ +120.7y(t)u(t+1) - 107.3y(t)u(t) + 299.6u(t+1)^2 \\ -232.8u(t+1)u(t) - 84.17u(t)^2 \end{aligned} \tag{11}$$

The result is that the classical state space form does not exist for system (11).

Example 2: Consider the continuous-time i/o equation:

$$\dddot{y}(t) = \ddot{y}(t)u(t) + \dot{u}(t)y(t) + u(t) \tag{12}$$

The classical state equations for (12) are:

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= u(t)x_1(t) + x_3(t) \\ \dot{x}_3(t) &= u(t)(1 + u(t)x_1(t) - x_2(t) + x_3(t)) \\ y(t) &= x_1(t) \end{aligned} \tag{13}$$

3.4 Accessibility

A function φ_r in $\mathcal{K}$ is said to be an autonomous variable for system (1) or (2), if there exist an integer $\mu \geq 1$ and a non-zero meromorphic function F so that $F(\varphi_r, s\varphi_r, \ldots, s^\mu\varphi_r) = 0$ or $F(\varphi_r, \delta\varphi_r, \ldots, \delta^\mu\varphi_r) = 0$ for continuous- and discrete-time system, respectively. The system (1) or (2) is said to be accessible if there does not exist any non-zero autonomous variable in $\mathcal{K}$.

Theorem 2. (Aranda-Bricaire et al., 1996; Conte et al., 1999) The nonlinear system is strongly accessible iff

$$\mathcal{H}_\infty = \{0\}. \tag{14}$$

Note that the *Theorem 1* and 2 holds both in continuous- and discrete-time cases though the rules to calculate the subspaces $\mathcal{H}_k$ are different.

`Accessibility` returns `True` if the nonlinear system is strongly accessible and `False` otherwise.

For computing `Accessibility` the system has to be given in the state space form with the list of state and input variables.

The Mathematica block sent to the server looks as follows:

```
<msp:evaluate>
   MSPBlock[ {$$f, $$Xt, $$Ut},
      MSPFormat[ Accessibility[
         DStateSpace [$$f, $$Xt, $$Ut, t]],
         OutputForm
      ]
   ]
</msp:evaluate>
```

Example 1: Consider a grain drying process (Kotta and Nurges, 1985).

$$\begin{array}{rcl} x_1(t+1) & = & -0.0081u(t)x_1(t)+x_2(t) \\ x_2(t+1) & = & -0.01772892u(t)x_1(t) \\ & + & 1.6332x_2(t)+x_3(t) \\ x_3(t+1) & = & (-0.1751-0.00360073u(t)) \\ & & x_1(t)-0.4567x_2(t) \end{array} \tag{15}$$

The result is `True`, that is system (15) is accessible.

Example 2: Consider a simple academic example:

$$\begin{array}{rcl} x_1(t) & = & x_1(t)(x_3^2(t)+1)^2 \\ x_2(t) & = & x_2(t)(x_3^2(t)+1)^3 \\ x_3(t) & = & x_3(t)+u(t) \end{array} \tag{16}$$

The result is `False`. $\mathcal{H}_\infty = span\{3x_2dx_1 - 2x_1dx_2\}$.

3.5 Identifiability

Identifiability property characterizes the possibility to find the unknown parameters of the system from the measured input-output data. Consider the continuous-time nonlinear system

$$\begin{array}{rcl} \dot{x} & = & f(x,u,\theta) \\ y & = & h(x,u,\theta), \end{array} \tag{17}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and the parameter vector $\theta \in \mathbb{R}^q$. There are two concepts for identifiability - with and without the knowledge of initial conditions. At the moment only identifiability without knowledge of the initial state is implemented on our website. Our website offers two alternative methods to check identifiability of the system. The first is algebraic method, that enables to find the parameters by solving certain algebraic equations depending only on input-output information, see (Xia and Moog, 2003). The second method checks if the parameters can be found by the least squares method (LSM). To check identifiability by the LSM one has to find the i/o representation $E(y,\dot{y},\ldots,y^{(n)},u,\dot{u},\ldots,u^{(s)},\theta) = 0$ for system (17). If the above i/o equation admits a separable form $\sum_{i=1}^{n_i} g_i(\theta)E_i(y,\dot{y},\ldots,y^{(n)},u,\dot{u},\ldots,u^{(s)})$ and the condition $g(\theta) = g(\theta^*) \Leftrightarrow \theta = \theta^*$ is satisfied, the least squares method described in (Pearson, 1989), is applicable for parameter identification.

For computing `Identifiability` the system has to be given in the state space form with the list of state, input and output variables.

The Mathematica block sent to the server looks as follows:

```
<msp:evaluate>
   method = Switch[ MSPToExpression[ $$method],
       algebraic, Algebraic, pearson, Pearson];
   MSPBlock[
      {$$f, $$Xt, $$Ut, $$h, $$Yt, $$theta},
      MSPFormat[ Identifiability[
         CStateSpace[f, Xt, Ut, t, h, Yt],
            {theta},Method->method],
         OutputForm
      ]
   ]
</msp:evaluate>
```

Example: Consider the simple academic example

$$\begin{array}{rcl} \dot{x}_1(t) & = & \theta_1 x_1(t)^2 + \theta_2 x_1(t)x_2(t) + u(t) \\ \dot{x}_2(t) & = & \theta_3 x_1(t)^2 + \theta_4 x_1(t)x_2(t) \\ y(t) & = & x_1 \end{array} \tag{18}$$

The system is identifiable neither with LSM nor with algebraic method.

3.6 Comparative Evaluation

In this section our web-based tools are compared by web-based tools for nonlinear control systems in (Ondera and Huba, 2005). Besides the fact the problems handled are different ((Ondera and Huba, 2005) is dedicated solely to the exact static state feedback linearization problem) there are other points to be mentioned.

Our web-site enables user to specify the system equations and calculate the state transformation and linearizing control law (work in progress, not described in this paper). The web-tools in (Ondera and Huba, 2005) allow the user additionally to submit a desired closed-loop poles of a pole-placement controller and to perform a simulation.

There is also a difference in chosen technology. The web-tools in (Ondera and Huba, 2005) are based on MATLAB and its Symbolic Math Toolbox. This toolbox is a Maple 8 symbolic kernel that was bought from Maplesoft and implemented into MATLAB by The MathWorks. The different platform also implies a different internet implementation. In (Ondera and Huba, 2005) tools are web-accessible via MATLAB Web Server that is based on CGI technology, whereas webMathematica is java and javascript-based.

Both web-tools relieve users from installing Mathematica or MATLAB on their computers and help to make programs available to everyone without seeing program code.

4 CONCLUSION

WebMathematica is a web version of Mathematica that uses a web server technology, HTML and Java Servlet Pages. Calculations entered via web pages are sent to kernel, where the result is calculated and sent back to web pages. Several different functions programmed by us are gathered into one Mathematica package called NLControl for solving different modeling, analysis and synthesis problems. At moment we have implemented five functions from this package into webMathematica website. These functions are `Submersivity`, `SequenceHk`, `Realization`, `Accessibility` and `Identifiability`. In the future we are expanding our website with functions `Linearization` and `PrimeForm`. The function `Linearization` checks if the state equations can be linearized via the static state feedback and coordinate transformation, and finds necessary transformations. The function `PrimeForm` transforms the system into the prime form, whenever possible, using the static state feedback, and the coordinate transformations in the state and output spaces. Besides continuous- and discrete-time systems we are also programming functions for systems, described on homogeneous time scales (Bartosiewicz et al., 2007).

ACKNOWLEDGEMENTS

This work was partially supported by the Estonia Science Foundation Grant No 6922.
The authors thank M. Ondera for discussions on comparative evaluation of web-based symbolic tools presented in this paper and in (Ondera and Huba, 2005).

REFERENCES

Aranda-Bricaire, E., Kotta, Ü., and Moog, C. (1996). Linearization of discrete-time systems. In *SIAM J. Control and Optimization*, volume 34, pages 1999–2023.

Bartosiewicz, Z., Ü. Kotta, Pawluszewicz, E., and Wyrwas, M. (2007). Irreducibility conditions for nonlinear input-output equations on homogeneous time scales. Submitted to IFAC Symp. on Nonlinear Control Systems.

Bazanella, A. and Reginatto, R. (2000). Robustness margins for indirect field oriented control of induction motors. In *IEEE Trans. Automatic Control*, volume 45, pages 1226–1231.

Benamor, S., Hammouri, H., and Couenne, F. (1997). A luenberger-like observer for discrete-time nonlinear systems. In *Proc. European Control Conf.*, Brussels, Belgium.

Conte, G., Moog, C., and Perdon, A. (1999). Nonlinear control systems. In *Lecture Notes in Control and Information Sciences*, volume N242, London. Springer.

Kazantzis, N. and Kravaris, C. (2001). Discrete-time nonlinear observer design using functional equations. In *Systems and Control Letters*, volume 42, pages 81–94.

Keulers, M., Sepp, K., Breur, A., and Reyman, G. (1993). A simulation study of nonlinear structure identification of a fed batch bakers' yeast process. In *Proc. American Control Conference*, pages 2256–2260, San Fransisco.

Kotta, Ü. and Nurges, Ü. (1985). Identification of input-output bilinear systems. In *Proc. 9th IFAC World Congress*, volume 2, pages 723–727. Pergamon Press.

Kotta, Ü. and Tõnso, M. (1999). Transfer equivalence and realization of nonlinear higher order input/output difference equations using mathematica. In *Journal of Circuits, Systems and Computers*, volume 9, pages 23–25.

Kotta, Ü. and Tõnso, M. (2003). Linear algebraic tools for discrete-time nonlinear control systems with mathematica. In *(Lecture Notes in Control and Information Sciences; 281)*, pages 195–205, Nonlinear and Adaptive Control, NCN4 2001 / Eds. A.Zinober, D.Owens. Berlin [etc.]:. Springer.

Kotta, Ü., Zinober, A., and Liu, P. (2001). Transfer equivalence and realization of nonlinear higher order input-output difference equations. In *Automatica*, volume 37, pages 1771–1778.

Moog, C., Kotta, Ü., and Nõmm, S. (2003a). Extensions of linear algebraic methods to linear systems: an educational perspective. In *Proc. of the 6th IFAC Symposium on Advances in Control Education*, pages 179–184, Finland, Oulu.

Moog, C., Zheng, Y.-F., and Liu, P. (2003b). Input-output equivalence of nonlinear systems and their realizations. In *Proc. of the 15th IFAC World Congress: International Federation of Automatic Control*, pages 265–270, Barcelona, Spain, 21-26 July 2002. Plenary, Survey and Milestone Volume / Eds. E. F. Camacho [et al.]. [Oxford]: Pergamon.

Ondera, M. and Huba, M. (2005). Web-based tools for exact linearization control design. In *Proc. of the 16th IFAC World Congress*, Prague, Czech Republic, 4-8 July 2005.

Pearson, A. (1989). Identifiability and well-posedness in nonlinear systems modeling. Tampa, Florida, U.S.A.

Xia, X. and Moog, C. (2003). Identifiability of nonlinear systems with application to hiv/aids models. In *IEEE Transactions on Automatic Control*, volume 48(2), pages 330–335. Pergamon Press.

THE STRATEGIC GAMES MATRIX AS A FRAMEWORK FOR INTELLIGENT AUTONOMOUS AGENTS HIERARCHICAL CONTROL STRATEGIES MODELING

Eliezer Arantes da Costa and Celso Pascoli Bottura
LCSI – FEEC - UNICAMP – Cidade Universitária Zeferino Vaz, Campinas, SP, Brazil
elicosta@uol.com.br, bottura@dmcsi.fee.unicamp.br

Keywords: Autonomous agents, competitive games, cooperative games, distributed intelligent control, hierarchical architectures, hierarchical control, multiple agent control, Strategic Games Matrix, strategies modeling.

Abstracts: This paper presents a framework for strategy formulation in multilevel multiple-agent control system architectures based on the Strategic Games Matrix (SGM), having game theory and control systems theory as basic concepts and models. New methodologies for analysis and for design of hierarchical control architectures with multiple intelligent autonomous agents, based on the SGM concept, are applied. Illustrative hierarchical control applications to system architectures analysis and synthesis based on the SGM are presented.

1 INTRODUCTION

The study of hierarchical multi-agent control systems is receiving growing attention within the control community. Driving applications of multiple agents control include: mobile robots coordination and control, satellite clusters, automated highways, unmanned aerial vehicles (UAV), distributed artificial intelligence, and strategic planning in general.

A wide diversity of multi-controller and coordination problems has been treated recently, e.g., multiple mobile agents moving coordination and control (Shi, Wang and Chu, 2005), traffic congestion control (Alpcan and Başar, 2002), multiple mobile robot control (Shao, Xie, Yu and Wang, 2005), collision avoidance scheme in navigation control (Dimaragonas and Kyriakopoulus, 2005), secure routing in communication networks (Bohacek, Hespanha and Obraczka, 2002), optimal bidding strategies in the electricity market (Rahimi-Kian, Tabarraei and Sadeghi, 2005), automa-teams coordination and control (Liu, Galati and Simaan, 2004), attack and deception strategies in military operations (Castañón, Pachter and Chandler, 2004), and intrusion detection in access control systems (Alpcan and Başar, 2004).

Mathematical approaches used in these papers treat the control problems as Nash, Pareto, Stackelberg, Minimax games, or some variations of them, in an insulated manner.

The formulation of optimal strategies in competitive and/or cooperative environments has constituted one of the main challenges for researchers and scholars (Schelling, 1960; Brandenburger and Nalebuff, 1995; and Bottura and Costa, 2004) and a wide variety of approaches has been proposed and used (Başar and Older, 1999; Costa Fº., 1992; and Cruz Jr., 1978). However, a structured combination of all these possible approaches on the *same* hierarchical architecture should be conceived, formulated, and should have its usefulness exhibited. Here, an integrated framework considering these classical games on the same analytical structure, by going a step further on the traditional approach used in papers like the above mentioned, is presented.

In this paper, an 'agent' represents a controller, a decision-maker, a commander, an autonomous robot, a player – person or team –, software, a policy-maker, a UAV, a stakeholder, or any human being. Our approach treats hierarchical, non-hierarchical, or heterarchical architectures as a structured collections of sub-games.

2 STRATEGIC GAMES MATRIX

The concepts, formulations and results from *non-cooperative dynamic game theory* (Başar and Olsder, 1999) open new possibilities as conceptual platform for optimal strategy formulation.

In generic conflict of interests' situations, the description and mapping of a particular cooperative or competitive confrontation between two or more *players* can be accomplished with only two dimensions: the '*player posture assumption'* and the 'player *power-ratio assumption*'. They are used to build a (3x3) matrix called *strategic games matrix* (SGM) (Costa and Bottura, 2006): The matrix horizontal axis represents the player postures assumptions: as *rival*, or *individualistic,* or *associative* and, on the vertical axis represents the *player power-ratio assumptions*: as *hegemonic*, or *balanced,* or *weak*, as shown in Figure 1.

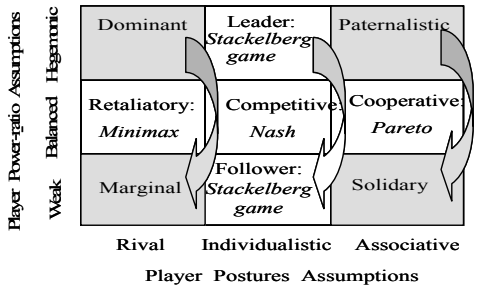


Figure 1: Typical strategic positions on the SGM highlighting, in gray, the two hierarchical limit-case strategic games.

These nine resulting strategic positions, at each of the nine matrix's cells, are named, respectively: *Dominant*, *Leader*, *Paternalistic*, *Retaliatory*, *Competitive*, *Cooperative*, *Marginal*, *Follower*, and *Solidary*, which are words that represent each one of the typical competitive confrontation strategic positions players may explicitly or implicitly adopt in a conflict of interests situation. In subsections 2.1 to 2.4 the five strategic positioning to which classic equilibrium strategies apply - *Minimax, Nash, Pareto*, for non-hierarchical games, and *Stackelberg,* for hierarchical games - and the respective situations where they normally occur, are described (Başar and Olsder, 1999; Costa F^{o}., 1992).

In subsections 2.5 and 2.6, the four special limit-cases strategic positions, representing two hierarchical games, not well covered by classic equilibrium strategies from game theory, here called *Dominant-Marginal*, and *Paternalistic-Solidary*, are presented in the next Sections. (The formal concept of dynamic games, of *equilibrium point* and of *equilibrium strategy* here used can be found in (Başar and Older, 1999)).

2.1 Retaliatory Games - Minimax

This strategic positioning applies to *lose-win* type games - at the left-center SGM cell -, where the players assume, explicit or implicitly, that a gain for one implies in losses to the remainder, characterizing a *retaliatory game*. For a zero-sum game, a solution, if it exists, for which each player acts towards what it understands as the most favorable to optimize its own objective function, considering all the possibilities the others could do, is called a *saddle-point*. This point has the peculiar characteristic that any deviation from it, by any of the players, makes its result worsen in relation to its objective function. For N players, a strategic decision $\hat{u}^i \in U^i$ by each player P_i is defined as a *saddle-point equilibrium solution* if, for every admissible set $\{u^1,\cdots,u^i,\cdots,u^N\} \in U$, the following relation is valid:

$$\max_{u^1,\cdots,u^{i-1},u^{i+1},\cdots,u^N} J_i(u^1,\cdots,u^{i-1},\hat{u}^i,u^{i+1},\cdots,u^N) \leq \leq \max_{u^1,\cdots,u^{i-1},u^{i+1},\cdots,u^N} J_i(u^1,\cdots,u^N)$$

This strategy applies also to real situations where a player $\boldsymbol{P}_i$ can imagine that another player may have non-rational or erratic behavior, or even malicious, i.e., that an adversary may make moves to 'damage' $\boldsymbol{P}_i$'s objectives.

2.2 Competitive Games - Nash

The strategic position at the center-center SGM cell, named here as C*ompetitive*, describes situations of 'perfect competition', or 'free market', with many suppliers, where none of them is capable of dominating the remainders. In the non-cooperative variable-sum games, where a player decides to play a competitive strategic game, it seeks to optimize its objective function ignoring what the other players are doing or intending to do. If this solution exists, it is characterized by the situation where none of the players is able to improve its result by changing only its own decision-control. Such set of decisions is the *Nash equilibrium point*, defined below: A *Nash equilibrium point*

$$\hat{u}^* = (\hat{u}^1,\ldots,\hat{u}^i,\ldots,\hat{u}^N) \in U \ ,$$

if it exists, for a non-cooperative game, with K=1, and variable sum, with N players, is defined if, for all $u^i \in U^i$, $i \in N$, it obeys simultaneously the N following objective function inequalities:

$J_1(\hat{u}^1,\ldots,\hat{u}^i,\ldots,\hat{u}^N) \le J_1(u^1,\ldots,\hat{u}^i,\ldots,\hat{u}^N)$, ... ,

$J_i(\hat{u}^1,\ldots,\hat{u}^i,\ldots,\hat{u}^N) \le J_i(\hat{u}^1,\ldots,u^i,\ldots,\hat{u}^N)$, ... ,

$J_N(\hat{u}^1,\ldots,\hat{u}^i,\ldots,\hat{u}^N) \le J_N(\hat{u}^1,\ldots,\hat{u}^i,\ldots,u^N)$.

2.3 Cooperative Games – Pareto

For variable-sum games - at the right-center SGM cell - the cooperation among players may lead to results - for all of them - that are better than those they would obtain if each one tries to optimize its objective function without an *a priori* knowledge of other's decisions. When players decide to share information on the respective constraints and conditions, alternative actions and objective functions, it is possible for them to find a point of equilibrium, the '*Pareto optimum*', which is 'the best' possible for all players. This point, if it exists, is characterized by the fact that none of the players can improve its result without, with its action, harming the other's results. These are the so called 'win-win games'. This type of game requires good faith and loyalty among all participants. For a variable-sum cooperative game (K=1) with **N** players, the point $\hat{u}^* = (\hat{u}^1,\ldots,\hat{u}^i,\ldots,\hat{u}^N) \in U$ is defined as a *Pareto optimum* if there is no other point

$u = (u^1,\ldots,u^i,\ldots,u^N) \in U$ such that

$J_i(u^i) \le J_i(\hat{u}^i)$, $\forall i \in N$.

This condition requires that $J_i(u^i) \le J_i(\hat{u}^i)$, $\forall i \in N$, only if $J_i(u^i) = J_i(\hat{u}^i)$, $\forall i \in N$, with a strict inequality for at least one $i \in N$.

2.4 Leader-Follower Stackelberg Games

The strategies applicable to hierarchical games with a strongest player, the *leader*, and another weaker player, the *follower*, are called *Stackelberg strategies* and correspond to two opposed positions: center-upper and center-lower SGM cells. Consider a simplified *hierarchical game* between a player *M*, called *leader*, and a player *P*, called *follower,* with strategic decisions λ and u , and objective functions $R(\lambda,u)$ and $J(\lambda,u)$, associated to players *M* and *P*, respectively (Haimes and Li, 1988; Costa F°. and Bottura, 1990, 1991). Let us suppose also that, by the structure and rules of the game, player *M* selects first its strategic decision λ and, then, player *P* selects its strategic decision u , *knowing beforehand* the *M*'s decision. The pair $(\hat{\lambda},\hat{u}) \in (L,U)$, if it exists, defines a *Stackelberg equilibrium point* for which:

(a) There is a transformation $T : L \to U$ such that, for any given $\lambda \in L$, $J(\lambda, T\lambda) \le J(\lambda, u)$ for every $u \in U$, and (b) There is a $\hat{\lambda} \in L$ such that $R(\hat{\lambda}, T\hat{\lambda}) \le R(\lambda, T\lambda)$ for every $\lambda \in L$, where $\hat{u} = T\hat{\lambda}$. Note that, to obtain a *Stackelberg equilibrium point*, it is necessary that the *follower* be a rational agent, always making optimal decisions under its own game limitation. For this game structure, one can determine a pair of *Stackelberg* strategies - for the *leader* and for the *follower* - typically applied to situations of conflict of interests between a very strong player and another very weak, both with *individualistic* concurrent assumptions.

2.5 Dominant-Marginal Games

The Dominant-Marginal games are played by two players in two hierarchical antagonist strategic positions, both with rival posture assumption:

(1) Dominant strategic position: A *Dominant strategic position* - at the left-upper SGM cell - characterizes the player which has all strength and has the intention of destroying the smaller competitors. Its attitude may be of intimidation, blackmail, price war, for instance, to try to bankrupt the small ones. It may pressure its clients not to purchase from the small ones. A *Dominant equilibrium point* limit-case for this game can be obtained through the solution of a mono-criterion stochastic optimization problem in which the player in *Dominant position* ignores all the objective functions of its 'small' opponents and simply optimizes its own objective function. The player at a *Dominant* position could treat the possible actions of 'small' competitors simply as random noises.

(2) Marginal strategic position: Countering the *Dominant position* as described above, is the *marginal strategic position* - at the left-lower SGM cell -, where a weaker however courageous and competitive player in the game does everything it understands as necessary to survive, trying, as much as possible, to obtain some advantages upon causing losses to the major game dominator. A *marginal equilibrium point* limit-case for this game can be obtained through the solution of an optimization problem in which the *Marginal* position player, for instance, instead of minimizing, tries to *maximize* the main and stronger competitor's objective function

with the purpose of infringing upon it the maximum possible damage.

2.6 Paternalistic-Solidary Games

This game is played also by two players in two hierarchical antagonist strategic positions, both with associative posture assumption:

(1) Paternalistic strategic position: The *paternalistic strategic position* **-** at the upper-right SGM cell - occurs in games where a more powerful player, by its own decision, shapes its own actions and those of the remaining weaker players in the game, seeking preservation and development of the system as a whole. It is a game similar to the situation of a family father, supposed to have complete authority over the small children: he does all he comprehends to be necessary to promote the development, growth and harmony within his family, in a paternalistic way. A *paternalistic equilibrium point* limit-case game can be found as follows: Let $0 \leq \alpha_i \leq 1$ be a relative importance weight for the player $\mathbf{P_i}$ such that $\sum_{i=1}^{N} \alpha_i = 1$, and let $z = \sum_{i=1}^{N} \alpha_i J_i^{(\ldots)}$ be a multi-criteria objective function, encompassing all the objective functions of all the N players, the new function to be optimized. A *paternalistic equilibrium point* for this limit-case game can be found as a solution to a multi-criteria optimization problem (Bryson and Ho, 1975) where the new objective function is a linear combination of all the objective functions for all players. Otherwise, the *Paternalistic* player should take in account, on its decision, the 'risk' of a *Solidary* player decision for an alternative *solitary strategy*, leaving the game.

(2) Solidary strategic position: In opposition to the *paternalistic position* described above is the *Solidary position* - at the right-lower SGM cell -, that represents the situation of a player, in a game, in a weaker, however associative position which, without the power to impose its interests upon the others, seeks to follow the rules established by the 'ruling power', looking for some individual advantage. Otherwise it prefers to leave the game. This is how a member behaves in relation to its cooperative organization: it simply needs to decide whether it should join the 'collective' and obtain some advantage or, alternatively, it should rather act on its own. A *solidary equilibrium solution* for this limit-case game can be treated as a simple *decision tree problem* with only two branches, representing the alternative decisions: '*join the collective*', or '*work alone*'.

3 HIERARCHICAL GAMES

Departing from classic concepts and formulations from *dynamic game theory*, a formal conceptual platform for multilevel multiple decision-control problem formulation is built. A *deterministic dynamic game* (DDG) with several participants and multiple stages can be modeled as a systems optimization problem with multiple decentralized and autonomous decision-makers, called the '*players' –or intelligent autonomous agents.* From the point of view of *systems control theory*, a DDG is associated with a particular problem of *optimal control* with *multiple intelligent autonomous controllers, or agents* (Bryson and Ho, 1975).

In this type of games, each one of the N agents - or *players* - receiving information progressively disclosed by the structure of the game and considering the possible decisions of other agents, makes a sequence of decisions, stage by stage, attempting to optimize one's *objective function* - while obeying the game constraints. For a formal presentation of the optimization problem introduced above, let us adopt the notation derived from the terminology of *systems theory* (Başar and Olsder, 1999). Hierarchical architectures games with two levels, designed by HG2, and with three levels, designed by HG3, for multiple intelligent autonomous agents control strategies, are here described. A two-level hierarchical game, HG2, can be modeled through a similar process of forming a group of subsystems, each one representing a competing agent – for instance, a company. Each company - the i^{th} - here represented by a subsystem CS_i, vies in the market for raw materials, specialized production manpower, managerial resources, financial resources, technology, and other supplies. On the other hand, it also competes in the market for clients' preferences. The market, in the broader sense, also interferes in the game, acting upon prices and quantities transacted by the N agents with their clients and providers. The formulation of this concept can be obtained through a convenient partition and segmentation process of the DDG game: The HG2 is formed by two types of subsystems: the ***Companies Subsystems, CS_i,*** and the ***Market Coordinator Subsystems, MCS***. The CS_i modules communicate with the market coordinator subsystem, MCS, which informs to each one of them, at the beginning of each new period, its decision parameter. The CS_i, in turn, informs the MCS about their coordinated decisions for the next period. The dynamic hierarchical game HG-2 can be similarly expanded applying to each subsystem CS_i a segmentation process, where each i^{th} competing

agent is assumed to consist of *G Managerial Units*, MU_{ij}, where $j \in \{1,2,...,G\}$, introducing *G* new intelligent autonomous agents for each company. These managerial units, MU_{ij}, represent the main functional or managerial areas of the company. In this sense, each MU_{ij}, as any intelligent autonomous agent, has its own state transition equation, information structure, strategy, decision, and specific objective function to be optimized. Therefore, the segmentation described produces a *three-level hierarchical game* HG-3 wherein the coordination, at the second level, is achieved by a new module called CSC_i, representing the coordination of all the MU_{ij}, by the i^{th} company's chief executive.

4 SGM APPLICATIONS

Let us apply, now, with illustrative purposes, the SGM methodology for a complex structure analysis to some HG-3 structured games.

4.1 Structure with One Coordinator

Suppose a complex business-economic structured system, with three decision hierarchical levels. Proceeding accord to this methodology the following results can be obtained:

(A) The four sub-games identified are: $\{CS_1,\ldots,CS_i,\ldots,CS_N\}$ competing - or cooperating - sub-game; $\{MU_{i1},\ldots, MU_{ij}, MU_{iG}\}$ competing - or cooperating - subgame; $\{MCS, CS_i\}$ hierarchical coordination sub-game; $\{CSC_i, MU_{ij}\}$ hierarchical coordination sub-game.

(B) The application of one or another equilibrium strategy on each specific sub-game depends on each particular situation of conflict of interests and on the postures and assumptions present in each case:

(i) The competitive sub-game among CS_i companies could be treated as a game where the agents are supposed to work in a *variable-sum* objective function environment, acting independently from each other and prevented from sharing information and from cooperating with each other. They are forbidden to make coordinated decisions to optimize together their objective functions; consequently, for this sub-game, the *Nash equilibrium strategy* is the applicable, as in subsection 2.2.

(ii) Among those responsible for the MU_{ij} Managerial Units on the same company, a sub-game is played where the agents aim to optimize a *variable-sum objective function* for which cooperation among the unit managers in charge is expected; hence, for this sub-game, the *Pareto equilibrium strategy* is the applicable, as in subsection 2.3.

(iii) The relationship between the agent MCS, the market coordinator, representing the market action, and each CS_i company could be interpreted as a sub-game with hierarchical coordination among them; therefore, the *Stackelberg equilibrium strategies* pair is applicable, considering the market coordinator as the *Leader* and each CS_i as a *Follower*, as in subsection 2.4;

(iv) The relationship between the agent CSC_i, internal coordinator of each company, and each MU_{ij} could be considered as a hierarchical coordination sub-game; so, the Stackelberg equilibrium strategy pair is applicable, considering the coordinator CSCi as the Leader and each MUij as a Follower, as in subsection 2.4.

(C) The structured mapping resulting from the fourth stage, easy to obtain in this case, is also indicated in Figure 2. Classic ways of solving these types of *optimal control problems* could use, for instance, *Pontryagin's Minimum Principle*, or *Calculus of Variations,* or *Dynamic Programming* (Bryson and Ho, 1975), depending on the case.

4.2 Structure with Two Coordinators

This subsection presents, in a summarized form, another illustrative application of this methodology for analysis of another type of hierarchic structure. Let us take the former HG-3 as a basis and introduce a second coordinator agent at the first level, as shown in Figure 2.

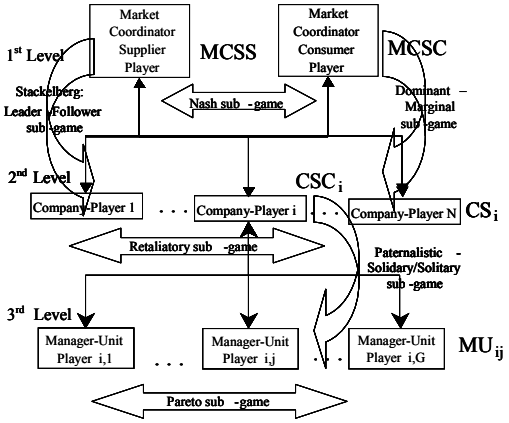


Figure 2: Game equilibrium strategies applied to a three-level multiple decision control architecture with two coordinators.

This structure has now two market coordinators, one representing the *market coordinator –supplier–*,

MCSS, and another *market coordinator –consumer– ,* MCSC**.** The resulting structural mapping obtained from a similar use of the four stages methodology, and the corresponding equilibrium strategies applicable to each sub-game identified, are shown in Figure 2.

5 FINAL CONCLUSIONS

In this paper the *strategic games matrix* (SGM) modeling framework is used as a tool for:

- Describing, characterizing, and mapping a wide variety of conflicts of interests situations among intelligent autonomous agents, both for hierarchical and for non-hierarchical games, in an integrated manner;
- Modeling, analysis and design of multilevel multiple-agent control architectures in an integrated manner, making explicit the obvious conflicts of interests possibilities;
- Establishing a useful two-way conceptual bridge between game theory and multiple-agent structures analysis and design.

The SGM permits to evidence that, for a specific real complex problem, we should be more concerned with the choice of the *right game to model*, than with the *right way to solve the game*, in spite of the importance of these techniques.

REFERENCES

Alpcan, T., Başar, T., 2002. "A game-theoretic framework for congestion control in general topology networks", *Proc. 41st IEEE CDC*, Las Vegas, Nevada.

Alpcan, T., Başar, T., 2004. "A game theoretic analysis of intrusion detection in access control systems", *Proc. 43rd IEEE CDC*, Atlantis, Paradise Islands, Bahamas.

Başar, T., Olsder, G. J., 1999. *Dynamic non-cooperative game theory*. Philadelphia, PA: SIAM, Series in Classics in Applied Mathematics, 1999.

Bohacek, S., Hespanha, J. P., Obraczka, K., 2002. "Saddle policies for secure routing in communication networks", *Proc. 41st IEEE CDC*, Las Vegas, Nevada.

Bottura C. P., Costa, E. A., 2004, "Business strategy formulation modeling via hierarchical dynamic game", *Proc. CSIMTA International Conference*, Cherbourgh, France.

Brandenburger, A. M., Nalebuff, B. J., 1995. "The right game: Use of game theory to shape strategy", *Harvard Business Review,* pp.57-81.

Bryson Jr., A. E, Ho, Y. C., 1975. *Applied optimal control*. Washington, DC: Hemisphere.

Castañón, D. A., Pachter, M., Chandler, P. R., 2004. "A game of deception", *Proc. 43rd IEEE, CDC*, Atlantis, Paradise Islands, Bahamas.

Costa Fº, J. T., Bottura, C. P., 1990. "Parallel optimal hierarchical control using a MIMD architecture", *Proc. 29th IEEE CDC*, Honolulu.

Costa Fº, J. T., Bottura, C. P., 1991. "Hierarchical multidecision making on a computer network with distributed coordination and control", *Proc. 39th Annual Allerton Conference on Communication Control and Computing*, Urbana, IL, pp. 703-704.

Costa Fº., J. T., 1992. "Proposta para computação assíncrona paralela e distribuída de estruturas especiais de jogos dinâmicos", *Universidade Estadual de Campinas*, Faculdade de Engenharia Elétrica, Tese de Doutorado, Campinas, SP, Brazil.

Costa, E. A., Bottura, C. P., 2006. "The Strategic Games Matrix (SGM) as a new tool for strategic management via game theory", *Sistemas & Gestão,* 1 (1) pp. 17-41. (in http://www.latec.com.br/sg/arevista/Volume1/Numero1/V1_1_index.htm).

Cruz Jr., J. B., 1978. "Leader-follower strategies for multilevel systems", *IEEE Trans. on Automatic Control*, vol. AC-23 (2), pp. 244-255.

Dimarogonas, D. V., Kyriakopoulus, K. J., 2005. "A feedback stabilization and collision avoidance scheme for multiple independent nonholonomic non-point agents", *Proc. 2005 IEEE Int. Symposium on Intelligent Control*, Limassol, Cyprus.

Haimes, Y. Y., Li, D., 1988. "Hierarchical multiobjective analysis for large-scale systems: Review and current status", *Automatica,* vol. 24 (1), pp. 53-69.

Liu, Y., Galati, D. G., Simaan, M. A., 2004, "A game theoretic approach to team dynamics and tactics in mixed initiative control of automa-teams", *Proc. 43rd IEEE CDC*, Atlantis, Paradise Islands, Bahamas.

Rahimi-Kian, A. Tabarraei, H., Sadeghi, B., 2005. "Reinforcement learning based supplier-agents for electricity markets", *Proc. 2005 IEEE Int. Symposium on Intelligent Control*, Limassol, Cyprus.

Schelling, T. C., 1960. *The strategy of conflict*. New York, NY: Harvard University Press.

Shao, J., Xie, G., Yu, J., Wang, L., 2005. "Leader-following formation control of multiple mobile robots", *Proc. 2005 IEEE Int. Symposium on Intelligent Control*, Limassol, Cyprus.

Shi, H., Wang, L., T. Chu, T., 2005. "Coordination of multiple dynamic agents with asymmetric interactions", *Proc. 2005 IEEE Int. Symposium on Intelligent Control*, Limassol, Cyprus.

MINIMIZATION OF l_2-SENSITIVITY FOR 2-D SEPARABLE-DENOMINATOR STATE-SPACE DIGITAL FILTERS SUBJECT TO l_2-SCALING CONSTRAINTS USING A LAGRANGE FUNCTION AND A BISECTION METHOD

Takao Hinamoto, Yukihiro Shibata and Masayoshi Nakamoto
Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima 739-8527, Japan
{*hinamoto,nights400,msy*}*@hiroshima-u.ac.jp*

Keywords: Two-dimensional (2-D) state-space digital filters, separable denominator, l_2-sensitivity, l_2-scaling constraints, scaling-constrained sensitivity minimization, optimal realization.

Abstract: The problem of minimizing l_2-sensitivity subject to l_2-scaling constraints for two-dimensional (2-D) separable-denominator state-space digital filters is investigated. The coefficient sensitivity of the filter is analized by using a pure l_2-norm. An iterative algorithm for minimizing an l_2-sensitivity measure subject to l_2-scaling constraints is then explored by introducing a Lagrange function and utilizing an efficient bisection method. A numerical example is also presented to illustrate the utility of the proposed technique.

1 INTRODUCTION

In the fixed-point finite-word-length (FWL) implementation of recursive digital filters, the characteristics of an actual transfer function deviate from the original ones due to either truncation or rounding of filter coefficients. So far, several techniques for synthesizing two-dimensional (2-D) filter structures with low coefficient sensitivity have been reported (Kawamata et al., 1987)-(Hinamoto and Sugie, 2002). Some of them use a sensitivity measure evaluated by a mixture of l_1/l_2-norms (Kawamata et al., 1987; Hinamoto et al., 1992; Hinamoto and Takao, 1992), while the others rely on the use of a pure l_2-norm (Li, 1998; Hinamoto et al., 2002; Hinamoto and Sugie, 2002). Moreover, minimization of frequency-weighted sensitivity for 2-D state-space digital filters has been considered in accordance with both a mixed l_1/l_2-sensitivity measure and a pure l_2-sensitivity measure (Hinamoto et al., 1999). The l_2-sensitivity minimization is more natural and reasonable than the conventional l_1/l_2-mixed sensitivity minimization, but it is technically more challenging. Alternatively, a state-space digital filter with l_2-scaling constraints is beneficial for suppressing overflow oscillations (Mullis and Roberts, 1976; Hwang, 1977). However, satisfactory solution methods for l_2-sensitivity minimization subject to l_2-scaling constraints are still needed (Hinamoto et al., 2004; Hinamoto et al., 2005).

In this paper, an l_2-sensitivity minimization problem subject to l_2-scaling constraints for 2-D separable-denominator digital filters is formulated. An efficient iterative algorithm is explored to solve the constrained optimization problem directly. This is performed by applying a Lagrange function and an efficient bisection method. Computer simulation results by a numerical example demonstrate the validity and effectiveness of the proposed technique.

2 SENSITIVITY ANALYSIS

There is no loss of generality in assuming that a 2-D digital filter which is separable in the denominator can be described by the Roesser local state-space (LSS) model $\{A_1, A_2, A_4, b_1, b_2, c_1, c_2, d\}_{m+n}$ (Roesser, 1975; Hinamoto, 1980) as

$$\begin{bmatrix} x^h(i+1,j) \\ x^v(i,j+1) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \\ \mathbf{0} & A_4 \end{bmatrix} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u(i,j)$$

$$y(i,j) = \begin{bmatrix} c_1 & c_2 \end{bmatrix} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + d\,u(i,j) \tag{1}$$

where $x^h(i,j)$ is an $m \times 1$ horizontal state vector, $x^v(i,j)$ is an $n \times 1$ vertical state vector, $u(i,j)$ is a scalar input, $y(i,j)$ is a scalar output, and A_1, A_2, A_4,

b_1, b_2, c_1, c_2, and d are real constant matrices of appropriate dimensions. The LSS model in (1) is assumed to be asymptotically stable, separately locally controllable and separately locally observable (Kung et al., 1977). The transfer function of the LSS model in (1) is given by

$$\begin{aligned} &H(z_1,z_2)\\ &= \begin{bmatrix} c_1 & c_2 \end{bmatrix} \begin{bmatrix} z_1 I_m - A_1 & -A_2 \\ \mathbf{0} & z_2 I_n - A_4 \end{bmatrix}^{-1} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} + d \\ &= \begin{bmatrix} 1 & c_1(z_1 I_m - A_1)^{-1} \end{bmatrix} \\ &\qquad \cdot \begin{bmatrix} d & c_2 \\ b_1 & A_2 \end{bmatrix} \begin{bmatrix} 1 \\ (z_2 I_n - A_4)^{-1} b_2 \end{bmatrix}. \end{aligned} \tag{2}$$

Definition 1: Let X be an $m \times n$ real matrix and let $f(X)$ be a scalar complex function of X, differentiable with respect to all the entries of X. The sensitivity function of f with respect to X is then defined as

$$S_X = \frac{\partial f}{\partial X} \text{ with } (S_X)_{ij} = \frac{\partial f}{\partial x_{ij}} \tag{3}$$

where x_{ij} denotes the (i,j)th entry of the matrix X.

With these notations, it is easy to show that

$$\begin{aligned} \frac{\partial H(z_1,z_2)}{\partial A_1} &= Q^T(z_1) F^T(z_1,z_2) \\ \frac{\partial H(z_1,z_2)}{\partial A_2} &= Q^T(z_1) P^T(z_2) \\ \frac{\partial H(z_1,z_2)}{\partial A_4} &= G^T(z_1,z_2) P^T(z_2) \\ \frac{\partial H(z_1,z_2)}{\partial b_1} &= Q^T(z_1) \\ \frac{\partial H(z_1,z_2)}{\partial b_2} &= G^T(z_1,z_2) \\ \frac{\partial H(z_1,z_2)}{\partial c_1^T} &= F(z_1,z_2) \\ \frac{\partial H(z_1,z_2)}{\partial c_2^T} &= P(z_2) \end{aligned} \tag{4}$$

where

$$\begin{aligned} F(z_1,z_2) &= (z_1 I_m - A_1)^{-1}[b_1 + A_2 P(z_2)] \\ G(z_1,z_2) &= [c_2 + Q(z_1) A_2](z_2 I_n - A_4)^{-1} \\ P(z_2) &= (z_2 I_n - A_4)^{-1} b_2, \quad Q(z_1) = c_1(z_1 I_m - A_1)^{-1}. \end{aligned}$$

The term d and the sensitivity with respect to it are coordinate independent, therefore they are neglected here.

Definition 2: Let $X(z_1,z_2)$ be an $m \times n$ complex matrix valued function of the complex variables z_1 and z_2. The l_p-norm of $X(z_1,z_2)$ is then defined as

$$||X||_p = \left[\frac{1}{(2\pi j)^2} \oint \oint_{\Gamma^2} ||X(z_1,z_2)||_F^p \frac{dz_1 dz_2}{z_1 z_2} \right]^{1/p} \tag{5}$$

where $||X(z_1,z_2)||_F$ is the Frobenius norm of the matrix $X(z_1,z_2)$ defined by

$$||X(z_1,z_2)||_F = \left[\sum_{p=1}^{m} \sum_{q=1}^{n} \left| x_{pq}(z_1,z_2) \right|^2 \right]^{1/2}.$$

The overall l_2-sensitivity measure is now defined by

$$\begin{aligned} M_2 = &\left|\left| \frac{\partial H(z_1,z_2)}{\partial A_1} \right|\right|_2^2 + \left|\left| \frac{\partial H(z_1,z_2)}{\partial A_4} \right|\right|_2^2 \\ &+ \left|\left| \frac{\partial H(z_1,z_2)}{\partial b_1} \right|\right|_2^2 + \left|\left| \frac{\partial H(z_1,z_2)}{\partial b_2} \right|\right|_2^2 \\ &+ \left|\left| \frac{\partial H(z_1,z_2)}{\partial c_1^T} \right|\right|_2^2 + \left|\left| \frac{\partial H(z_1,z_2)}{\partial c_2^T} \right|\right|_2^2 \\ &+ \left|\left| \frac{\partial H(z_1,z_2)}{\partial A_2} \right|\right|_2^2. \end{aligned} \tag{6}$$

From (4)-(6), it follows that

$$\begin{aligned} M_2 = &\operatorname{tr}\left[M_{A_1} + M_{A_4} + W^h + W^v + K^h + K^v \right] \\ &+ \operatorname{tr}[W^h] \operatorname{tr}[K^v] \end{aligned} \tag{7}$$

where

$$\begin{aligned} M_{A_1} &= \frac{1}{(2\pi j)^2} \oint_{|z_1|=1} \oint_{|z_2|=1} [F(z_1^{-1}, z_2^{-1}) Q(z_1^{-1})] \\ &\qquad \cdot [Q^T(z_1) F^T(z_1,z_2)] \frac{dz_1 dz_2}{z_1 z_2} \\ M_{A_4} &= \frac{1}{(2\pi j)^2} \oint_{|z_1|=1} \oint_{|z_2|=1} [G^T(z_1,z_2) P^T(z_2)] \\ &\qquad \cdot [P(z_2^{-1}) G(z_1^{-1}, z_2^{-1})] \frac{dz_1 dz_2}{z_1 z_2} \end{aligned}$$

$$\begin{aligned} K^h &= \frac{1}{(2\pi j)^2} \oint_{|z_1|=1} \oint_{|z_2|=1} F(z_1,z_2) F^*(z_1,z_2) \frac{dz_1 dz_2}{z_1 z_2} \\ K^v &= \frac{1}{2\pi j} \oint_{|z_2|=1} P(z_2) P^*(z_2) \frac{dz_2}{z_2} \\ W^h &= \frac{1}{2\pi j} \oint_{|z_1|=1} Q^*(z_1) Q(z_1) \frac{dz_1}{z_1} \\ W^v &= \frac{1}{(2\pi j)^2} \oint_{|z_1|=1} \oint_{|z_2|=1} G^*(z_1,z_2) G(z_1,z_2) \frac{dz_1 dz_2}{z_1 z_2}. \end{aligned}$$

The matrices $K = K^h \oplus K^v$ and $W = W^h \oplus W^v$ are called the local controllability Gramian and local observability Gramian, respectively, and can be obtained

by solving the following Lyapunov equations (Kawamata and Higuchi, 1986):

$$\begin{aligned} K^v &= A_4 K^v A_4^T + b_2 b_2^T \\ K^h &= A_1 K^h A_1^T + A_2 K^v A_2^T + b_1 b_1^T \\ W^h &= A_1^T W^h A_1 + c_1^T c_1 \\ W^v &= A_4^T W^v A_4 + A_2^T W^h A_2 + c_2^T c_2. \end{aligned} \tag{8}$$

Apply the following eigenvalue-eigenvector decompositions:

$$K^v = \sum_{i=1}^{n} \sigma_i^v u_i u_i^T, \qquad W^h = \sum_{i=1}^{m} \sigma_i^h v_i v_i^T \tag{9}$$

where σ_i^v and u_i (σ_i^h and v_i) are the ith eigenvalue and eigenvector of K^v (W^h), respectively. Then, we can write (7) as (Hinamoto and Sugie, 2002)

$$\begin{aligned} M_2 = &\sum_{i=0}^{n} \sigma_i^v \mathrm{tr}[W_i^h(I_m)] + \sum_{i=0}^{m} \sigma_i^h \mathrm{tr}[K_i^v(I_n)] \\ &+ \mathrm{tr}[W^h + W^v + K^h + K^v] + \mathrm{tr}[W^h]\mathrm{tr}[K^v] \end{aligned} \tag{10}$$

where $\sigma_0^v = \sigma_0^h = 1$,

$$\tilde{u}_i = \begin{cases} b_1 & \text{for } i = 0 \\ A_2 u_i & \text{for } i \geq 1 \end{cases}$$

$$\tilde{v}_i = \begin{cases} c_2^T & \text{for } i = 0 \\ A_2^T v_i & \text{for } i \geq 1 \end{cases}$$

and an $m \times m$ matrix $W_i^h(P_1)$ and an $n \times n$ matrix $K_i^v(P_4)$ are obtained by solving the following Lyapunov equations:

$$\begin{bmatrix} W_i^h(P_1) & * \\ * & * \end{bmatrix} = \begin{bmatrix} A_1 & \tilde{u}_i c_1 \\ \mathbf{0} & A_1 \end{bmatrix} \begin{bmatrix} W_i^h(P_1) & * \\ * & * \end{bmatrix} \cdot \begin{bmatrix} A_1 & \tilde{u}_i c_1 \\ \mathbf{0} & A_1 \end{bmatrix}^T + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_1 \end{bmatrix}$$

$$\begin{bmatrix} K_i^v(P_4) & * \\ * & * \end{bmatrix} = \begin{bmatrix} A_4 & \mathbf{0} \\ b_2 \tilde{v}_i^T & A_4 \end{bmatrix}^T \begin{bmatrix} K_i^v(P_4) & * \\ * & * \end{bmatrix} \cdot \begin{bmatrix} A_4 & \mathbf{0} \\ b_2 \tilde{v}_i^T & A_4 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_4^{-1} \end{bmatrix}.$$

3 SENSITIVITY MINIMIZATION

3.1 Problem Formulation

The following class of state-space coordinate transformations can be used without affecting the input-output map:

$$\begin{bmatrix} \bar{x}^h(i,j) \\ \bar{x}^v(i,j) \end{bmatrix} = \begin{bmatrix} T_1 & \mathbf{0} \\ \mathbf{0} & T_4 \end{bmatrix}^{-1} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} \tag{11}$$

where T_1 and T_4 are $m \times m$ and $n \times n$ nonsingular constant matrices, respectively. Performing this coordinate transformation to the LSS model in (1) yields a new realization $\{\bar{A}_1, \bar{A}_2, \bar{A}_4, \bar{b}_1, \bar{b}_2, \bar{c}_1, \bar{c}_2, d\}_{m+n}$ characterized by

$$\begin{aligned} &\bar{A}_1 = {T_1}^{-1} A_1 T_1, \qquad \bar{A}_2 = {T_1}^{-1} A_2 T_4 \\ &\bar{A}_4 = {T_4}^{-1} A_4 T_4, \qquad \bar{b}_1 = {T_1}^{-1} b_1 \\ &\bar{b}_2 = {T_4}^{-1} b_2, \qquad \bar{c}_1 = c_1 T_1, \qquad \bar{c}_2 = c_2 T_4 \\ &\overline{K}^h = T_1^{-1} K^h {T_1}^{-T}, \quad \overline{K}^v = {T_4}^{-1} K^v {T_4}^{-T} \\ &\overline{W}^h = T_1^T W^h T_1, \qquad \overline{W}^v = T_4^T W^v T_4. \end{aligned} \tag{12}$$

For the new realization, the l_2-sensitivity measure M_2 in (10) is changed to

$$\begin{aligned} M_2(P) = &\sum_{i=0}^{n} \sigma_i^v \mathrm{tr}[W_i^h(P_1) P_1^{-1}] + \sum_{i=0}^{m} \sigma_i^h \mathrm{tr}[K_i^v(P_4) P_4] \\ &+ \mathrm{tr}[W^h P_1 + W^v P_4 + K^h P_1^{-1} + K^v P_4^{-1}] \\ &+ \mathrm{tr}[W^h P_1]\mathrm{tr}[K^v P_4^{-1}] \end{aligned} \tag{13}$$

where $P = P_1 \oplus P_4$ and $P_i = T_i T_i^T$ for $i = 1, 4$.

If l_2-norm dynamic-range scaling constraints are imposed on the new local state vector $[\bar{x}^h(i,j)^T, \bar{x}^v(i,j)^T]^T$, then

$$\begin{aligned} (\overline{K}^h)_{ii} &= ({T_1}^{-1} K^h {T_1}^{-T})_{ii} = 1 \\ (\overline{K}^v)_{jj} &= ({T_4}^{-1} K^v {T_4}^{-T})_{jj} = 1 \end{aligned} \tag{14}$$

are required for $i = 1, 2, \cdots, m$ and $j = 1, 2, \cdots, n$.

From the above arguments, the problem is now formulated as follows: For given A_1, A_2, A_4, b_1, b_2, c_1 and c_2, obtain an $(m+n) \times (m+n)$ nonsingular matrix $T = T_1 \oplus T_4$ which minimizes (13) subject to l_2-scaling constraints in (14).

3.2 Problem Solution

If we sum up m constraints and n constraints in (14) separately, then we have

$$\mathrm{tr}[K^h P_1^{-1}] = m, \qquad \mathrm{tr}[K^v P_4^{-1}] = n. \tag{15}$$

Consequently, the problem of minimizing $M_2(P)$ in (13) subject to the constraints in (14) can be *relaxed* into the problem

$$\begin{aligned} &\text{minimize } M_2(P) \text{ in (13)} \\ &\text{subject to } \mathrm{tr}[K^h P_1^{-1}] = m \text{ and } \mathrm{tr}[K^v P_4^{-1}] = n. \end{aligned} \tag{16}$$

In order to solve (16), we define a Lagrange function of the problem as

$$\begin{aligned} J(P, \lambda_1, \lambda_4) = &M_2(P) + \lambda_1(\mathrm{tr}[K^h P_1^{-1}] - m) \\ &+ \lambda_4(\mathrm{tr}[K^v P_4^{-1}] - n) \end{aligned} \tag{17}$$

where λ_1 and λ_4 are Lagrange multipliers. It is well known that the solution of problem (16) must satisfy the Karush-Kuhn-Tucker (KKT) conditions $\partial J(P,\lambda_1,\lambda_4)/\partial P_i = \mathbf{0}$ for $i = 1,4$ where the gradients are found to be

$$\begin{aligned}\frac{\partial J(P,\lambda_1,\lambda_4)}{\partial P_1} &= F_1(P) - P_1^{-1}F_2(P_1,\lambda_1)P_1^{-1}\\ \frac{\partial J(P,\lambda_1,\lambda_4)}{\partial P_4} &= F_3(P_4) - P_4^{-1}F_4(P,\lambda_4)P_4^{-1}\end{aligned} \tag{18}$$

with

$$\begin{aligned}F_1(P) &= \sum_{i=0}^{n}\sigma_i^v K_i^h(P_1) + (1+\mathrm{tr}[K^v P_4^{-1}])W^h\\ F_2(P_1,\lambda_1) &= \sum_{i=0}^{n}\sigma_i^v W_i^h(P_1) + (\lambda_1+1)K^h\\ F_3(P_4) &= \sum_{i=0}^{m}\sigma_i^h K_i^v(P_4) + W^v\\ F_4(P,\lambda_4) &= \sum_{i=0}^{m}\sigma_i^h W_i^v(P_4) + (\lambda_4+1+\mathrm{tr}[W^h P_1])K^v\end{aligned}$$

$$\begin{bmatrix} K_i^h(P_1) & * \\ * & * \end{bmatrix} = \begin{bmatrix} A_1 & \mathbf{0} \\ \tilde{u}_i c_1 & A_1 \end{bmatrix}^T \begin{bmatrix} K_i^h(P_1) & * \\ * & * \end{bmatrix} \cdot \begin{bmatrix} A_1 & \mathbf{0} \\ \tilde{u}_i c_1 & A_1 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_1^{-1} \end{bmatrix}$$

$$\begin{bmatrix} W_i^v(P_4) & * \\ * & * \end{bmatrix} = \begin{bmatrix} A_4 & b_2\tilde{v}_i^T \\ \mathbf{0} & A_4 \end{bmatrix} \begin{bmatrix} W_i^v(P_4) & * \\ * & * \end{bmatrix} \cdot \begin{bmatrix} A_4 & b_2\tilde{v}_i^T \\ \mathbf{0} & A_4 \end{bmatrix}^T + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_4 \end{bmatrix}.$$

Hence the above KKT conditions become

$$\begin{aligned}P_1F_1(P)P_1 &= F_2(P_1,\lambda_1)\\ P_4F_3(P_4)P_4 &= F_4(P,\lambda_4).\end{aligned} \tag{19}$$

Two equations in (19) are highly nonlinear with respect to P_1 and P_4. An effective approach to solving two equations in (19) is to *relax* them into the following recursive second-order matrix equations:

$$\begin{aligned}P_1^{(i+1)}F_1(P^{(i)})P_1^{(i+1)} &= F_2(P_1^{(i)},\lambda_1^{(i+1)})\\ P_4^{(i+1)}F_3(P_4^{(i)})P_4^{(i+1)} &= F_4(P^{(i)},\lambda_4^{(i+1)})\end{aligned} \tag{20}$$

with the initial condition $P^{(0)} = P_1^{(0)} \oplus P_4^{(0)} = I_{m+n}$. The solutions $P_1^{(i+1)}$ and $P_4^{(i+1)}$ of (20) are given by

$$\begin{aligned}P_1^{(i+1)} &= F_1^{-\frac{1}{2}}(P^{(i)})[F_1^{\frac{1}{2}}(P^{(i)})F_2(P_1^{(i)},\lambda_1^{(i+1)})\\ &\quad\cdot F_1^{\frac{1}{2}}(P^{(i)})]^{\frac{1}{2}}F_1^{-\frac{1}{2}}(P^{(i)})\\ P_4^{(i+1)} &= F_3^{-\frac{1}{2}}(P_4^{(i)})[F_3^{\frac{1}{2}}(P_4^{(i)})F_4(P^{(i)},\lambda_4^{(i+1)})\\ &\quad\cdot F_3^{\frac{1}{2}}(P_4^{(i)})]^{\frac{1}{2}}F_3^{-\frac{1}{2}}(P_4^{(i)})\end{aligned} \tag{21}$$

respectively. Here, Lagrange multipliers $\lambda_1^{(i+1)}$ and $\lambda_4^{(i+1)}$ can be efficiently obtained using a bisection method so that

$$\begin{aligned}f_1(\lambda_1^{(i+1)}) &= m - \mathrm{tr}[\tilde{K}_h^{(i)}\tilde{F}_2^{(i)}(\lambda_1^{(i+1)})] = 0\\ f_4(\lambda_4^{(i+1)}) &= n - \mathrm{tr}[\tilde{K}_v^{(i)}\tilde{F}_4^{(i)}(\lambda_4^{(i+1)})] = 0\end{aligned} \tag{22}$$

are satisfied where

$$\begin{aligned}\tilde{K}_h^{(i)} &= F_1^{\frac{1}{2}}(P^{(i)})K^hF_1^{\frac{1}{2}}(P^{(i)})\\ \tilde{K}_v^{(i)} &= F_3^{\frac{1}{2}}(P_4^{(i)})K^vF_3^{\frac{1}{2}}(P_4^{(i)})\\ \tilde{F}_2^{(i)}(\lambda_1^{(i+1)}) &= [F_1^{\frac{1}{2}}(P^{(i)})F_2(P_1^{(i)},\lambda_1^{(i+1)})F_1^{\frac{1}{2}}(P^{(i)})]^{-\frac{1}{2}}\\ \tilde{F}_4^{(i)}(\lambda_4^{(i+1)}) &= [F_3^{\frac{1}{2}}(P_4^{(i)})F_4(P^{(i)},\lambda_4^{(i+1)})F_3^{\frac{1}{2}}(P_4^{(i)})]^{-\frac{1}{2}}.\end{aligned}$$

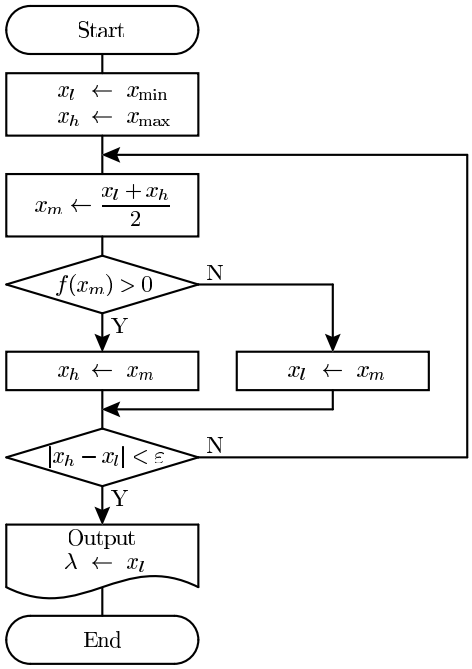


Figure 1: A flow chart of the bisection method.

A flow chart of the above bisection method is shown in Fig. 1. The iteration process continues until

$$|J(P^{(i+1)},\lambda_1^{(i+1)},\lambda_4^{(i+1)}) - J(P^{(i)},\lambda_1^{(i)},\lambda_4^{(i)})| < \varepsilon \tag{23}$$

is satisfied for a prescribed tolerance $\varepsilon > 0$. If the iteration is terminated at step i, then $P^{(i)}$ is viewed as a solution point.

Once positive-definite symmetric matrices P_1 and P_4 satisfying $\mathrm{tr}[K_1P_1^{-1}] = m$ and $\mathrm{tr}[K_4P_4^{-1}] = n$ were obtained, it is possible to construct an $m \times m$ orthogonal matrix U_1 and an $n \times n$ orthogonal matrix U_4 so that matrix $T = P_1^{1/2}U_1 \oplus P_4^{1/2}U_4$ satisfies L_2-scaling constraints in (14). (Hinamoto et al., 2005)

4 ILLUSTRATIVE EXAMPLE

Suppose that a 2-D separable-denominator digital filter $\{A_1^o, A_2^o, A_4^o, b_1^o, b_2^o, c_1^o, c_2^o, d\}_{3+3}$ in (1) is specified by

$$A_1^o = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0.599655 & -1.836929 & 2.173645 \end{bmatrix}$$

$$A_2^o = \begin{bmatrix} 0.064564 & 0.033034 & 0.012881 \\ 0.091213 & 0.110512 & 0.102759 \\ 0.097256 & 0.151864 & 0.172460 \end{bmatrix}$$

$$A_4^o = \begin{bmatrix} 0 & 0 & 0.564961 \\ 1 & 0 & -1.887939 \\ 0 & 1 & 2.280029 \end{bmatrix}$$

$$b_1^o = \begin{bmatrix} 0.047053 \\ 0.062274 \\ 0.060436 \end{bmatrix}, \qquad b_2^o = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$c_1^o = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

$$c_2^o = \begin{bmatrix} 0.016556 & 0.012550 & 0.008243 \end{bmatrix}$$

$$d = 0.019421.$$

By performing the l_2-scaling for the above LSS model with a diagonal coordinate-transformation matrix $T^o = T_1^o \oplus T_4^o$ where

$$T_1^o = \text{diag}\{0.992289, 0.987696, 0.964582\}$$

$$T_4^o = \text{diag}\{4.636056, 10.980193, 8.012802\}$$

we obtained

$$A_1 = \begin{bmatrix} 0.000000 & 0.995371 & 0.000000 \\ 0.000000 & 0.000000 & 0.976599 \\ 0.616880 & -1.880945 & 2.173645 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0.301648 & 0.365538 & 0.104015 \\ 0.428136 & 1.228560 & 0.833645 \\ 0.467440 & 1.728723 & 1.432628 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 0.000000 & 0.000000 & 0.976460 \\ 0.422220 & 0.000000 & -1.377725 \\ 0.000000 & 1.370331 & 2.280029 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} 0.047419 & 0.063050 & 0.062655 \end{bmatrix}^T$$

$$b_2 = \begin{bmatrix} 0.215701 & 0.000000 & 0.000000 \end{bmatrix}^T$$

$$c_1 = \begin{bmatrix} 0.992289 & 0.000000 & 0.000000 \end{bmatrix}$$

$$c_2 = \begin{bmatrix} 0.076755 & 0.137801 & 0.066050 \end{bmatrix}$$

and the l_2-sensitivity of the scaled LSS model was found to be

$$M_2 = 4526.0790.$$

Choosing $P^{(0)} = P_1^{(0)} \oplus P_4^{(0)} = I_6$ in (21) as initial estimate, $x_{min} = -2^{20}$ and $x_{max} = 2^{20}$ in the bisection method, and tolerance $\varepsilon = 10^{-8}$ in Fig. 1 and (23), it took the proposed algorithm 15 iterations to converge to the solution $P^{opt} = P_1^{opt} \oplus P_4^{opt}$ where

$$P_1^{opt} = \begin{bmatrix} 0.992455 & 0.702756 & 0.373871 \\ 0.702756 & 0.724033 & 0.597920 \\ 0.373871 & 0.597920 & 0.674661 \end{bmatrix}$$

$$P_4^{opt} = \begin{bmatrix} 2.200512 & -2.005367 & 1.676709 \\ -2.005367 & 1.913721 & -1.647192 \\ 1.676709 & -1.647192 & 1.480797 \end{bmatrix}$$

or equivalently, $T^{opt} = T_1^{opt} \oplus T_4^{opt}$ where

$$T_1^{opt} = \begin{bmatrix} -0.975337 & -0.066061 & 0.191859 \\ -0.619458 & 0.147201 & 0.564479 \\ -0.291519 & 0.450550 & 0.621839 \end{bmatrix}$$

$$T_4^{opt} = \begin{bmatrix} -0.799684 & 0.585116 & -1.103928 \\ 0.493843 & -0.684596 & 1.095978 \\ -0.336031 & 0.804236 & -0.849167 \end{bmatrix}.$$

The minimized l_2-sensitivity measure in (17) corresponding to the above solution was found to be

$$J(P^{opt}, \lambda_1, \lambda_4) = 101.0064$$

with $\lambda_1 = 4.786834$ and $\lambda_4 = -4.094596$. By substituting $T = T^{opt}$ obtained above into (12), the optimal state-space filter structure that minimizes (13) subject to the l_2-scaling constraints in (14) was synthesized as

$$\overline{A}_1 = \begin{bmatrix} 0.694418 & -0.112298 & -0.412379 \\ -0.096981 & 0.765920 & -0.345179 \\ 0.282990 & 0.456524 & 0.713306 \end{bmatrix}$$

$$\overline{A}_2 = \begin{bmatrix} 0.138105 & -0.073790 & 0.140661 \\ -0.132057 & 0.634682 & -0.262494 \\ 0.158022 & -0.104957 & 0.516782 \end{bmatrix}$$

$$\overline{A}_4 = \begin{bmatrix} 0.699418 & -0.018435 & 0.273811 \\ -0.091049 & 0.837579 & 0.358967 \\ -0.257686 & -0.254075 & 0.743031 \end{bmatrix}$$

$$\overline{b}_1 = \begin{bmatrix} -0.038277 & 0.028296 & 0.062312 \end{bmatrix}^T$$

$$\overline{b}_2 = \begin{bmatrix} -0.758218 & 0.129041 & 0.422255 \end{bmatrix}^T$$

$$\overline{c}_1 = \begin{bmatrix} -0.967816 & -0.065551 & 0.190380 \end{bmatrix}$$

$$\overline{c}_2 = \begin{bmatrix} -0.015522 & 0.003691 & 0.010209 \end{bmatrix}$$

whose horizontal and vertical controllability Gramians were given by

$$K_{opt}^h = \begin{bmatrix} 1.000000 & -0.090933 & -0.400242 \\ -0.090933 & 1.000000 & 0.400242 \\ -0.400242 & 0.400242 & 1.000000 \end{bmatrix}$$

$$K_{opt}^v = \begin{bmatrix} 1.000000 & -0.126238 & -0.520618 \\ -0.126238 & 1.000000 & 0.520618 \\ -0.520618 & 0.520618 & 1.000000 \end{bmatrix}.$$

Profile of the l_2-sensitivity measure, and profile of the parameters λ_1 and λ_4 during the first 15 iterations of the proposed algorithm are shown in Figs. 2 and 3, respectively.

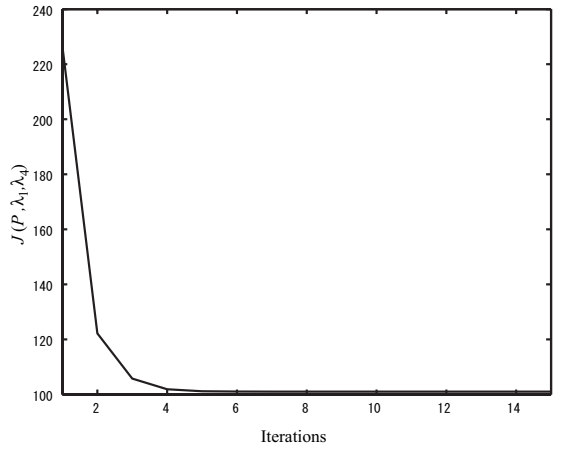


Figure 2: l_2-Sensitivity Performance.

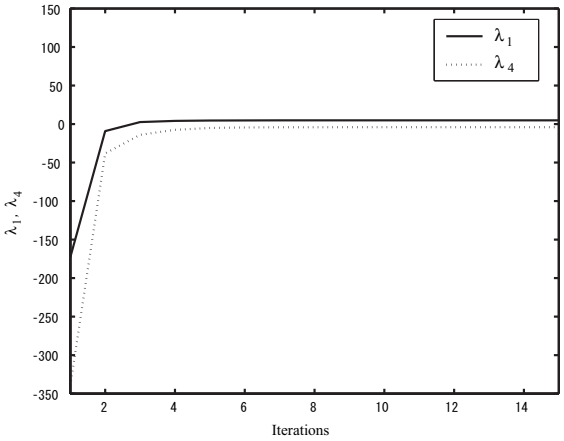


Figure 3: λ_1 and λ_4 Performances.

5 CONCLUSION

The problem of minimizing the l_2-sensitivity measure subject to l_2-scaling constraints for 2-D separable-denominator state-space digital filters has been formulated. An iterative method for minimizing l_2-sensitivity subject to l_2-scaling constraints has been explored. This has been performed by using a Lagrange function and an efficient bisection method. Computer simulation results have demonstrated the validity and effectiveness of the proposed technique.

REFERENCES

Hinamoto, T. (1980). Realization of a state-space model from two-dimensional input-output map. *IEEE Trans. Circuits Syst.*

Hinamoto, T., Iwata, K., and Lu, W.-S. (2005). State-space digital filters with minimum l_2-sensitivity subject to l_2-scaling constraints. In *Proc. 2005 IEEE Int. Conf. Acoust., Speech, Signal Processing*.

Hinamoto, T., Ohnishi, H., and Lu, W.-S. (2004). Minimization of l_2-sensitivity for 2-d state-space digital filters subject to l_2-scaling constraints. In *Proc. 2004 IEEE Int. Symp. Circuits Syst.*

Hinamoto, T. and Sugie, Y. (2002). l_2-sensitivity analysis and minimization of 2-d separable-denominator state-space digital filters. *IEEE Trans. Signal Processing*.

Hinamoto, T. and Takao, T. (1992). Synthesis of 2-d state-space filter structures with low frequency-weighted sensitivity. *IEEE Trans. Circuits Syst. II*.

Hinamoto, T., Takao, T., and Muneyasu, M. (1992). Synthesis of 2-d separable-denominator digital filters with low sensitivity. *J. Franklin Institute*.

Hinamoto, T., Yokoyama, S., Inoue, T., Zeng, W., and Lu, W.-S. (2002). Analysis and minimization of l_2-sensitivity for linear systems and two-dimensional state-space filters using general controllability and observability gramians. *IEEE Trans. Circuits Syst. I*.

Hinamoto, T., Zempo, Y., Nishino, Y., and Lu, W.-S. (1999). An analytical approach for the synthesis of two-dimensional state-space filter structures with minimum weighted sensitivity. *IEEE Trans. Circuits Syst. I*.

Hwang, S. Y. (1977). Minimum uncorrelated unit noise in state-space digital filtering. *IEEE Trans. Acoust., Speech, Signal Processing*.

Kawamata, M. and Higuchi, T. (1986). Synthesis of 2-d separable denominator digital filters with minimum roundoff noise and no overflow oscillations. *IEEE Trans. Circuits Syst.*

Kawamata, M., Lin, T., and Higuchi, T. (1987). Minimization of sensitivity of 2-d state-space digital filters and its relation to 2-d balanced realizations. In *Proc. 1987 IEEE Int. Symp. Circuits Syst*.

Kung, S. Y., Levy, B. C., Morf, M., and Kailath, T. (1977). New results in 2-d systems theory, part ii: 2-d state-space models -realization and the notions of controllability, observability, and minimality. In *Proc. IEEE*.

Li, G. (1998). Two-dimensional system optimal realizations with l_2-sensitivity minimization. *IEEE Trans. Signal Processing*.

Mullis, C. T. and Roberts, R. A. (1976). Synthesis of minimum roundoff noise fixed-point digital filters. *IEEE Trans. Circuits Syst.*

Roesser, R. P. (1975). A discrete state-space model for linear image processing. *IEEE Trans. Automat. Contr.*

MODIFIED MODEL REFERENCE ADAPTIVE CONTROL FOR PLANTS WITH UNMODELLED HIGH FREQUENCY DYNAMICS

L. Yang, S. A. Neild and D. J. Wagg
Department of Mechanical Engineering, University of Bristol, Queens Building, University Walk, Bristol BS8 1TR, U.K.
lin.yang@bristol.ac.uk, simon.neild@bristol.ac.uk, david.wagg@bristol.ac.uk

Keywords: Model reference adaptive control, Robustness, Unmodelled dynamics, Frequency response technique.

Abstract: In this paper we develop a modified MRAC strategy for use on plants with unmodelled high frequency dynamics. The MRAC strategy is made up of two parts, an adaptive control part and a fixed gain control part. The adaptive algorithm uses a combination of low and high pass filters such that the frequency range for the adaptive part of the strategy is limited. This reduces adaptation to unexpected high frequency dynamics and removes low frequency gain wind-up. In this paper we consider two examples of plants with unmodelled high frequency dynamics, both of which exhibit unstable behaviour when controlled using the standard MRAC strategy. By using the modified strategy we demonstrate that robustness is significantly improved.

1 INTRODUCTION

Two of the major challenges in the application of model reference adaptive control (MRAC) strategies are disturbances and plant uncertainty (Aström and Wittenmark, 1995; Sastry and Bodson, 1989; Landau, 1979; Popov, 1973). One effect of disturbances, such as transducer noise, is that control gains can 'wind-up' (Ioannou and Kokotovic, 1984; Virden and Wagg, 2005). An effective way to remove gain wind-up behaviour is to eliminate the inherent zero eigenvalue in the (localised) MRAC system by introducing a complementary low pass filter (Yang et al., 2006). Plants with unmodelled high frequency dynamics are one important case of plant uncertainty, and previous studies have shown how this can cause system instability in many real applications (Rohrs et al., 1985; Nikzad et al., 1996; Crewe, 1998; Neild et al., 2005b).

As an example of using MRAC on plants with higher order unmodelled dynamics, we consider the application of the MRAC to hydraulic shaking tables. Hydraulic shaking tables are widely used in the earthquake engineering community for dynamic testing of structures subjected to extreme loading. Adaptive control is desirable due to the changing dynamics of the test specimen attached to the table when exposed to extreme loading (Stoten and Gómez, 2001). Generally hydraulic actuators may be modelled as first order systems (Neild et al., 2005a), however attaching a large mass, such as the table and payload, to the actuator can lead to significant higher frequency dynamics due to oil column resonance (Nikzad et al., 1996; Crewe, 1998; Neild et al., 2005b).

In this paper we present a modified MRAC algorithm which uses complementary filters at both low and high frequency. We demonstrate that when this new modified MRAC algorithm is applied to systems with unmodelled high frequency dynamics a stable response can be achieved.

2 FORMULATIONS OF MODIFIED MRAC STRATEGY

In this section a brief introduction of ρ/ϕ modified MRAC algorithm is given for a single-input single-output (SISO) system. For more detailed discussions of standard MRAC can be found in (Landau, 1979; Sastry and Bodson, 1989; Aström and Wittenmark, 1995). The system studied in this paper is based on a first-order linear plant approximation given by the transfer function $G(s) = X(s)/U(s) = b/(s+a)$, where $X(s)$ is the plant state ($x(t)$ in the time domain),

$U(s)$ is the control signal and a and b are the plant parameters. The control signal is generated from the state variable and the reference (or demand) signal $r(t)$, using adaptive control gains K and K_r, such that $u(t) = Kx(t) + K_r r(t)$, where K is the feedback adaptive gain and K_r the feed forward adaptive gain. The plant is controlled to follow the output from a reference model $G_m(s) = X_m(s)/R(s) = b_m/(s+a_m)$, where X_m is the state of the reference model and a_m and b_m are the reference model parameters which are specified by the controller designer. The block diagram of MRAC is illustrated by Fig.1.

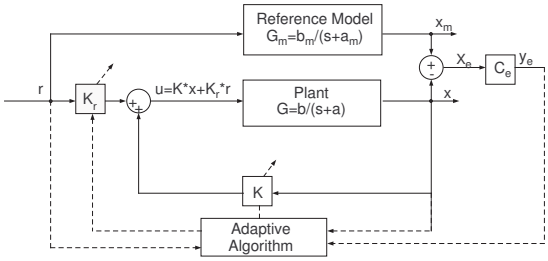


Figure 1: Schematic block diagram of the model reference adaptive control system. K and K_r are the adaptive gains generated using the MRAC algorithm.

The object of the MRAC algorithm is for $x_e \to 0$ as $t \to \infty$, where $x_e = x_m - x$ is the error signal. The dynamics of the system can be rewritten in terms of the error such that

$$\dot{x}_e = (-a + bK)x_e + b(K^E - K)x_m + b(K_r^E - K_r)r, \tag{1}$$

where K^E and K_r^E are Erzberger gains. The Erzberger gains are defined as the linear gains which results in the plant response matching the reference model response (Khalil, 1992);

$$K^E = \frac{a - a_m}{b}, \;\; K_r^E = \frac{b_m}{b}. \tag{2}$$

For general model reference adaptive control, the adaptive gains are commonly defined by using Hyperstability rule (Popov, 1973), which is a proportional plus integral formulation

$$\begin{aligned} K(t) &= \alpha \textstyle\int_0^t C_e x_e x(\tau)\mathrm{d}\tau + \beta C_e x_e x(t) + K_0, \\ K_r(t) &= \alpha \textstyle\int_0^t C_e x_e r(\tau)\mathrm{d}\tau + \beta C_e x_e r(t) + K_{r0}, \end{aligned} \tag{3}$$

where α and β are adaptive control weightings representing the adaptive effort and K_0 and K_{r0} are the initial gain values. In the case of a first-order implementation, C_e is a scalar and therefore may be incorporated into the α and β adaptive control weightings.

2.1 Mrac with ρ Modification

The purpose of introducing the ρ modification to the MRAC algorithm is to resolve the problem of gain 'wind-up' observed using standard the MRAC strategy on plants with output disturbances. The modified adaptive gains $K_{m\rho}$ and $K_{rm\rho}$ are given by

$$\begin{aligned} K_{m\rho}(s) &= \tfrac{s}{s+\rho^2}K(s) + \tfrac{\rho^2}{s+\rho^2}K^*(s), \\ K_{rm\rho}(s) &= \tfrac{s}{s+\rho^2}K_r(s) + \tfrac{\rho^2}{s+\rho^2}K_r^*(s), \end{aligned} \tag{4}$$

where ρ is a constant, $K(s)$ and $K_r(s)$ are the standard adaptive control gains in the Laplace domain, and $K^*(s)$ and $K_r^*(s)$ are constant gains. This modification eliminates a zero eigenvalue in the localised error dynamics about the equilibrium point, replacing it with an eigenvalue of $-\rho^2$, hence making all the system eigenvalues asymptotically stable (Yang et al., 2006). The ρ modified MRAC can also be explained in terms of frequency response. A bode plot of Eq.4 is shown in Fig.2(a), we can see how the ρ term works as a low frequency filter on the adaptive gains and stops gain wind-up by pushing gains to fixed values. Experimental tests have demonstrated the effectiveness of ρ modified MRAC on preventing gain wind-up in a small scale motor-driven shaking table (Yang et al., 2006).

2.2 Mrac with ρ/ϕ Modification

In this paper we present an additional modification to MRAC through the use of an additional high frequency complementary filter. A ϕ term is introduced as the complementary filter to reduce adaptation to high frequencies, e.g. due to the unmodelled dynamics. This is illustrated in Fig.2(b).

The ρ/ϕ modified MRAC control gains may be described in the Laplace domain as

$$\begin{aligned} K_m(s) = \;& \tfrac{\phi^2 s}{(s+\rho^2)(s+\phi^2)}K(s) + \tfrac{\rho^2}{s+\rho^2}K^*(s) \\ & + \tfrac{s^2}{(s+\rho^2)(s+\phi^2)}K^*(s), \\ K_{rm}(s) = \;& \tfrac{\phi^2 s}{(s+\rho^2)(s+\phi^2)}K_r(s) + \tfrac{\rho^2}{s+\rho^2}K_r^*(s) \\ & + \tfrac{s^2}{(s+\rho^2)(s+\phi^2)}K_r^*(s), \end{aligned} \tag{5}$$

where ρ and ϕ are constants which need to be selected by the designer, and K^* and K_r^* are steady-state gains, ideally they are set to the values of the Erzberger Gains. K and K_r are the standard MRAC control gains.

By inspecting Eq.5, we note that the modified control gain K_m is made up of an adaptive part and a fixed

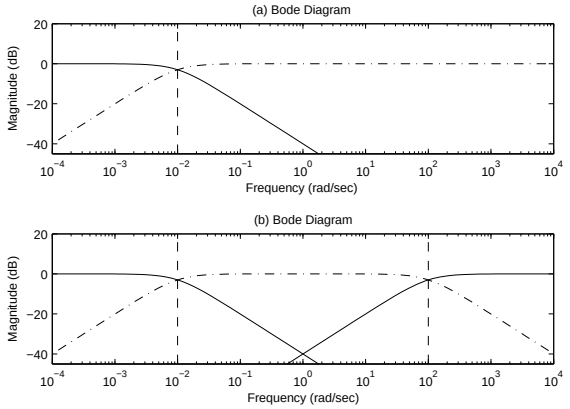


Figure 2: (a) ρ modified MRAC adaptive gain structure. Solid line represents the fixed gain control part, K^* or K_r^*, and dash-dot line represents gains adaptive part, K or K_r. The vertical dash line shows the value of ρ^2 corresponding to the complementary filters break point. (b) ρ and ϕ modified MRAC structure. Solid line represents the fixed gain control part K^* or K_r^*, and dash-dot line represents gains adaptive part K or K_r. Vertical dash line shows the value of ρ^2 and ϕ^2.

gain control part. The first term on the right hand side is the adaptive part, and the second and third terms are fixed gain control terms based on the constant steady-state gain $K^*(s)$. The same situation can be found in the modified gain K_{rm}. Given ρ and ϕ are non-zero real values, the fixed gain part of Eq.5 has all poles on left half plane, hence this part is stable. Now we focus on the stability of the adaptive part of modified control gains. By applying the Laplace transform given zero initial conditions to Eq. 3 we have

$$\frac{K(s)-K_0}{P_1(s)}=\frac{\beta s+\alpha}{s},\quad \frac{K_r(s)-K_{r0}}{P_2(s)}=\frac{\beta s+\alpha}{s}, \quad (6)$$

where $P_1(s)=C_eX_e(s)X(s)$ and $P_2(s)=C_eX_e(s)R(s)$. We note that a zero pole exists which makes the transfer function marginally stable. Substituting $K(s)$ and $K_r(s)$ in Eq.5 by Eq.6, the adaptive part of Eq.5 becomes

$$\begin{aligned}\frac{K_m(s)-K_0}{P_1(s)}&=\frac{\phi^2(\beta s+\alpha)}{(s+\rho^2)(s+\phi^2)},\\ \frac{K_{rm}(s)-K_{r0}}{P_2(s)}&=\frac{\phi^2(\beta s+\alpha)}{(s+\rho^2)(s+\phi^2)}.\end{aligned} \quad (7)$$

Comparing Eq.7 with standard MRAC control gains of Eq.6, we noticed that the zero pole in the standard MRAC control gain is replaced by two negative poles, given ρ and ϕ are non-zero real values, and this makes the control gains asymptotically stable.

Now we consider the overall transfer function path from the input signal r to error signal e. Given plant transfer function G and reference model transfer function G_m, the transfer function from reference signal r to plant output x can be written as (Aström and Wittenmark, 1995)

$$G_c(s)=\frac{X(s)}{R(s)}=\frac{K_rG}{1-KG}. \quad (8)$$

So the error signal x_e becomes $X_e(s)=[G_m(s)-G_c(s)]R(s)$, hence the transfer function from reference signal r to error signal x_e can be written as $X_e(s)/R(s)=G_m-G_c$, substituting $G_c(s)$ by Eq.8 and rearranging it we have

$$\frac{X_e(s)}{R(s)}=\frac{G_m-KG_mG-K_rG}{1-KG}. \quad (9)$$

Since the transfer function of plant is $G(s)=b/(s+a)$ and the transfer function of reference model is $G_m(s)=b_m/(s+a_m)$, Eq.9 can be calculated as

$$\frac{X_e(s)}{R(s)}=\frac{b(K_r^E-K_r)s+b_mb(K^E-K)+a_mb(K_r^E-K_r)}{(s+a_m)(s+a-bK)}. \quad (10)$$

Eq.10 represents the error response of the overall system. We notice there are two poles $-a_m$ and $bK-a$ in this transfer function. To make the overall system stable, we need to ensure both poles are on left half plane. Since a_m is defined as positive, the $-a_m$ pole is on left-half plane. To make $bK-a<0$, the condition of $K<a/b$ need to be satisfied. We notice if $K=K^E=(a-a_m)/b$ this condition will always be satisfied.

As a further insight into Eq.5 and Eq.10, if $\rho=\infty$ and $\phi=0$, Eq.5 will become $K_m=K^*$ and $K_{rm}=K_r^*$, which means the system will be completely controlled by fixed gains. Hence to increase ρ from 0 and decrease ϕ from infinite means to add weights on fixed gain control. In Eq.10 if $K=K^*=K_r^E$ and $K_r=K_r^*=K_r^E$ the error signal will become zero, which means the system has ideal response. We therefore set K^* and K^* to our best estimate of the Erzberger gains K^E and K_r^E.

3 MODIFIED MRAC APPLIED TO ROHRS EXAMPLE

Knowledge of the Erzberger gains, to set K^* and K^*, is important to the accuracy to the ρ/ϕ modified MRAC algorithm. In many practical situations the Erzberger Gains can not be estimated precisely and in some cases can only be crudely approximated. One such case is a plant with unmodelled high frequency dynamics, for example 'Rohrs model' (Rohrs et al., 1985). In this section we show how the modified MRAC algorithm copes with Rohrs example. The plant transfer function is given as

$$G(s)=\frac{2}{(s+1)}\frac{229}{(s^2+30s+229)}, \quad (11)$$

which is a nominally first order plant $2/(s+1)$ multiply by a second order unmodelled dynamics $229/(s^2+30s+229)$ which has almost critical damping, $\zeta = 1.02$. The plant thus has two poles $s = -15 \pm 2i$ neglected in the model used to design the adaptive controller. The reference model is given as

$$G_m(s) = \frac{3}{s+3}. \tag{12}$$

The initial conditions for both control gains K and K_r are zero. As in Rohrs example the input signal is set as

$$r(t) = 0.3 + 1.85\sin(16.1t), \tag{13}$$

much higher frequency than the nominal first order plant break frequency (1 rad/sec). Nominal Erzberger gains can be calculated according to Eq.2 as $K^E = -1$ $K_r^E = 1.5$. The α/β ratio is chosen as 1, which is the same as nominal plant break frequency.

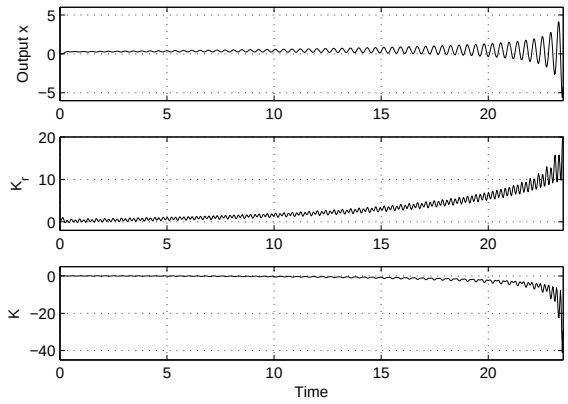


Figure 3: Standard MRAC with unmodelled dynamics (Rohrs model), input signal r(t)=0.3+1.85sin(16.1t), $\alpha = \beta = 1$. The system is unstable.

If the standard MRAC strategy is applied to the nominal first order plant, $G = 2/(s+1)$, the response is stable and the gains tend to the Erzberger values.

However if the higher frequency dynamics, as described by Eq.11 are included in the plant gain wind-up occurs which results in system instability. Fig.3 shows the system response with $\alpha = \beta = 1$, which results in system instability within 25 seconds.

Fig.4 shows the plant response, for the case where with high frequency dynamic are included, using ρ modified MRAC, $\rho^2 = 0.4$ and $\alpha = \beta = 1$. The modified strategy results in a stable response with no gain wind-up.

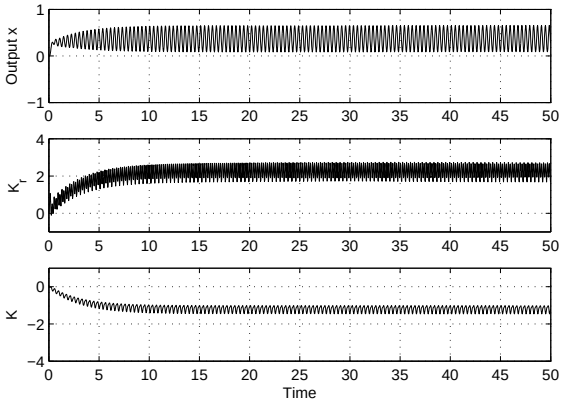


Figure 4: ρ modified MRAC with Rohrs model, input signal r(t)=0.3+1.85sin(16.1t), $\alpha = \beta = 1$, $\rho^2 = 0.4$. The system is stable.

4 MODIFIED MRAC APPLIED TO SHAKING TABLES

In this section, to demonstrate the difference in behaviour due to the ρ and the ϕ modifications, we consider the application of the MRAC strategy to control hydraulic shaking tables. Under general operating conditions, a large hydraulic shaking table used for earthquake tests will have a low frequency demand which is affected by high frequency dynamics due to oil column resonance. Typically system identification of hydraulic shaking tables over the low frequency operating range, around 0-10 Hz, results in a first order approximation to the system dynamics with the break frequency occurring within the operating range. However oil column resonance causes an unmodelled high frequency resonance with low damping, in the order of 10% of critical damping, (Nikzad et al., 1996; Crewe, 1998; Neild et al., 2005b).

To simulate this type of application, we will make the following changes to Rohrs example considered in the last section. Firstly, we change the demand signal frequency to 1 rad/sec, such that it coincides with the nominal plant break frequency:

$$r(t) = 0.3 + 1.85\sin(1t). \tag{14}$$

Secondly, we add white Gaussian noise to the plant output, resulting in an approximate signal to noise ratio of 20, to mimic transducer noise. Thirdly, we change the damping ratio of the higher frequency unmodelled dynamics to $\zeta = 0.1$ to represent the oil column resonance to give the overall plant transfer function:

$$G(s) = \frac{2}{(s+1)}\frac{229}{(s^2+3s+229)}, \tag{15}$$

where the nominal first order plant is still $2/(s+1)$, but the unmodelled dynamics becomes $229/(s^2+3s+229)$. A Bode plot of the plant is given in Fig.5(a). The reference model and other conditions remain unaltered.

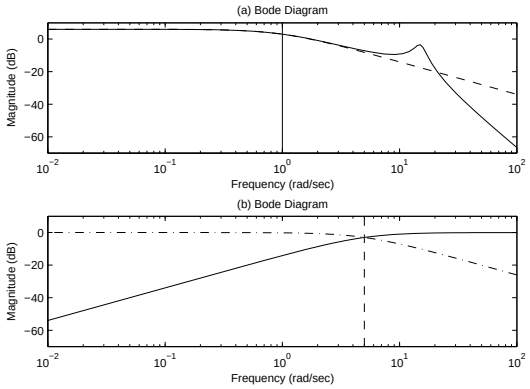


Figure 5: (a) Plant dynamics Bode plot: solid line shows the plant with unmodelled high frequency dynamics, the dash line is the nominal first order plant and the vertical line represents input signal operating frequency 1 rad/sec. (b) ϕ modified MRAC: dash-dot line represents adaptive part of the control gains K or K_r, solid line represents the fix part K^* or K_r^* and the vertical dash line represents the ϕ complementary filter break frequency; $\phi^2 = 5$ rad/sec.

As with Rohrs example, the standard MRAC strategy exhibits gain windup resulting in system instability when applied to the plant with higher frequency dynamics.

Fig.6 shows the control performance using ρ modified MRAC (with $\rho^2 = 0.5$ and $\alpha = \beta = 0.5$). We can see that, in contrast to Rohrs example, this system is still unstable despite the ρ modification. This is because the ρ modification is designed to removing windup rather than the gain oscillations that occur when the unmodelled higher order dynamics has low damping.

Fig.7 is the system response using the ϕ modified MRAC algorithm (with $\phi^2 = 5$ and $\alpha = \beta = 0.5$). The value of ϕ has been selected to reduce gain adaptation at the oil column resonance frequency of 11 rad/sec. The system is stable, with the error and both gains settle around 150 seconds. Comparing Fig.7 with Fig.6, we observe that ϕ plays a different role from ρ in the modified control algorithm. The ϕ modification results in filtering out the unmodelled high frequency dynamics directly to avoid the system adapting to these undesirable dynamics. In this example setting $\phi^2 = 5$ can minimise the gain adaptation to the oil column resonance, as illustrated by Fig.5(b) which shows the resulting complementary filters applied to the adaptive, K, and linear, K^*, gains.

Finally, Fig.8 shows the control result using the combined ρ/ϕ modified MRAC algorithm (with $\alpha = \beta = 0.5$, $\rho^2 = 0.5$ and $\phi^2 = 5$). The system has a stable response, with the error and gains settle within around 10 seconds – faster than when ϕ modified MRAC was used. The reason is that by increasing ρ the fixed gain contribution to the controller, which requires no time to settle, becomes more dominant.

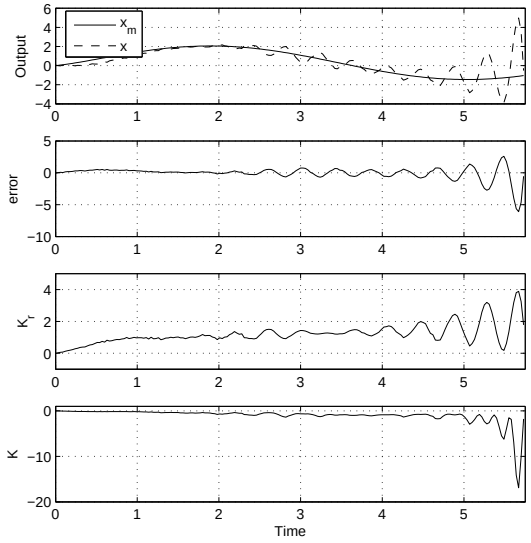


Figure 6: Plant with unmodelled high frequency dynamics, damping ratio 0.1, controlled by ρ modified MRAC. Input signal r(t)=0.3+1.85sin(1t), $\alpha = \beta = 0.5$, $\rho^2 = 0.5$. System is unstable.

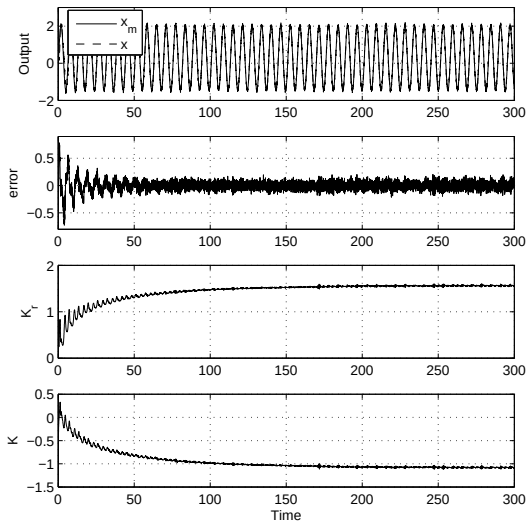


Figure 7: Plant with unmodelled high frequency dynamics, damping ratio 0.1, controlled by ϕ modified MRAC. Input signal r(t)=0.3+1.85sin(1t), $\alpha = \beta = 0.5$, $\phi^2 = 5$. System is stable. Error and gains settle within around 150 seconds.

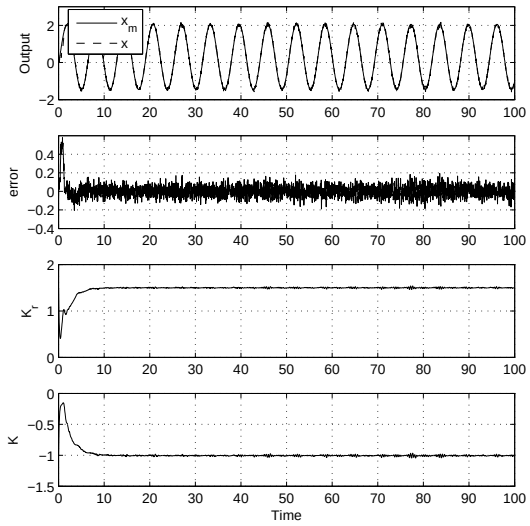


Figure 8: Plant with unmodelled high frequency dynamics, damping ratio 0.1, controlled by ρ/ϕ modified MRAC. Input signal r(t)=0.3+1.85sin(1t), $\alpha = \beta = 0.5$, $\rho^2 = 0.5$, $\phi^2 = 5$. System is stable. Error and gains settle within around 10 seconds.

5 CONCLUSION

In this paper we have introduced a ρ/ϕ modified MRAC strategy and tested it on plants with unmodelled high frequency dynamics. The modified MRAC strategy is made up of two parts, an adaptive control part and a fix gain control part. In the frequency domain, the ρ and ϕ modifications are first-order complementary filters which replace the adaptive gain with a fixed gain at low and high frequency respectively. Two types of unmodelled high frequency dynamics are considered. Firstly using Rohrs model, in which the unmodelled dynamics are almost critical damped, it was observed that ρ modified MRAC eliminated the gain wind-up. Secondly when the plant has lightly damped unmodelled dynamics case, similar to the oil column dynamics observed with hydraulic shaking table control, using ϕ modified MRAC prevents the system adapting to unmodelled high frequency dynamics, hence stabilizing the system. Simulation results show that ϕ modification results in filtering off the unmodelled high frequency dynamics directly to avoid the system adapting to these undesirable dynamics whereas the ρ modification eliminates gain wind-up. Hence the ρ/ϕ modified MRAC is a effective way to control systems with unmodelled high frequency dynamics.

ACKNOWLEDGEMENTS

The authors would also like to acknowledge the support of the EPSRC. Lin Yang is supported by the Dorothy Hodgkin Postgraduate Award scheme (EPSRC-BP) and David Wagg by an Advanced Research Fellowship.

REFERENCES

Aström, K. J. and Wittenmark, B. (1995). *Adaptive control*. Addison-Wesley, second edition.

Crewe, A. (1998). *The Characterisation and Optimisation of Earthquake Shaking Table Performance.* PhD thesis, University of Bristol.

Ioannou, P. and Kokotovic, P. (1984). Instability analysis and improvement of robustness of adaptive control. *Automatica*, 20(5):583–594.

Khalil, H. (1992). *Nonlinear Systems*. Macmillan:New York.

Landau, Y. (1979). *Adaptive control:The model reference approach*. Marcel Dekker:New York.

Neild, S., Drury, D., and Stoten, D. (2005a). An improved substructuring control strategy based on the mcs control algorithm. *Proceedings of the I. Mech. E. Part I, Journal of Systems and Control Engineering*, 219(5):305–317.

Neild, S., Stoten, D., Drury, D., and Wagg, D. (2005b). Control issues relating to real-time substructuring experiments using a shaking table. *Earthquake Engineering and Structural Dynamics*, 34(9):1171–1192.

Nikzad, K., Ghaboussi, J., and Paul, S. (1996). Actuator dyanamics and delay compensation using neurocontrollers. *Journal of Engineering Mechanics*, 122-10:966–975.

Popov, V. (1973). *Hyperstability of control systems*. Springer.

Rohrs, C., Valavani, L., Athans, M., and Stein, G. (1985). Robustness of continuous-time adaptive control algorithms in the presence of unmodeled dynamics. *IEEE Trans. Automat. Contr*, AC-30:881–889.

Sastry, S. and Bodson, M. (1989). *Adaptive control : Stability, convergence and robustness*. Prentice-Hall : New Jersey.

Stoten, D. and Gómez, E. (2001). Real-time adaptive control of shaking tables using the minimal control synthesis algorithm. *Phil. Trans. R Soc. Lond. A.*, 359:1697–1723.

Virden, D. and Wagg, D. (2005). System identification of a mechanical system with impacts using model reference adaptive control. *Proc. IMechE. Pat I: J. Syst. Control Eng.*, 219:121–132.

Yang, L., Neild, S., Wagg, D., and Virden, D. (2006). Model reference adaptive control of a nonsmooth dynamical system. *Nonlinear Dynamics*, 46(3):323–335.

IMPEDANCE MATCHING CONTROLLER FOR AN INDUCTIVELY COUPLED PLASMA CHAMBER

L-type Matching Network Automatic Controller

Giorgio Bacelli, John V. Ringwood and Petar Iordanov
Department of Electronic Engineering, National University of Ireland, Maynooth, Ireland
gbacelli@eeng.nuim.ie, john.ringwood@eeng.nuim.ie, Petar.Iordanov@eeng.nuim.ie

Keywords: Automatic Impedance Matching, Matching Network, Impedance Sensor, Inductively Coupled Plasma.

Abstract: Plasma processing is used in a variety of industrial systems, including semiconductor manufacture (deposition and etching) and accurate control of the impedance matching network is vital if repeatable quality is to be achieved at the manufacturing process output. Typically, impedance matching networks employ series (tune) and parallel (load) capacitors to drive the reflection coefficient on the load side of the network to zero. The reflection coefficient is normally represented by real and imaginary parts, giving two variables to be controlled using the load and tune capacitors. The resulting problem is therefore a nonlinear, multivariable control problem. Current industrial impedance matching units employ simple single-loop proportional controllers, which take no account of interaction between individual channels and, in many cases, may fail to tune altogether, if the starting point is far away from the matching point. A hierarchical feedback controller is developed which, at the upper level, performs a single-loop tuning, but with the important addition of a variable sign feedback gain. When convergence to a region in the neighbourhood of the matching point is achieved, a dual single-loop controller takes over, which gives fine tuning of the matching network.

1 INTRODUCTION

The BAsic Radio frequency Inductive System (BARIS) is an experimental inductively coupled plasma chamber used to study the closed-loop control of plasma states. Inductively coupled plasma is ignited by an electromagnetic field irradiated from an antenna connected to a Radio Frequency (RF) power supply. An Impedance Matching Unit (IMU) is used to match the impedance of the antenna to the impedance of the generator, in order to deliver the maximum power to the plasma. The IMU is composed of a matching network, a Phase and Magnitude Detector (PMD) and a controller that automatically tunes the matching network using the information supplied by the PMD. Each time plasma parameters or plasma state set-points are changed (i.e. RF power, pressure, gas mixture), the plasma impedance also changes. In addition, when the controller is tuning the matching network, the reflection coefficient is decreasing, therefore the power delivered to the plasma is increasing causing a variation of the plasma states and, as a consequence, a variation of plasma impedance. The main issue regarding the existing driver circuitry associated with the original controller is the global convergence (Mazza, 1970), that is, if the initial conditions of the system are far away from the matching point, the controller may not be able to tune the matching network.

The automatic impedance matching problem has been solved using neural networks (Vai and Prasad, 1993), genetic algorithms (Thompson and Fidler, 2000) (Sun and J.K., 1997) (Sun and J.K., 1999), deterministic tuning algorithms with look-up tables (Moritz and Sun, 2001) and using adaptive systems (Parro and Pait, 2003) (Ida et al., 2004c) (Ida et al., 2004a) (De Mingo et al., 2004) (Ida et al., 2004b); nonlinear control systems have been also considered (Cottee, 2003). In all of the above mentioned cases, the load impedance is not affected by the matching conditions while, in the case studied (inductively coupled plasma discharges), the load impedance is varying during the matching process. In this paper a hierarchal structure controller has been designed; it is composed of

a higher level coarse controller that drives the system close to the matching point, and lower level feedback controller for the fine tuning. An impedance sensor has been also designed to supply more reliable measurements of the reflection coefficient to the controller.

2 BARIS IMPEDANCE MATCHING

The BARIS is an experimental plasma chamber, used to study plasma phenomenon for applications in semiconductor manufacturing, and generates an argon and oxygen plasma that is ignited by a 13.56MHz magnetic field irradiated from an antenna. The main parts that compose this device are the plasma discharge chamber, the RF power supply, the matching unit and the real time monitoring and control of the system.

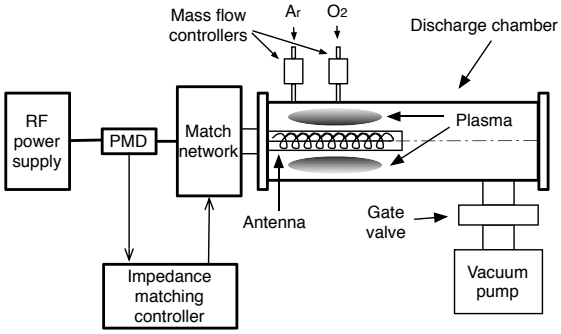


Figure 1: BARIS block diagram.

2.1 Plasma Discharge Chamber

The plasma discharge chamber is a stainless steel cylindrical vacuum chamber of internal diameter 200mm and length 900mm (Fig.1). The helical antenna is placed along the axis of the chamber, inside a sealed 50mm diameter quartz tube, in order to keep it outside of the vacuum region. The gasses are injected into the chamber by the mass flow controllers and are evacuated through the gate valve using a vacuum pump. The pressure at which the plasma is ignited is usually between 10$mTorr$ and 100$mTorr$, and it is regulated by adjusting the position of the gate valve and the gas flows.

2.2 RF Power Supply

The RF power generator is the ACG-10B made by MKS Instruments, which can deliver a maximum of 1000W at a frequency of 13.56 MHz into a 50Ω load.

2.3 Plasma Process Monitoring and Control

The control of the plasma process is achieved using the Matlab xPC Target anvironment. This system is composed of two PCs, one running Windows XP (Host PC) and the other one running the real time xPC Target operative system (Target PC) as in Fig.2. The

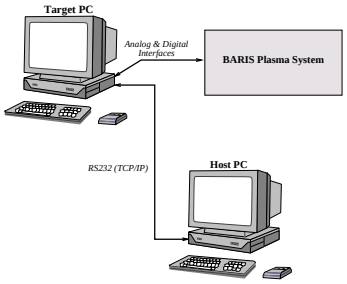


Figure 2: Matching network schematic.

Target PC is equipped with analog and digital interfaces in order to read data from sensors and control actuators and other devices. The role of the Host PC is to upload the software to be executed in real time by the Target PC, to start it, stop it and to monitor it while running using the RS-232 interface. This kind of configuration gives a considerable amount of computational power, allowing the implementation of complex control algorithm for the plasma process (Iordanov et al., 2006).

2.4 Matching Network

The matching network transforms the plasma load impedance (Z_{PL}) into the $Z_0 = 50\Omega$ characteristic impedance of the transmission line. It is a basic "L" configuration (Fig.3) characterized by eq.(1) and composed of "Load" (C_L) and "Tune"(C_T) variable capacitors , both driven by servomotors.

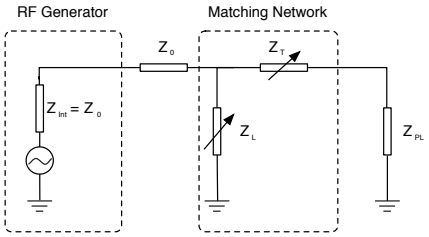


Figure 3: Matching network schematic.

$$Z_{PL} = \left[\frac{Z_0 Z_L}{Z_0 + Z_L} + Z_T\right]^*$$

$$= \left[\frac{Z_0}{(1+\omega^2 Z_0^2 C_L^2)} + j\frac{(1+\omega^2 Z_0^2 C_L (C_L + C_T))}{\omega C_T (1+\omega^2 Z_0^2 C_L^2)}\right]^* \quad (1)$$

with:

$$Z_T = \frac{1}{j\omega C_T}, \quad Z_L = \frac{1}{j\omega C_L}, \quad \omega = 2\pi 13.56 \cdot 10^6 rad/s$$

where $[...]^*$ denotes complex conjugation and ω is the circular frequency.

3 SENSOR

The impedance sensor is based on the Analog Devices AD8302 phase and gain detector, which gives information about the amplitude ratio and the phase difference between two signals. The inputs of the circuit are two sinusoidal signals proportional to the voltage and the current waves in the power line respectively. By measuring the ratio between voltage and current and their phase difference, it is possible to calculate the impedance or the reflection coefficient. The impedance of a load connected in a transmission line is defined as (2),

$$Z_L = \frac{V_0}{I_0} \quad (2)$$

where V_0 and I_0 are the vectors of voltage and current respectively measured on the load and Z_L the load impedance. The last expression can be written using the vectors in the exponential form as in (3):

$$Z_L = \frac{V_0}{I_0} = \frac{|V_0| \cdot e^{j\theta_V}}{|I_0| \cdot e^{j\theta_I}} = \frac{|V_0|}{|I_0|} \cdot e^{j(\theta_V - \theta_I)} = G \cdot e^{\Delta\theta} \quad (3)$$

$$G = \frac{|V_0|}{|I_0|} \qquad \Delta\theta = (\theta_V - \theta_I)$$

where G is the ratio between the voltage and the current magnitudes and $\Delta\theta$ is the phase difference between voltage and current waves. The impedance sensor provides two analog signals that are proportional to G and $\Delta\theta$. This device is divided in two parts, the "V-I Sensor" and the "Phase and Gain Sensor", each one enclosed in a shielded aluminum box in order to attenuate the effect of radio frequency disturbances (Fig.4). The former is connected along the high power transmission line, and supplies two signals proportional to the voltage and the current of the main line. The "Phase and Gain Sensor" takes the output signals of the "V-I Sensor" and provides their phase difference and amplitude ratio. At the inputs of the AD8302 there are two integrated low pass filters (MINI-CIRCUITS SCLF-10.7) in order to remove the harmonics components.

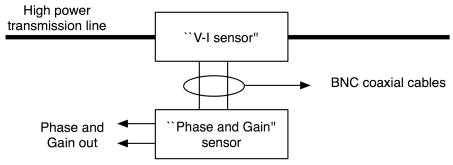


Figure 4: Block schematic of the impedance sensor.

4 CONTROLLER

The main property required of the controller is global convergence, that is the ability to drive the capacitors to the matching point from any starting condition. A model of the plasma impedance has been studied (Keville et al., 2006), but it is quite complicated, not suitable for the problem of the impedance matching because it is computationally demanding. The dynamics of the plasma process are stable and time invariant, in addition the part related to the RF power delivered P_D (Fig.5) is much faster than the dynamic of the matching unit, therefore it has been decided to consider the variation of the plasma impedance during the matching as a static disturbance. In this case the only dynamics terms considered in the system are due to the servomotors described by $G(s)$. Considering

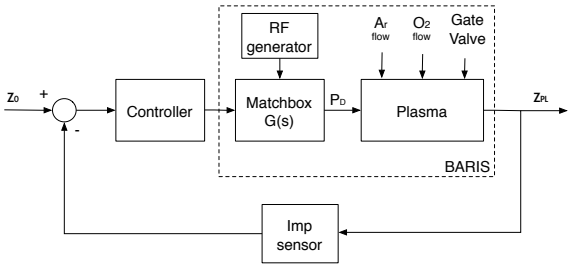


Figure 5: Block schematic of the BARIS.

the magnitude of the reflection coefficient $|\Gamma|$ as a function of the capacitors C_L and C_T, for a given value of the load impedance (Z_{PL}), using (1) is possible to plot the graph in Fig.6. The main characteristic of this function is that there is only one critical point corresponding to the global minimum, that is the matching point ($|\Gamma| = 0$). In this situation, the control problem

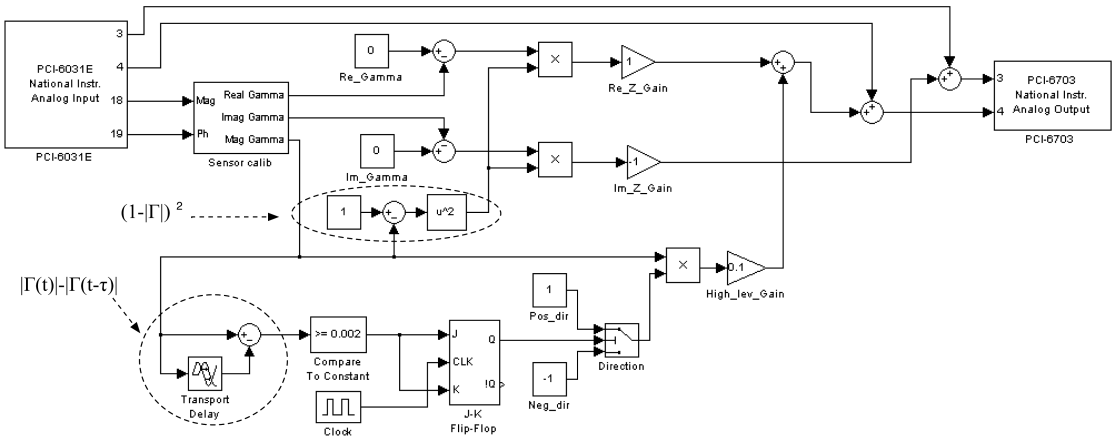


Figure 8: Simulink implementation of the hierarchical controller.

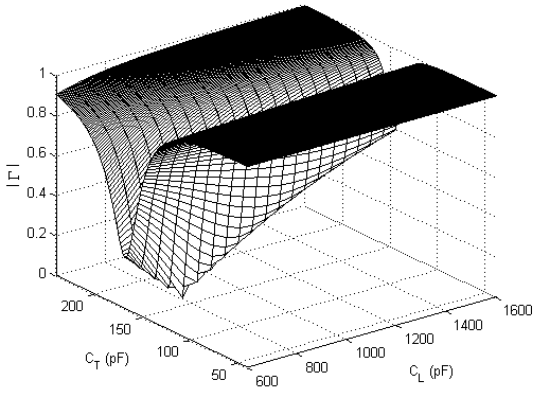


Figure 6: $|\Gamma|$ as function of (C_L, C_T).

can be considered as a function minimization problem. A first possible approach can be to drive the capacitors in the opposite direction of the gradient of $|\Gamma|$, but the plasma impedance is variable and its value it is unknown, therefore it is not possible to calculate this vector. From Fig.6 is possible to see that $|\Gamma|$ is significantly more sensitive respect to C_T than C_L. We have decided to use a hierarchical structure composed of two parts: a coarse and a fine tune controller. The coarse controller brings the system close to the matching point, where the fine tune controller takes over and drives the capacitors to the final position. The coarse controller is based on an iterative minimization algorithm for $|\Gamma|$ respect to C_T, as in the flow chart in fig.7. At regular intervals of time δ it checks if the reflection coefficient and if it is increasing, it inverts the direction of movement of C_T. When

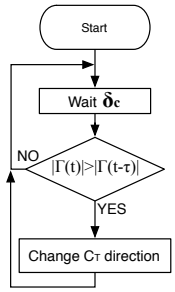


Figure 7: Iterative minimum search algorithm flow chart.

the system is approaching to the matching point there is a smooth transition between the coarse controller and the fine tune controller. The fine tune controller is a dual SISO proportional controller (Fig.9) in which C_L is driven by $Im[\Gamma]$ and C_T is driven by $Re[\Gamma]$.

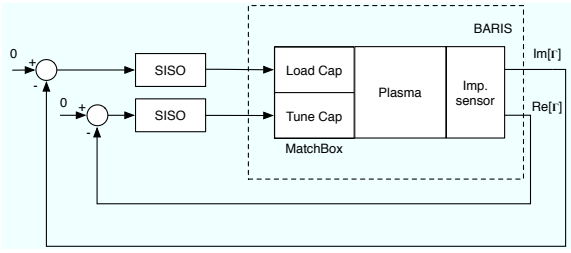


Figure 9: Block schematic of the fine tune controller.

4.1 Implementation

Fig.8 illustrates the Simulink implementation of the controller. The AD converter provides measurements

of G and $\Delta\theta$ at a sampling period of $\delta_f = 20mS$, which is also the sampling rate used by the fine tune controller. The coarse controller uses a sampling period of $\delta_c = 100mS$. $|\Gamma|$ is determined via eq (4) by the fine tune controller, allowing the delayed value of $|\Gamma(t-\tau)|$ to be available at the δ_c sampling instants ($\tau = 4\delta_f$).

$$\Gamma = \frac{Z_L - Z_0}{Z_L + Z_0} \tag{4}$$

The coarse controller checks the variation of the reflection coefficient $|\Gamma(t)| - |\Gamma(t-\tau)|$. If it is increasing, the J-K flip-flop inverts the direction of movement of C_T. Since the speed of C_T is proportional to $|\Gamma|$, the controller can be considered as a proportional controller. The gains of the fine tune SISO controllers are multiplied by $(1-|\Gamma|)^2$ in order to reduce its effect in the non-convergence region, in which it tends to drive the capacitors in the wrong direction. In this way, when approaching to impedance matching condition, $|\Gamma|$ is decreasing and $(1-|\Gamma|)^2$ is increasing, and there is a smooth transition from the coarse to the fine tune controller.

5 RESULTS

The controller has been tested both in simulation and in the BARIS chamber. The simulation has been performed using constant loads, and giving different starting positions for the capacitors. Figs.10 and 11 show that the system converges both when the initial conditions are close and far away to the matching point, that is the controller drives the capacitors in the right direction in order to minimize $|\Gamma|$. Fig.12

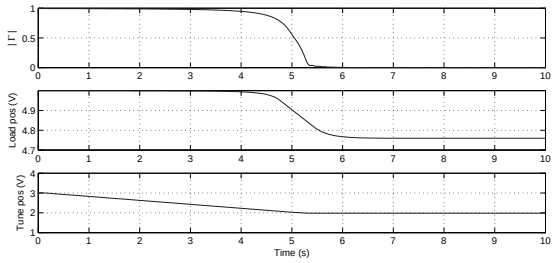


Figure 10: Simulation results with starting conditions close to the matching point.

shows the behavior of the system when the fine tune controller's gains are not multiplied by $(1-|\Gamma|)^2$; for a starting condition far away from the matching point there is no convergence. In this case the coarse controller can't take over and the fine tune controller drives the capacitors in the wrong direction.

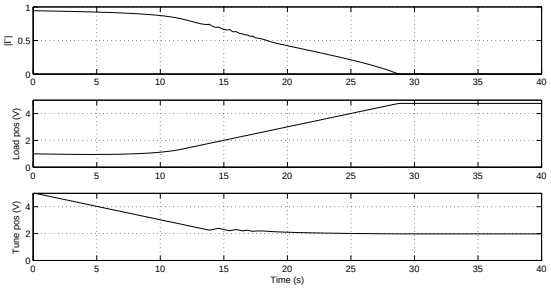


Figure 11: Hierarchal controller with starting conditions far away from the matching point.

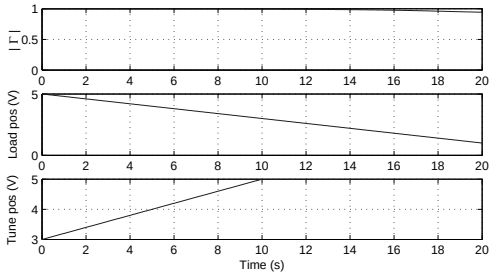


Figure 12: Hierarchical controller, with no fine tune controller gain attenuation.

The controller has been tested also in the BARIS system; this test has been performed using step functions for the plasma variables (RF power, pressure, A_r and O_2 flows). From the results of this test (Fig.13) it is possible to see that each time the plasma state changes, the controller tunes the matching network, minimizing the magnitude of the reflection coefficient $|\Gamma|$. In particular, mark A denotes a step in RF power, mark B denotes a step in the gate valve position, mark C denotes a step in O_2 flow and mark D denotes a step in A_r flow.

6 CONCLUSION

The hierarchical controller shows good performances regarding the convergence. Besides, it is computationally not demanding, giving the possibility to be implemented using a simple micro-controller. A multivariable controller, which observes the dependance of $Im[Z_{PL}]$ on both C_L and C_T was also designed, but requires an extensive look-up table and matrix inversion, which is in stark contrast to the attractive simplicity of the final controller presented above. The underlying principle of the controller is based only on the matching network structure, therefore it can be implemented also in other applications using a similar "L-type" matching network.

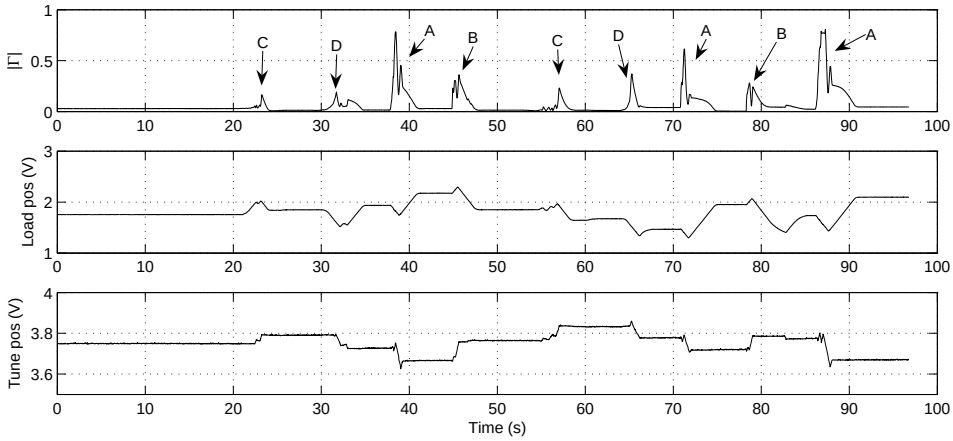


Figure 13: Experimental measurements in the BARIS.

ACKNOWLEDGEMENTS

The authors are grateful for the financial support of the Irish Research Council for Science Engineering and Technology (IRCSET) and INTEL Ireland Ltd.

REFERENCES

Cottee, C.J.; Duncan, S. (2003). Design of matching circuit controllers for radio-frequency heating. *IEEE Transactions on Control Systems Technology*, 11(1):91–100.

De Mingo, J., Valdovinos, A., Crespo, A., Navarro, D., and Garcia, P. (2004). An RF electronically controlled impedance tuning network design and its application to an antenna input impedance automatic matching system. *IEEE Transactions on Microwave Theory and Techniques*, 52(2):489–497.

Ida, I., Takada, J., Toda, T., and Oishi, Y. (2004a). An adaptive impedance matching system and considerations for a better performance. *IEEE 5th International Symposium on Multi-Dimensional Mobile Communications Proceedings*, 2:563–567.

Ida, I., Takada, J., Toda, T., and Oishi, Y. (2004b). An adaptive impedance matching system and its application to mobile antennas. *IEEE Region 10 Conference*, 3:543–546.

Ida, I., Takada, J., Toda, T., and Oishi, Y. (2004c). An adaptive impedance matching system for mobile communication antennas. *IEEE Antennas and Propagation Society International Symposium*, 3:3203–3206.

Iordanov, P., Keville, B., Ringwood, J., and Doherty, S. (2006). On the closed-loop control of an argon plasma process,. *Irish Signals and Systems Conference*.

Keville, B., Iordanov, P., Ringwood, J., Doherty, S., Faulkner, R., Soberon, F., and McCarter, A. (2006). On the modeling and closed loop control of an inductively coupled plasma chamber. *IFAC Workshop on Advanced Process Control for Semiconductor Manufacturing*.

Mazza, N. (1970). Automatic impedance matching system for RF sputtering. *IBM Journal of Research and Development*, 14(2).

Moritz, J. and Sun, Y. (2001). Frequency agile antenna tuning and matching. *IEE Eighth International Conference on HF Radio Systems and Techniques*, 148:177–182.

Parro, V. and Pait, F. (2003). Design of an automatic impedance matching system for industrial continuous microwave ovens. *Sociedade Brasileira de Microondas Optoeletronica/IEEE Microwave and Optoelectronics Conference*, 2:20–23.

Sun, Y. and J.K., F. (1997). Component values ranges of tunable impedance matching networks in rf communications systems. *IEE HF Radio Systems and Techniques*, 441.

Sun, Y. and J.K., F. (1999). Antenna impedance matching using genetic algorithms. *IEE National Conference on Antennas and Propagation*, 441:31–36.

Thompson, M. and Fidler, J. (2000). Application of the genetic algorithm and simulated annealing to LC filter tuning. *IEE Circuits, Devices and Systems*, 474(2):169–174.

Vai, M. and Prasad, S. (1993). Automatic impedance matching with a neural network. *IEEE Microwave and Guided Wave Letters*, 3(10).

RUN-TIME RECONFIGURABLE SOLUTIONS FOR ADAPTIVE CONTROL APPLICATIONS

George Economakos
μLab, School of ECE, NTU of Athens, Iroon Polytexneiou 9, GR 15780 Athens, Greece
geconom@microlab.ntua

Christoforos Economakos
Department of Automation, Halkis Institute of Technology, GR 34400 Psachna, Evia, Greece

Sotirios Xydis
μLab, School of ECE, NTU of Athens, Iroon Polytexneiou 9, GR 15780 Athens, Greece

Keywords: Hardware, Adaptive control, Run-time systems.

Abstract: The requirement for short time-to-market has made FPGA devices very popular for the implementation of general purpose electronic devices. Modern FPGA architectures offer the advantage of partial reconfiguration, which allows an algorithm to be partially mapped into a small and fixed FPGA device that can be reconfigured at run time, as the mapped application changes its requirements. Such a feature can be beneficial for modern control applications, that may require the change of coefficients, models and control laws with respect to external conditions. This paper presents an embedded run-time reconfigurable architecture and the corresponding design methodologies that support flexibility, modularity and abstract system specification for high performance adaptive control applications. Through experimental results it is shown that this architecture is both technically advanced and cost effective so, it can be used in increasingly demanding application areas like automotive control.

1 INTRODUCTION

During the last years, consumer digital devices have been built using either application specific hardware modules (ASICs) or general purpose software programmed microprocessors, or a combination of them. Hardware implementations offer high speed and efficiency but they are tailored for a specific set of computations. If an alternative implementation is needed, a new and expensive design process has to be performed. On the contrary, software implementations can be modified freely during the life-cycle of a device, through patches and updates. However, they are much more inefficient in terms of speed and area.

Reconfigurable computing is intended to fill the gap between hardware and software, achieving potentially much higher performance than software, while maintaining a higher level of flexibility than hardware. Reconfigurable devices, including *Field-Programmable Gate Arrays* (FPGAs), contain an array of computational elements whose functionality is determined through multiple programmable configuration bits. These elements, usually called logic blocks, are connected using a set of routing resources that are also programmable. In this way, custom digital circuits can be mapped to the reconfigurable hardware by computing the logic functions of the circuit within the logic blocks, and using the configurable routing to connect the blocks together to form the necessary circuit. Currently, the most common configuration technique is to use *Look-Up Tables* (LUTs), implemented with *Random Access Memory* (RAM). A survey of reconfigurable devices and the underlying technologies can be found in (Hartenstein, 2001).

Frequently, the areas of a program that can be accelerated through the use of reconfigurable hardware are too numerous or complex to be loaded simultaneously onto the available hardware. There, it is beneficial to be able to swap different configurations in and out of the reconfigurable hardware as they are needed during program execution. This concept is known as *Run-Time Reconfiguration* (RTR). RTR supports the concept of *Virtual Hardware*, like the concept of virtual memory offered by all modern operating systems. Through RTR, more sections of an application can be mapped into hardware and thus, despite reconfiguration time overhead, a potential for an overall performance improvement is provided. RTR can be applied

on different phases of the design process, according to the granularity of the reconfigurable blocks, which may be complex functions, simple arithmetic and storage components or LUTs. The reconfiguration data can be stored inside the reconfigurable device or transfered from an embedded or host processor.

RTR FPGAs can be used in demanding applications like modern adaptive control found in the automotive industry, where a clear trend prevails today: electronics in the vehicle are gaining more and more significance (Javaherian et al., 2004). The number of microcontrollers in the automobile is consistently increasing. For example, luxury vehicles may have up to 100 on-board microcontroller units in the near future. All this functionality involve a lot of computations that can be accelerated with embedded special purpose hardware. On the other hand, applications like speed control need to provide solutions to a variety of problems like smooth throttle movement, zero steady-state speed error, good speed tracking over varying road slopes, robustness to system variations and operating conditions and minimum controller calibrations. To achieve all these, an adaptive controller may need to change coefficients, models and control laws during its everyday operation which involve a lot of reconfiguration.

This paper presents an embedded RTR architecture for control applications. It is based on a modern family of FPGA devices (Xilinx Virtex 4 (Xilinx, 2006)) that offer many advanced reconfiguration options. It consists of a general purpose microprocessor (PowerPC), built inside the FPGA device, and a number of reconfigurable modules. Reconfiguration is done by the microprocessor through an internal configuration port and using configuration data stored in on-chip block RAM (BRAM). All reconfigurable modules are small size and thus, reconfiguration time overhead is minimal. This paper also presents the corresponding design methodologies that support flexibility, modularity and abstract system specification. Through experimental results it is shown that this architecture is both technically advanced and cost effective so, it can be used in increasingly demanding application areas like automotive control.

2 FPGA ARCHITECTURE

FPGAs are the evolution of PLAs and PLDs. They contain pre-build programmable circuit elements and programmable interconnects that can realize any digital system with low cost and reduced time-to-market. The weak points of programmable logic are efficiency

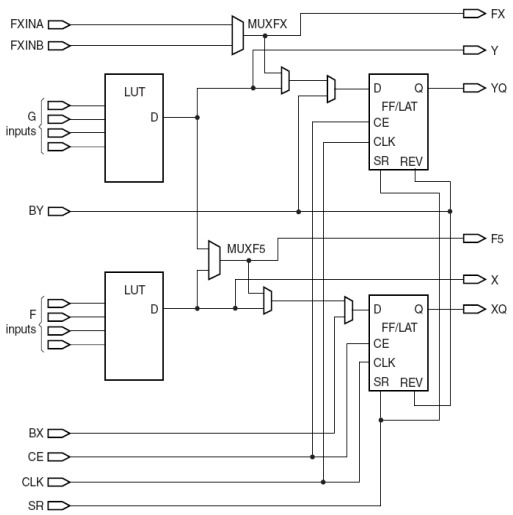


Figure 1: Simplified circuit of 2 CLB slices.

and performance but this is starting to change. A typical FPGA device consists of programmable logic blocks, interconnection resources and I/O blocks, arranged in an array structure.

For the devices built by Xilinx (Xilinx, 2006) the programmable logic blocks are called *Configurable Logic Blocks* (CLBs) and are divided into four slices. Two are more powerful (called SLICEM) and two less (called SLICEL). A simplified circuit of SLICEM is shown in figure 1. Each slice, either in SLICEM or SLICEL, consists of two logic-function generators or *Look-Up Tables* (LUTs), two storage elements, wide-function multiplexers, carry logic, and arithmetic gates. The extra power of SLICEM is that it can be configured to support two additional functions: storing data using distributed RAM and shifting data with 16-bit registers.

LUT function generators are implemented as 4-input RAM. There are four independent inputs for each of the two function generators in a slice (F and G). The function generators are capable of implementing any arbitrarily defined four-input Boolean function. The propagation delay through a LUT is independent of the function implemented. In addition to the basic LUTs, slices contain multiplexers that can be used to combine up to eight function generators to provide any function of five, six, seven, or eight inputs in a CLB.

The other elements of the CLB may vary from device to device. Dedicated carry logic provides fast arithmetic addition and subtraction. The Xilinx Virtex-4 CLB has two separate carry chains. The storage elements in a each slice can be configured as ei-

ther edge-triggered D-type flip-flops or level-sensitive latches. The D input can be driven directly by a LUT output or by the slice inputs bypassing the function generators, using multiplexers. Finally, the dedicated arithmetic logic includes an XOR gate that allows a 2-bit full adder to be implemented within a slice and an AND gate to improve the efficiency of multiplier implementation.

The interconnection resources, called *General Routing Matrix* (GRM), provides an array of configurable routing switches, called *Programmable Switch Matrices* (PSMs), between each component. Each CLB is tied to a PSM, allowing multiple connections. The overall programmable interconnection is hierarchical and designed to support high-speed designs. PSMs are controlled by values stored in static memory cells during configuration and can be reloaded to change the functions of the programmable elements.

I/O blocks can be configured as inputs, outputs or bidirectional and are connected to the GRM and to the chip pads. They have configurable high-performance drivers and receivers,supporting a wide variety of standard interfaces.

FPGAs are programmed by writing a bitstream in the configuration memory (all configuration bits and LUT contents). The bitstream is usually externally supplied through a serial link. For RTR, when an application requires a change configuration memory while the device is operational, the Xilinx Virtex-4 architecture defines the global *Internal Configuration Access Port* (ICAP), which provides the user logic with access to the configuration interface.

3 RELATED RESEARCH

Real-time embedded control is an important application area for microelectronic devices. With the introduction and wide distribution of FPGA devices a lot of efficient hardware controller implementation have been reported (Kim, 2000; Sanchez-Solano et al., 2002; Chan et al., 2004; Tipsuwanporn et al., 2004; Zhao et al., 2005). For more advanced control algorithms and systems, reconfigurable solutions have also been reported. An embedded reconfigurable architecture is presented in (Sancho-Pradel et al., 2002), with a number of processing elements with real-time reconfigurable software. The main processing element computes adaptive control coefficients in real-time and passes them to the control processing element, which changes its software controller implementation accordingly. A more advanced multi-agent architecture is presented in (Naji et al., 2004), which supports hardware reconfiguration but not real-time. In (Toscher et al., 2006) a real-time hardware reconfigurable controller is presented. It has a number of slots where reconfigurable modules are loaded in and out as needed. This approach is similar to the one presented here but involves large (coarse grain) reconfigurable modules and so reconfiguration overhead plays an important rope in the overall system performance.

4 DESIGN METHODOLOGY

This paper considers RTR for adaptive control applications. For small applications like a PID controller, minor modifications are required during system operation. If an adaptive algorithm is used to generate new coefficient values an update can replace the old values in a straightforward manner (details will be given in a subsequent section). When however complicated models or control laws are considered the corresponding hardware design methodology has to be changed. The solution proposed in this paper is to take the adaptive control algorithm of the whole system and apply *Algorithmic* or *High-Level Synthesis* (HLS) (Gajski et al., 1992) taking into account RTR.

HLS acts upon the dataflow graph of an application and schedules its primitive operations in consecutive control steps while mapping them onto available resources. The proposed solution is a novel resource constrained scheduling heuristic that utilizes RTR arithmetic units. After experimentation with different FPGA architectures, it has been found that a binary multiplier takes 3 to 4 times the LUTs required for an adder of the same input bit width. So, we can assume that we have an arithmetic component that can be used as a multiplier in some control steps and as 3 adders (at least) in all the others. If we perform resource constrained scheduling with such reconfigurable components we can reduce the latency, in terms of control steps, of our circuit.

For example, consider a digital filter with two inputs x and y and two outputs z_1 and z_2, where $z_1 = a_0x_0 + x_1 + x_2 + a_3x_3 + x_4 + a_5x_5$ and $z_2 = b_0y_0 + b_1y_1 + y_2 + y_3 + b_4y_4 + y_5$. If we want to build a circuit for this system, using two multipliers and one adder in every control step, we will come out with the schedule of figure 2. If one of the multipliers is reconfigurable, and as stated in the previous paragraph can be used as either a multiplier or 3 adders, we can reduce the latency drastically, as shown in figure 3.

Such a result is promising taking into account that RTR needs some time for reconfiguration at the beginning of some of the control steps. To formalize our approach we can modify a widely used HLS scheduling heuristic to support RTR datapath compo-

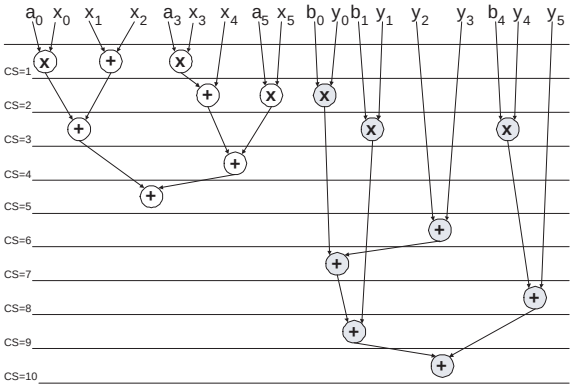


Figure 2: Schedule with 2 mult. and 1 add.

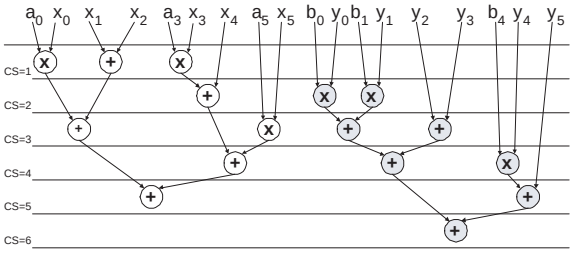


Figure 3: Schedule with 1 mult., 1 add. and 1 RTR mult.

nents. For resource constrained scheduling, that is, when the number of available hardware resources is fixed, a very efficient and widely used algorithm is list scheduling. A modified version of list scheduling, utilizing RTR components is shown below.

INSERT_READY_OPS(V,$PList_{t_1}$,$PList_{t_2}$,...,$PList_{t_m}$);
Cstep=0;
while (($PList_{t_1} \neq \emptyset$) or ... or ($PList_{t_1} \neq \emptyset$)) **do**
  Cstep=Cstep+1;
  for k=1 to m **do**
    for funit=1 to N_k **do**
      if ($PList_{t_k} \neq \emptyset$) **then**
        $S_{current}$=SCH_OP($S_{current}$,FIRST($PList_{t_k}$,Cstep));
        $PList_{t_k}$=DELETE($PList_{t_k}$,FIRST($PList_{t_k}$);
      endif
    endfor
  endfor
  {$RPList_{t_1}$,...,$RPList_{t_{Rn}}$}=MERGE($PList_{t_1}$,...,$PList_{t_m}$);
  for k=1 to Rn **do**
    if ($RPList_{t_k} \neq \emptyset$) **then**
      $S_{current}$=SCH_OPS($S_{current}$,NTH($RPList_{t_k}$,Cstep));
    endif
  endfor
  INSERT_READY_OPS(V,$PList_{t_1}$,$PList_{t_2}$,...,$PList_{t_m}$);
endwhile

The algorithm uses a priority list *PList* for each operation type $t_k \in T$. Each operation's priority is defined by its *mobility*, that is the difference between its ALAP and its ASAP scheduling value. The operations in all priority lists are scheduled into control steps based on N_k which is the number of functional units performing operation of type t_k. The function INSERT_READY_OPS scans the set of nodes V, determines if any of the operations in the set are ready (i.e., all its predecessors are scheduled), deletes each ready node from the set V and appends it to one of the priority lists based on its operation type. The function SC_OP($S_{current}$,o_i,s_j) returns a new schedule after scheduling the operation o_i in control step s_j. The function DELETE($PList_{t_k}$,o_i) deletes the indicated operation o_i from the specified list. Operations with low mobility are put first in the list. In other words, operations that do not have many opportunities to be scheduled in subsequent control steps are preferred for the current. As the algorithm moves on mobilities are dynamically re-calculated. After all available non-reconfigurable components have been used the algorithm constructs a set of merged priority lists {$RPList_{t_1}$,...,$PList_{t_{Rn}}$} for each control step with the function MERGE. Each merged list contains ready operations that a reconfigurable component can perform. Then, the function SCH_OPS, schedules all operations of the same type that are in the beginning of the merged list and cover the whole reconfigurable component (or as much as possible). These operations are returned by the function NTH. For example, if we have a reconfigurable component that can perform one multiplication or three additions and the merged priority list is {a_1,a_2,m_1,a_3,m_2} (where a_i denotes addition and m_i multiplication), a_1, a_2 and a_3 will be scheduled in the current control step.

The circuits designed using this heuristic are faster but have a reconfiguration timing overhead. Depending on the implementation technology different approaches can be taken to make the final implementation efficient. In architectures with very small reconfiguration time (10ns) we can extend the duration of every control cycle. In slower architectures we can restrict the number of possible reconfigurations so as the total reconfiguration delay is less than the speed gain. Additionally, in all cases, the proposed reconfiguration can be kept minimum by utilizing very few (less than five) reconfigurable components.

5 EXPERIMENTAL RESULTS

The scheduling algorithm of the previous section has been implemented on top of a custom C-to-RTL HLS synthesis environment. In order to evaluate the proposed methodology, six different DSP applications have been used as testbenches. These applications

Table 1: DSP schedules with RTR.

Application	Number of nodes	Number of cycles		
		3/3	2/1/2	1/1/2
Fircls	63	24	18	10
Firls	64	32	25	17
Firrcos	79	42	30	18
Invfreqz	41	25	18	10
Maxflat	115	51	38	22
Remez	55	28	20	17

were found in MATLAB's DSP tool box and were manually translated into untimed C (in fact SystemC) behavioral Descriptions. The applications were Fircls (Constrained least square FIR filter), Firls (Least square linear-phase FIR filter), Firrcos (Raised cosine FIR filter), Invfreqz (Discrete-time filter from frequency data) Maxflat (Generalized digital Butterworth filter) and Remez (Parks-McClellan optimal FIR filter). Table 1 shows three implementations for each application, one with 3 multipliers, 3 adders and no reconfigurable components, one with 2 regular multipliers, 1 reconfigurable multiplier and 2 adders and one with 1 regular multiplier, 1 reconfigurable multiplier and 2 adders. The implementations with only 1 regular multiplier have an average latency improvement of 53% and also occupy less area. In other words, under this approach a much better resource utilization is achieved. The penalty that has to be paid is that if reconfigurations are very frequent (for example at the beginning of every control step) the total reconfiguration delay may be too long. The 53% latency improvement however covers even a doubling in control step period (worst case) due to RTR.

As a more complicated example taken from car automation we chose the detection component of the cruise control system of (Le Beux et al., 2006). This component compares a reference and a returned radar signal and reports when an obstacle is found within the next 150 m. In such situation the cruise control system should decelerated the vehicle. Comparison is performed with a 3 stage correlation algorithm. Each correlation requires more than 100 multiplications. Following the same approach as the DSP experiments above we found that the whole algorithm has 472 dataflow nodes which can be arranged into 207 control steps when 4 multipliers and 4 adders are used, 153 when 3 multipliers, 3 adders and 1 reconfigurable multiplier is used and 98 when 2 multipliers, 2 adders and 2 reconfigurable multipliers are used. Again the latency improvement is enough to overcome reconfiguration delays. Practical details about the latter are given in the next section.

6 IMPLEMENTATION ISSUES

While the proposed algorithm is focused on future architectures with low RTR overhead, some implementation issues may be solved in an efficient way with present and widely spread FPGA devices. Such an issue is that if we want to have really fast reconfiguration all action must be performed inside the reconfigurable fabric, because any external source of reconfiguration data (like serial connection with a host computer) is too slow. An answer for that problem is the Virtex family of Xilinx FPGAs, which is equipped with an internal reconfiguration access port (ICAP) used by internal logic to access and modify the configuration memory. Xilinx offers a ready-to-use IP called HWICAP (Xilinx, 2004), which can read a portion of the configuration memory into block RAM, modify it, and write it back, through the ICAP port. HWICAP can be used in embedded self-reconfigurable devices (Blodget et al., 2003; Ferreira and Silva, 2005).

The proposed architecture is given in figure 4. The HWICAP controller can be connected with an embedded processor like PowerPC through the OPB bus (or any bus and an OPB bridge). The processor communicates with the HWICAP controller through the bus and requests that a part of the devices configuration memory is written in on-chip RAM (block RAM). Then the processor can modify this information (accessing directly block RAM) and request to be written back. So the processor, which is initially configured inside the FPGA, can reconfigure other parts of the device during run time. To do this the processor needs to know how to modify the copy of configuration memory to achieve the required results. In our approach, the differences between the multiplier and the three adders can be initially stored inside PowerPC (during the initial configuration phase) and exchanged on demand with appropriate interrupt service routines. If the differences are kept as small as possible, this is both feasible and efficient.

This approach is called difference-based reconfiguration and allows fast reconfiguration of Virtex-4 devices (Xilinx, 2006) at a rate of 400MB/s. The smallest partial bitstream that the HWICAP device can handle is a frame of 32 vertical slices (each slice contains 2 LUTs) which is 41 32bit words.

For our experiments we found that a 16 bit multiplier needs 54 slices while each 16 bit adder 9. In order to minimize the reconfiguration overhead, we used placement constraints to arrange the 3 adders (27 slices) of the reconfigurable multiplier in a common frame. In the beginning, this frame along with a number of neighboring slices is configured as a 16

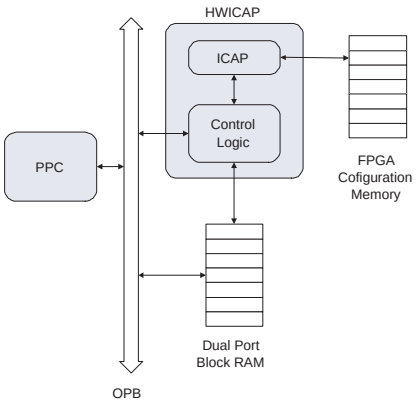


Figure 4: Implementation architecture.

bit multiplier. When reconfiguration is needed a hardware FSM generates an interrupt to PowerPC which sends through HWICAP the frame with the 3 adders. Also care is taken so that the reconfigurable component has ports for all devices (both the multiplier and the 3 adders) permanently connected to the registers and MUXs of the overall architecture. From all these details the reconfiguration time for each reconfigurable component can be calculated as 0.41μsec.

7 CONCLUSIONS

A novel design methodology for adaptive control applications, which utilizes reconfigurable datapath components has been presented in this work. Using reconfigurable multipliers, the resulting schedule can be shortened so as the gain in clock cycles can overcome the timing overhead of reconfiguration. The main advantage of this solution is that through RTR, more complicated algorithms can be mapped into smaller devices without speed degradation. The experimental results after integrating the proposed heuristic into an HLS environment shown an average 50% reduction in clock cycles that compensates for the worst cases of reconfiguration overhead, with better hardware utilization. Since RTR delays will be shortened even more in future devices, the proposed scheduling heuristic may be proved to be even more effective.

REFERENCES

Blodget, B., McMillan, S., and Lysaght, P. (2003). A lightweight approach for embedded reconfiguration of FPGAs. In *Design Automation and Test in Europe Conference and Exhibition*, pages 399–400. ACM/IEEE.

Chan, Y. F., Moallem, M., and Wang, W. (2004). Efficient implementation of PID control algorithm using FPGA technology. In *43rd Conference on Decision and Control*, pages 4885–4890. IEEE.

Ferreira, J. C. and Silva, M. M. (2005). Run-time reconfiguration support for FPGAs with embedded CPUs: The hardware layer. In *International Parallel and Distributed Processing Symposium*, pages 165–168. IEEE.

Gajski, D., Dutt, N., Wu, A., and Lin, S. (1992). *High-Level Synthesis*. Kluwer Academic Publishers.

Hartenstein, R. (2001). A decade of reconfigurable computing: A visionary retrospective. In *Design Automation and Test in Europe Conference and Exhibition*, pages 642–649. ACM/IEEE.

Javaherian, H., Liu, D., Zhang, Y., and Kovalenko, O. (2004). Adaptive critic learning techniques for automotive engine control. In *American Control Conference*, pages 4066–4071. IEEE.

Kim, D. (2000). An implementation of fuzzy logic controller on the reconfigurable FPGA system. *IEEE Transactions on Industrial Electronics*, 47(3):703–715.

Le Beux, S., Marquet, P., Labbani, Q., and Dekeyser, J. (2006). FPGA implementation of embedded cruise control and anti-collision radar. In *9th Conference on Digital System Design*, pages 280–287. EUROMICRO.

Naji, H. R., Wells, B. E., and Etzkorn, L. (2004). Creating an adaptive embedded system by applying multi-agent techniques to reconfigurable hardware. *Future Generation Computer Systems*, 20(6):1055–1081.

Sanchez-Solano, S., Senhadji, R., Cabrera, A., Baturone, I., Jimenez, C. J., and Barriga, A. (2002). Prototyping of fuzzy logic-based controllers using standard FPGA development boards. In *13th International Workshop on Rapid System Prototyping*, pages 25–32. IEEE.

Sancho-Pradel, D. L., Jones, S. R., and Goodall, R. M. (2002). System on programmable chip for real-time control implementations. In *International Conference on Field-Programmable Technology*, pages 276–283. IEEE.

Tipsuwanporn, V., Runghimmawan, T., Intajag, S., and Krongratana, V. (2004). Fuzzy logic PID controller based on FPGA for process control. In *International Symposium on Industrial Electronics*, pages 1495–1500. IEEE.

Toscher, S., Reinemann, T., and Kasper, R. (2006). An adaptive FPGA-based mechatronic control system supporting partial reconfiguration of controller functionalities. In *1st NASA/ESA Conference on Adaptive Hardware and Systems*, pages 225–228. IEEE.

Xilinx (2004). *OPB HWICAP Product Specification v1.3*.

Xilinx (2006). *Virtex-4 User Guide*.

Zhao, W., Kim, B. H., Larson, A. C., and Voyles, R. M. (2005). FPGA implementation of closed-loop control system for small-scale robot. In *12th International Conference on Advanced Robotics*, pages 70–77. IEEE.

EFFICIENT IMPLEMENTATION OF FAULT-TOLERANT DATA STRUCTURES IN PC-BASED CONTROL SOFTWARE

Michael Short
Embedded Systems Laboratory, University of Leicester,University Road, Leicester,UK
mjs61@le.ac.uk

Keywords: Open architecture controllers, software fault tolerance, critical systems.

Abstract: Recent years have seen an increased interest in the use of open-architecture, PC-based controllers for robotic and mechatronic systems. Although such systems give increased flexibility and performance at low unit cost, the use of commercial processors and memory devices can be problematic from a safety perspective as they lack many of the built-in integrity testing features that are typical of more specialised equipment. Previous research has shown that the rate of undetected corruptions in industrial PC memory devices is large enough to be of concern in systems where the correct functioning of equipment is vital. In this paper the mechanisms that may lead to such corruptions and the level of risk is examined. A simple, portable and highly effective software library is also presented in this paper that can reduce the impact of such memory errors. The effectiveness of the library is verified in a small example.

1 INTRODUCTION

Recent years have seen much interest in the use of open-architecture controllers for robotic and mechatronic systems. Such systems typically consist of a combination of commercial off the shelf (COTS) PC equipment alongside motion control, network interface, and sensor/actuator equipment – e.g. (Hong et al. 2001; Short 2003; Lee & Mavroidis 2000; Schofield & Wright, 1998). Such architectures have been used to successfully implement novel control algorithms in a number of research installations (e.g. fuzzy force control (Burn et al. 2003) and H_∞ vibration control (Lee & Mavroidis 2000)); they are also being used in increasing numbers in industrial applications (e.g. KUKA®, STAUBLI®, RWT® systems). The flexibility of such platforms primarily arises by giving engineers the ability to develop and/or modify the control and interfacing hardware and software, which is typically developed in C++.

However, despite increased flexibility and performance (along with marked unit cost reductions), such COTS equipment lacks many of the built-in integrity testing elements which are often employed in more proprietary, specialised control equipment. Many robotic and mechatronic systems, by virtue of their design, are somewhat critical in nature.

A study of available data by Dhillon and Fashandi (1997) concluded that robot-related accidents are primarily caused by unexpected or unplanned motions of the manipulator; a contributory cause of which was malfunctions of the robot control system. Unexpected motions of a manipulator or tooling may result in damage (or complete destruction) of the system itself and any (potentially expensive) equipment in the systems' workspace. Additionally, when considering applications such as surgical robotics, any unexpected or unplanned motion could also result in serious injury or death.

When such COTS equipment is used in situations where their correct functioning is vital, special care must be therefore taken to ensure that the system is both reliable and safe (Storey 1996; Levenson 1995). When considering the equipment employed in a typical robot control system, attention must be paid to potential permanent, transient or intermittent failures of the hardware and software. The need for fault-tolerant techniques is dependant on the potential risk, which is primarily dependant on the application and environment the system is employed in.

Much research has concentrated on providing hardware fault tolerance for such systems (e.g. see Storey 1996). Recent years have also seen the development of several software-based approaches to implementing transient fault detection on COTS

processors. They are based around instruction counting, instruction/task duplication and control flow checking (e.g. Rajabzadeh & Miremadi 2006; Rebaudengo et al. 2002; Oh et al. 2000).

Although such techniques are effective at detecting many control flow errors, systems which incorporate them may still be vulnerable to transient errors in data memory (which may not result in control-flow errors). This paper is particularly concerned with the mitigation of transient errors in COTS memory devices used in open-architecture controllers. In section 2 of the paper, we will consider the mechanisms that may lead to memory corruption, the resulting effects, and the level of risk.

In section 3, a simple yet highly effective software library that reduces the impact of memory corruptions and overcomes these implementation difficulties is presented and described in detail. In section 4 we apply this library to a simple test program; a 6 x 6 matrix multiplication program. Fault injection results are described for both the un-hardened and hardened programs. Section 5 concludes the paper.

2 MEMORY ERRORS

2.1 Mechanisms

Corruption of data in memory devices can come from a variety of sources. Single event effects - (SEE's) - caused by particle strikes, may manifest themselves in a variety of ways. They may cause transient disturbances known as single event upsets (SEU's), manifested as random bit-flips in memory. They may also cause permanent stuck-at faults over an array of memory, caused by damage to the read/write circuitry or chip latchup.

In addition, memory devices may also fail due to normal electrical and thermal breakdown effects. Such electrical or thermal failures and disturbances in memory devices may be highly unpredictable, manifesting themselves as complete device failures or stuck-at faults over part (or all) of the memory array.

Memory devices are also susceptible to electromagnetic interference (EMI) from a variety of sources. For example in an industrial robot workcell, numerous devices such as electromechanical relays, motor drives and welding equipment are all sources of noise that are capable of corrupting many electronic circuits (Ong & Pont 2002). Other mechanisms that may lead to memory upsets include power supply fluctuations and radio frequency interference (RFI).

2.2 Level Of Risk

Failure rates for SEU's in ground-based installations are in the region of 10^{-10} - 10^{-12} failures per bit per hour (Normand 1996). Failure rates for individual devices due to electrical effects may be calculated using a methodology such as (MIL 1991); they are typically in the region of 10^{-6} failures per device per hour. Predicting the effects of EMI, RFI and power supply disturbances are extremely difficult and highly dependant on the operating environment and the hardware mitigation techniques that are employed (e.g. signal shielding).

From a practical perspective, experimental studies have demonstrated that on COTS memory devices with built-in integrity checks (such as parity and error correction codes) the problem of undetected memory corruption is large enough to be of concern for some critical systems. Additionally, much PC hardware does not even support such integrity checks (Messer et al. 2001).

For example, a 4MB DRAM memory chip is likely to encounter 6000 undetected memory failures in 10^9 hours of operation (Messer et al. 2001). If a control system PC employs several such devices, with a total of 512 MB memory, this translates to an undetected memory corruption approximately every 55 days of operation.

2.3 Activation Effects

Obviously, not all memory errors will become activated. However, robotic control systems typically involve extremely data-intensive processing with hard real-time constraints. Techniques such as co-ordinate transforms, kinematics, resolved-motion rate control, path planning and force control all typically require hundreds (perhaps even thousands) of matrix manipulations and feedback control calculations every second (e.g. Fu et al. 1997). Considering that a simple 6 x 6 matrix multiplication and storing of the result typically involves 864 memory read/write operations, it can be argued that the probability of activating an error in such control software is relatively high when compared to (for example) a word processing application.

If a memory error does become activated, this can lead to a variety of unpredictable faults (Ong & Pont 2002). For example, they may cause an incorrect value to be output to a port or peripheral;

or they may cause a further area of memory to be corrupted by indexing an array out of its normal bounds. In an open-architecture controller, all of these faults can potentially escalate to full system failures, and cause unpredictable motions of the manipulator or tooling.

2.4 Mitigation Techniques

In order to address this vulnerability, some researchers have investigated the use of Single-Program Multiple Data (SPMD) techniques for data redundancy in both single and multi processor systems (e.g. Redaudengo et al. 2002; Gong et al. 1997). However, such approaches can be problematic from the point of view of the control system developer. Primarily because when the techniques are actually applied, the complexity of the resulting source code can increase dramatically, and the basic meaning of the code can become obscured. This may have an impact on code development, testing and subsequent code maintenance. To illustrate this point, consider the segment of C code shown in Figure 1.

```
01:   #define N (10)
02:   int i;
03:   int a[N],b[N];
04:    for(i=0;i<N;i++)
05:      {
06:      b[i]=a[i];
07:      }
```

Figure 1: Un hardened code.

For most programmers, this is "self documenting" code, and the meaning is clear (the programmer wishes to copy the contents of any array of ten integers to another array of the same size). Now, consider the same code, hardened using the technique suggested by Redaudengo et al. (2002). This is shown in Figure 2 (note the required checksum initialization code and the XOR macro CHK have been omitted for space reasons). The total code segment, including this initialization (which must be called before each operation), and the CHK macro, is in excess of 36 lines in length; the meaning of the code is also somewhat obscured. In addition, the variable ***i*** in Figure 2 remains un-hardened. If the variable ***i*** were to be hardened, the meaning of the code would become further obscured, with the check-and-correct code for ***i*** embedded within the ***for*** loop construct; as more nested variables are hardened, the problem can soon become difficult to manage. This can be particularly troublesome when writing matrix manipulation code which can often require many levels of nesting.

```
01:   #define N (10)
02:   int i;
03:   int a0[N],b0[N];
04    int b1[N],b1[N];
05    int c0,c1;
06:   for (i=0;i<N;i++)
07:      {
08:      c0=c0^b0;
09:      c1=c1^b1;
10:      b0[i]=a0[i];
11:      b1[i]=a1[i];
12:      c0=c0^b0;
13:      c1=c1^b1;
14:      if(a0[i]!=a1[i])
15:         {
16:         if(CHK(a0,b0)==C0)
17:            {
18:            a1[i]=a0[i];
19:            c1=c0;
20:            }
21:         else
22:            {
23:            a0[i]=a1[i];
24:            c0=c1;
25:            }
26:         }
27:      }
```

Figure 2: Hardened code.

Although this problem may be overcome by the use of automatic code generators, this adds an extra level of complexity and abstraction to the software development process, and adds a real possibility of introducing systematic errors into the design process.

In the following section, a software-based methodology will be proposed to simplify the implementation of data redundancy. This technique is an implementation of an SPMD-like architecture to provide fault tolerance to transient errors in data memory. This approach directly addresses the problems of code complexity and compatibility with other software-based approaches. It is primarily suited to C++, but can easily be ported to other object-oriented languages (e.g. JAVA).

3 THE NEW APPROACH

3.1 Requirements

The requirement for this software library was to provide a portable and highly flexible set of new data types for use with C++ programs. The new data types should encapsulate a Triple Modular Redundant (TMR) approach which is completely

hidden from the programmer. The data types, to all intents and purposes, appear to the programmer as their basic simplex counterparts; and all the new data types can also be used interchangeably with their simplex counterparts. The library should be as non-intrusive as possible and not require the use of an automatic code generator for its implementation.

Every write operation to the new types invokes a write to three duplicated variables of the corresponding type, and each read operation invokes a two-from-three vote on the duplicated variables. This concept is shown for a generic data type in Figure 3.

We assume that if the data is so corrupted that a two-from-three vote cannot be achieved, then a user-defined error handler is called. This required functionality of this error hander is highly application dependant; it could, for example, freeze the mechanical system and execute a software based self test (SBST) algorithm to verify that no permanent hardware failures have occurred in the CPU peripherals or RAM. Hamdioui et al. (2002) and Sosnowski (2006) have proposed efficient SBST algorithms to achieve this.

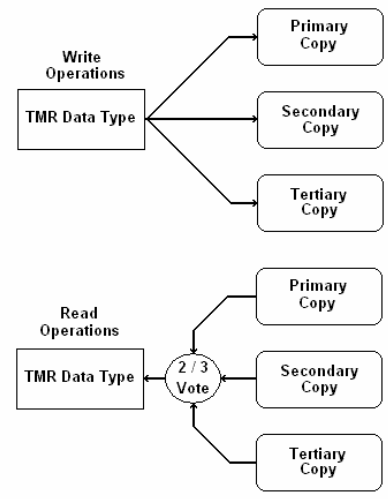


Figure 3: Generic TMR data concept.

3.2 C++ Implementation

In order to create a generic and flexible implementation, the required TMR behaviour was defined in a generic C++ class template named ***TMR_datatype***. The prototype of the class template is shown in Figure 4. This class template for a given data type ***T*** can then be applied to any of the basic in-built C++ data types by means of suitable ***#define*** statements, also shown in Figure 4.

```
01:   template <class T>
02:   class TMR_datatype
03:   {
04:   public:
05:   inline TMR_datatype(const T);
06:   inline TMR_datatype(void);
07:   inline T operator =(const T);
08:   inline operator T();
09:   inline T operator+=(const T);
10:   inline T operator-=(const T);
11:   inline T operator*=(const T);
12:   inline T operator/=(const T);
13:   inline T operator++(int);
14:   inline T operator++(void);
15:   inline T operator--(int);
16:   inline T operator--(void);
17:   inline T operator &=(const T);
18:   inline T operator |=(const T);
19:   inline T operator ^=(const T);
20:   private:
21:   T Primary_Copy;
22:   T Secondary_Copy;
23:   T Tertiary_Copy;
24:   };
25:   #define TMR_int TMR_datatype
      <int>
26:   #define TMR_float TMR_datatype
      <float>
```

Figure 4: TMR_Datatype class template.

From the class template, it can be seen that each derived object of the template contains three private data declarations, ***Primary_Data***, ***Secondary_Data*** and ***Tertiary_Data***, corresponding to the simplex data type ***T***. The required read and write operations on this data are then achieved by defining new operator member functions using the ***operator*** keyword. It can be seen that all of these operator functions are expanded inline by the compiler, with the use of the ***inline*** keyword; this is to reduce any overheads associated with the call of a member function.

By way of example, the member functions for both the assignment and reference operations on the class template are shown in Figure 5. Note that the use of the explicit reference operator is used in the implementation; thus only the operator functions that explicitly modify the data contents (such as =, ++, --, +=, and so on) needed to be overloaded; this creates a very efficient and portable implementation.

```
01:   inline T TMR_data::operator
      =(const T Value)
02:   {
03:   Primary_Copy=Value;
04:   Secondary_Copy=Value;
05:   Tertiary_Copy=Value;
06:   return(Value);
07:   }
08:   inline TMR_data::operator T()
09:   {
10:   if(Primary_Copy==Secondary_Copy)
```

```
11:   {
12:   Return(Primary_Copy);
13:   }
14:   else
      if(Primary_Copy==Tertiary_Copy)
15:   {
16:   Return(Primary_Copy);
17:   }
18:   else
      if(Secondary_Copy==Tertiary_Copy)
19:   {
20:   Return(Secondary_Copy);
21:   }
22:   else
23:   {
24:   Error();
25:   }
26:   }
```

Figure 5: Assignment and reference member functions.

From Figure 5, it can be seen that the TMR behaviour has been captured by the template; when an assignment (write) operator is encountered, the value is written to the three copies of the data. When the reference (read) operator is encountered, a two-from-three vote is employed and the data returned. If no vote is possible, the user defined function ***Error*** is called.

3.3 Hardening Procedure

In Figure 6, the code library described in this section is applied to the code example shown in Figure 1. From Figure 6 it can be seen that the length of the hardened source code is identical to the original and is also highly readable. Additionally, it is noted that – unlike the code shown in Figure 2 - the variable ***i*** is also hardened in this case.

```
01:   #define N (10)
02:   TMR_int i;
03:   TMR_int a[N],b[N];
04:   for (i=0;i<N;i++)
05:      {
06:      b[i]=a[i];
07:      }
```

Figure 6: Hardened code.

From these descriptions it can be seen that this library does not require the use of automatic code generators for its implementation: all that is required is for the programmer to have a basic understanding of the new data types. The hardening procedure can be accomplished extremely rapidly; all that is required is the inclusion of the new TMR template into a project, and altering the variable declarations that require hardening to their redundant counterparts.

4 EXPERIMENTAL RESULTS

To assess the effectiveness of the proposed code library, a fault-injection study was performed on a Intel® Pentium 4-based PC, with a CPU speed of 2.6 GHz and 512 MB RAM, running the Windows NT® operating system. A simple (yet representative) application program was created to perform a 6x6 floating point matrix multiplication. During each experiment, transient faults were injected into the program data area at random times, performing random single bit-flips in the used data areas. The fault injection was performed using a secondary application running on the PC. In the program, the source matrices are first initialized with known constant values. The matrix multiplication is then performed. The values contained in the result matrix are then compared with known constant results. The process then repeats endlessly. Any failures or corrected errors are logged by the application program.

Two different implementations of the program were considered; the normal (simplex) case, and the hardened TMR version. To asses the impact of applying the library on execution time, we also measured the iteration time for each loop of each program using the Pentium performance counter. Table 1 shows the recorded results. In the hardened program, the number of faults injected was increased to reflect the increased size of the program data areas. Fault effects were classified into one of three categories, as follows:

- Effect-less: the fault does not result in a computation failure.
- Corrected: the fault is detected and has been corrected.
- Failure: the fault is not detected or corrected and results in an invalid computation output.

From these results, it can be seen that for both cases, the error activation level was approximately 77%. Application of the TMR data structures increased the execution time of the multiplication task by a factor of 3.2; this is to be expected as we have introduced instruction duplication and voting. Additionally it should be noted that each hardened variable increases the overall memory usage due to its triplicate implementation. The increase in overall program code size was 7.1% in this case.

Table 1: Fault injection results for each program.

	Normal	Hardened
Injected	10000	30000
No Effect	2240	6690
Failures	7760	0
Corrected	0	23010
Calc. Time (us)	4.72	15.27

We can also see from the results that of the 77% of activated faults, 100% of these caused computation failures in the normal program case. The hardened case however, detected and corrected 100% of the activated faults.

5 CONCLUSIONS

In this paper, the mechanisms that can lead to memory corruption in COTS PC control devices have been considered. A novel approach to software implemented fault-tolerance has been presented. The approach, based on an SPMD architecture, can be used to compliment existing error detection and SBST techniques for COTS processors used in open architecture controllers. The approach relies on data and instruction duplication. It has been shown that the method is easily applied, results in readable code, and is able to tolerate 100% of the injected faults in the benchmark described. Whilst the application of the techniques provides high levels of data fault tolerance, there is obviously a trade-off with increases in the code and data size and task execution time. Prospective designers must obviously take these factors into account when considering the techniques.

ACKNOWLEDGEMENTS

The work described in this paper was supported by the Leverhulme Trust (Grant F/00 212/D).

REFERENCES

Burn, K., Short, M., Bicker, R., 2003. Adaptive And Nonlinear Force Control Techniques Applied to Robots Operating in Uncertain Environments. *Journal of Robotic Systems*, Vol. 20, No. 7, pp. 391-400.

Dhillon, B.S., Fashandi, A.R.M., 1997. Safety and reliability assessment techniques in robotics. *Robotica*, Vol. 15, pp. 701-708.

Fu, K.S., Gonzales, R.C., Lee, C.S.G., 1987. *Robotics: Control, Sensing, Vision And Intelligence*. McGraw-Hill International Editions.

Gong, C., Melhem, R., Gupta, R., 1997. On-line error detection through data duplication in distributed memory systems. *Microprocessors and Microsystems,* Vol. 21, pp. 197-209.

Hamdioui, S., van der Goor, A., Rogers, M., 2002. March SS: A Test for All Static Simple RAM Faults. In *Proc. Of the 2002 IEEE Intl. Workshop on Memory Tech., Design and Testing*.

Hong, K.S., Choi, K.H., Kim, J.G., Lee, S., 2001. A PC-based open robot control system: PC-ORC. *Robotics and ComputerIntegrated Manufacturing*, Vol. 17, pp. 355-365.

Lee, C.J., Mavroidis, C., 2000. WinRec V.1: Real-Time Control Software for Windows NT and its Applications. In *Proc. American Control Conf.*, Chicago, Il., pp. 651-655.

Levenson, N.G., 1995. *Safeware: System Safety and Computers*, Reading, M.A., Addison-Wesley.

Messer, A., Bernadat, P., Fu, G., Chen, G., Dimitrijevic, Z., Lie, D., Mannaru, D.D, Riska, A., Milojicic, D., 2001. Susceptibility of Modern Systems and Software to Soft Errors, In *Proc. Int. Conf. on Dependable Sys. And Networks*, Goteburg, Sweden.

MIL-HDBK-217F, 1991. *Military Handbook of Reliability Prediction of Electronic Equipment*. December 1991.

Normand, E., 1996. Single Event Effects in Avionics, *IEEE Trans. on Nuclear Science*, Vol. 43, No. 2.

Oh, N., Shivani, P.P., McCluskey, E.J., 2001. Control Flow Checking by Software Signature. *IEEE Trans. On Reliability*, September 2001.

Ong, H.L.R, Pont, M.J., 2002. The impact of instruction pointer corruption on program flow: a computational modelling study. *Microprocessors and Microsystems*, 25: 409-419.

Rajabzadeh, A., Miremadi, S.G., 2006. Transient detection in COTS processors using software approach, *Microelectronics Reliability*, Vol. 46, pp. 124-133.

Rebaudengo, M., Sonza Reorda, M., Violante, M., 2002. A new approach to software-implemented fault tolerance. In *Proc. IEEE Latin American Test Workshop*, 2002.

Schofield, S., Wright, P., 1998. Open Architecture Controllers for Machine Tools, Part 1: Design Principles. *Trans. ASME Journ. of Manufacturing Sci. & Engineer*, Vol. 120, Pt. 2, pp. 417-424.

Short, M., 2003. *A Generic Controller Architecture for Advanced and Intelligent Robots*. PhD. Thesis, University of Sunderland, UK.

Sosnowski, J., 2006. Software-based self-testing of microprocessors. *Journal of Systems Architecture*, Vol. 52, pp. 257-271.

Storey, N., 1996. *Safety Critical Computer Systems*. Addison Wesley Publishing.

DESIGN AND IMPLEMENTATION OF A MONITORING SYSTEM USING GRAFCET

Adib Allahham and Hassane Alla
GIPSA-Lab, Department of Control, Institute National de Polytechnique de Grenoble
961 Rue de la Houille Blanche - Domaine universitaire BP 46, 38402 Saint Martin D'hères, France
adib.al-lahham@inpg.fr, hassane.alla@inpg.fr

Keywords: Monitoring, Fault detection, Manufacturing systems, Stopwatch automata, Reachability analysis, Grafcet.

Abstract: A monitoring system based on a stopwatch automaton is proposed to detect the system faults as early as possible. Each location in the automaton corresponds to a system's situation. Its time space delimits exactly the range of the normal behavior in the corresponding system's situation. The monitoring system detects a fault when the time space corresponding to the actual system's situation is violated. The stopwatch automaton provides a formal foundation to model the system's behavior and to synthesize the exactly time space in each location. This paper aims to provide the grafcet monitor that allows to link the design of the monitoring system of a system with its implementation in a programmable logic controller.

1 INTRODUCTION

Monitoring complex manufacturing systems plays an important role for economic and security reasons. A wide variety of methods has been considered this problem. These methods consider a fault have occurred in a system if a faulty event occurs (Ghazel et al., 2005), reaching a faulty state (S. H. ZAd and Wonham, 2003) or more generally violating system specifications. Most systems monitor the timed system specifications by using Watchdogs. They detect a fault if the expected observation is produced early or late with respect to certain time bounds.
The increasingly stringent requirements in monitoring and fault detection problems lead to the necessity to detect the fault as early as possible without waiting the expiration of certain bounds. For that, we have proposed in (A.allahham and alla, 2006) a monitoring method which extends the method of residuals, well-known in continuous system. In (A.allahham and alla, 2006), we have introduced the notion of acceptable behavior of a system detailed in the following section. We model this acceptable behavior by a stopwatch automaton. In that representation, each location corresponds to a state of the system and the arcs are labeled by switching conditions between the different states. In each state, the differential equations express the progression or suspension of the task represented by the stopwatch due to a fault. The time sub-space in each location represented by a set of algebraic inequalities, delimits the range of stopwatches in the corresponding system's situation in the acceptable behavior. The monitoring system detects a fault when the system exceeds this time sub-space.
The stopwatch automaton provides a formal basis to model the system's behavior and to analyze it in order to characterize the exact time sub-space in each location, corresponding to the acceptable behavior.
In this paper, our objective is to provide the grafcet model that allows to link the design of monitoring system of a manufacturing system with its implementation in the logic controller. We show that the grafcet fulfils not only the sequential specification of the applications but also the continuous behavior specified in the monitoring stopwatch automaton.
The grafcet corresponding to monitoring automaton models a location by a step and a stopwatch by a timer where the following problem is encountered. The behavior of a stopwatch goes beyond the ability of a timer representing the simplest way to include the time in grafcet model. This problem in turn affects the method to represent the time sub-space associating to the steps of grafcet. However, we will show that this problem can be overcome by complet-

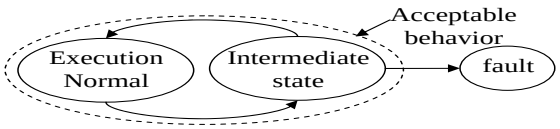


Figure 1: Acceptable behavior of a system.

ing the grafcet by actions associated with steps. Also, the grafcet will monitor permanently the consistency of the stopwatches within its acceptable range.

Section 2 describes the acceptable behavior of a system and its model based on stopwatch automaton. Our approach is given and used to delimit the time space characterizing this behavior. In Section 3, the method to translate a monitoring automaton into a Grafcet model is detailed. We apply this method in an illustrative example in Section 4.

2 THE ACCEPTABLE BEHAVIOR

The possible kinds of faults that affect the resources in a manufacturing system are the permanent faults, which dispossess a resource's ability to perform its task and the intermitting faults. These faults can appear several times during the task execution and disappear without any external action on the system while permanent faults disappear due to a repair of the fault (Huang et al., 1996). Our work considers only the intermitting faults that interrupt the task of a resource. We call it *malfunctions* and the task subjected to these malfunctions as *interruptible task*. The system containing these tasks is called as *interruptible system*. Because of malfunctions, an intermediate state can appear between a normal state and a faulty one. In this state, the system can come back to the normal behavior or it leaves toward a faulty state (Fig.1). We refer to this behavior by *acceptable behavior*. These malfunctions occur often in a manufacturing system, so the system's designer accepts to some extent this behavior for productivity motives. The question to answer is: how the designer takes into account these malfunctions in his system.

Let be a task $Task_i \in Task_{int}$ where $Task_{int}$ represents the set of interruptible tasks in a complex system S. $Task_i$ has a known execution duration $[\alpha_i, \beta_i]$ which is given in the technical characteristics of the resources that execute $Task_i$ or measured directly. Because of the interruptions resulting from malfunctions, the designer accepts a tolerated duration to execute $Task_i$. It is given by the interval $[\alpha_i, \gamma_i)$ where $\beta_i < \gamma_i$. We call $[\alpha_i, \beta_i]$ and $[\alpha_i, \gamma_i)$ respectively the normal and acceptable durations of $Task_i$.

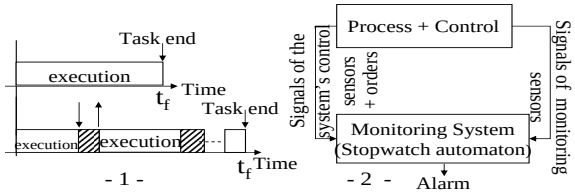


Figure 2: 1- Behavior of an interruptible task 2- Inputs\Output of monitoring system.

2.1 Monitoring of an Interruptible Task

We refer to the apparition and disappearing of a fault by its effect on the task execution, then we refer it by *interruption* and *resuming* of the task.

Hypothsis 1 *The execution speed is supposed to be constant or to vary sightly around a mean value.* □

Considering the properties of the tasks mentioned above, we distinguish the behavior of an interruptible task shown in Figure 2.1. Either $Task_i$ is executed without interruption, then $t_f \in [\alpha_i, \beta_i]$ or $Task_i$ has been executed but with several interruptions. After each interruption, the system resumes from the position at which it has been interrupted. In this case: $t_f \in [\alpha_i, \gamma_i)$.

To monitor $Task_i$, we use the timers x_i and y_i. The timers x_i and y_i have a values "0" when the task begins. x_i will be used to check that $Task_i$ has completed before the expiration of its tolerated deadline. y_i is used to monitor the effective time of execution. Then, $Task_i$ is correctly executed if $y_i \in [\alpha_i, \beta_i]$ and $x_i \in [\alpha_i, \gamma_i)$ when the task end occurs.

The arrows $\downarrow$ and $\uparrow$ in Figure 2.1 represent respectively the signal of logical sensor which detects the interruption and resuming of $Task_i$. These signals represent an input of our monitoring system (Fig. 2.2).

2.2 Modeling of an Interruptible System

We use the stopwatch automata *SWA* to model the interruptible system. It is a class of linear hybrid automaton where the time derivative of a clock in a location can be either 0 or 1 (Cassez and Larsen, 2000).

Definition 1 *A stopwatch automaton is a 7-tuple* $(L, l_0, X, \Sigma, A, I, \dot{X})$ *where:*

- L *is a finite set of locations,* l_0*: the initial location,*
- X *is a finite set of stopwatches,*
- Σ *is a finite set of labels,*
- A *is a finite set of arcs.* $a = (l, \delta, \sigma, R, l') \in A$ *is the arc between the locations* l *and* l'*, with the guard* $\delta \in C(X)$*, the label name* σ *and the set of stopwatches to reset* R*.* $C(X)$ *is the set of constraints over* X*.*

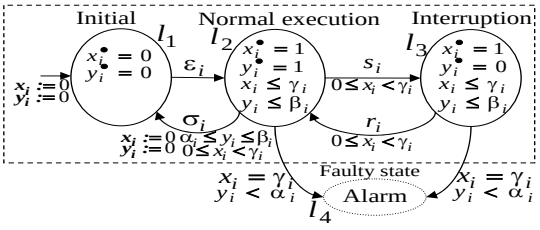


Figure 3: Stopwatch automaton of an interruptible task.

- $I \in C(X)^L$ *maps an invariant to each location,*
- $\dot{X} \in (\{0,1\}^X)^L$ *maps an activity to each location.* □

- ***SWA of an interruptible task***

We model the acceptable behavior of $Task_i$ by the Stopwatch automaton shown in Fig. 3. The location l_1 indicates that the resource is waiting to start the task, l_2 that the resource is executing its task and l_3 that the task is interrupted after having started. In this automaton, the clock y_i in l_3 does not progress while x_i evolves to express that the task is interrupted but the time remains progressing. The labels s_i and r_i represent respectively the stop and the resumption of $Task_i$ in the physical system, while label σ_i corresponds to the end of this task. ε_i which is the always true event, represents the necessary condition to start the task. Here it starts immediately.

The guard g_2 of the arc $l_2 \xrightarrow{g_2} l_3$ expresses that the interruption can occur at any instant during the acceptable duration while the guard g_3 associated to $l_3 \xrightarrow{g_3} l_2$ expresses that the resumption must occur before exceeding the acceptable duration. The execution of $task_i$, during its acceptable duration is represented by the guard g_4 of the arc $l_2 \xrightarrow{g_4} l_1$.

Figure 3 shows that $Task_i$ leaves the acceptable behavior to faulty state l_4 either from the location l_2 or l_3. The guards of arcs towards l_4 are identical and given by $g_5 = \neg g_4 = (x_i = \gamma_i \wedge y_i < \alpha_i)$. It expresses the fact that the acceptable duration of execution was expired and $Task_i$ is not executed.

2.3 Time Space State Delimiting the Acceptable Behavior

The acceptable behavior of a system S is represented by a stopwatch automaton $\mathbb{A}$. It is obtained by the composition of the different tasks automata according to the system specifications which represent the relation between these tasks.

Property 1 *The trajectories which lead $Task_i$ to the state $l_1 \times (0,0)$ from $l_2 \times (x_i, y_i)$ where $x_i \in [\alpha_i, \gamma_i)$ and $y_i \in [\alpha_i, \beta_i]$, represent all the possible evolutions characterizing the execution of $Task_i$.* □

The trajectories specified in Property 1 represent only a part of the possible ones. Thus, the synthesis problem of monitoring can be set as follows: given a stopwatch automaton $\mathbb{A}$ representing a system S, restrict the possible trajectories of this automaton in a way that all remaining ones satisfy Property 1, for all the tasks of S. As a result, we obtain an automaton $\mathbb{A}^*$ where all its trajectories characterize the acceptable execution of S. The calculation of the time space containing these trajectories E^* of $\mathbb{A}^*$ is the core of our synthesis algorithm. This is realized using of the Forward and backward reachability analysis. (Alur et al., 1995)

- ***Forward analysis of monitoring SWA:***

We use the forward analysis operators to calculate all the possible trajectories in the system. In other words: the reachable time space E in the automaton $\mathbb{A}$ mentioned above. The forward operators look for all the reachable states of a stopwatch automaton from its initial state remaining in the locations of automaton while the time progresses or by firing its transitions. The reachable time space by forward analysis in locations l_2 and l_3 of the automaton shown in Figure 3 is given in Figure 4.1. Note that the values of the stopwatches given by g_4 in Figure 3 define a polyhedron. We denote it as D_i, and call it as *the desired space* of $Task_i$ (Fig 4.2). Note also that the trajectories specified in Property 1 lead the task only to D_i. These trajectories represent only a part of the ones which are contained in reachable time space (Fig. 4.1). Thus,we must delimit the time space containing only these trajectories to characterize the acceptable execution.

- ***Backward analysis of monitoring SWA:***

It is not hard to see that the time space E^* of $\mathbb{A}^*$ can be obtained by removing from the time space of $\mathbb{A}$ the states from which system's evolutions do not lead to D_i of each interruptible task. In other words, one needs first to apply the backward operators (called as predecessors and annotated as *Pre* operators) to the guards of arcs representing the desired space of all the tasks over the automaton $\mathbb{A}$. Then, $E^* = E \cap (\bigcup Pre(D_i))$. The intuition behind the using the predecessors operators for a guard representing D_i of $Task_i$ is that we look for all the states that lead to this space D_i from the initial state of $\mathbb{A}$.

Applying the backward analysis for the automaton given in Figure 3 gives the time space shown in Figure 4.3. The intersection of this space and that of forward analysis is given in Figure 4.4. It is the space characterizing the execution acceptable of $Task_i$. One of the trajectories contained in synthesized space (Fig. 4.4) shows that the task reaches a faulty state, only from the location l_3 with the dynamics $\dot{x} = 1$ and $\dot{y} = 0$. Figure 4.5 presents the final monitoring automaton $\mathbb{A}^*$.

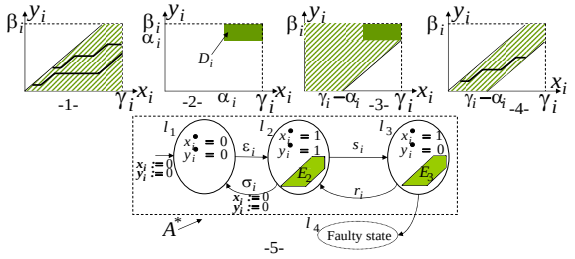


Figure 4: Time space in l_2 and l_3: (1) reachable by forward analysis, (2) desired, (3) reachable by backward analysis (4) delimiting acceptable execution (5) Synthesized Monitoring automaton of an interruptible task.

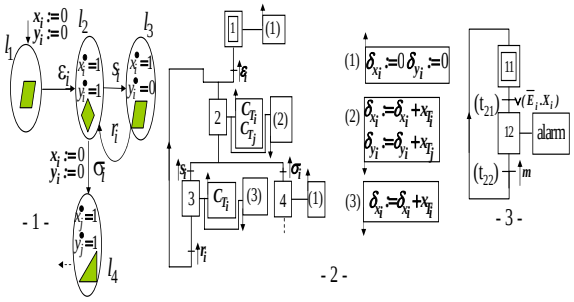


Figure 6: (1) A part of monitoring automaton (2) G_1 and shifting and initiation actions (3) G_2 model.

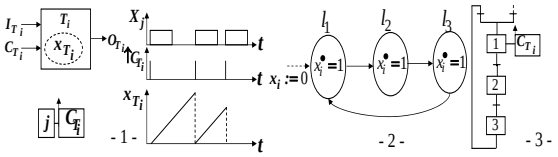


Figure 5: (1) Timer (2) A part of monitoring automaton (3) Corresponding grafcet G_1.

3 GRAFCET OF THE MONITORING SYSTEM

Grafcet and its international standard SFC (CEI/ IEC 60848 revised in 2002) are used for the implementation of discrete events models for manufacturing systems and many programmable logic controllers use it as a programming language. The basic concepts of the grafcet are: the step, action, transition and its associated receptivity (David, 1995). A Boolean variable X_i is associated with each step. Its value is 1 when step is active.

The general idea to translate the monitoring automaton $\mathbb{A}^*$ into a grafcet is to represent each location of the automaton by a step. The faulty state is also modeled by a step. Let $L = \{l_1, ..., l_n\}$ be the set of locations of $\mathbb{A}^*$. The set of steps corresponding to these locations is denoted by $\{1, ..., n\}$. An arc linking two locations is modeled by a transition linking the two corresponding steps. The transition receptivity is the label of the arc. The simplest way to include time in the grafcet model is to use timer objects, for that, each stopwatch will be modeled by a timer.

Figure 5 shows a timer (T_i) which is typically initialized with a value representing a duration (I_{T_i} input) and a control input (C_{T_i}) for starting the timer. This timer produces a boolean output (O_{T_i}). Associating an impulse action $\uparrow C_{T_i}$ with a step j will activate the timer T_i as soon as $\uparrow X_j = 1$. Here, we are not interested in the logic output of timer, but in the instantaneous value of the timer T_i denoted by x_{T_i}, which is supposed to be readable and testable in real time. In fact, many *PLC* manufacturers provides products with timers equipped with functions permitting to read and test the value x_{T_i}.

In these translation rules, the behavior of a stopwatch goes beyond the ability of a timer. To show that, we consider the part of monitoring automaton shown in Figure 5.2. In this automaton the stopwatch x_i is newly activate in l_1 and remains active in l_2 and l_3. Translating this model into a grafcet by using the method described above, gives the model shown in Figure 5.3 where T_i is the timer corresponding to stopwatch x_i. In this grafcet, we activate the timer T_i as soon as the $\uparrow X_1 = 1$. T_i remains active in steps 2 and 3. However this is not sufficient to represent the behavior of the monitoring automaton since an important issue is the behavior at the firing the arc of automaton between l_3 and l_1. The stopwatch x_i persists active after the commutation and has a certain value at the instant of reaching l_1, while there will be an initialization of the value of corresponding timer T_i when $\uparrow X_1 = 1$ in the grafcet. However, we show that this problem can be overcome by completing the grafcet by actions and by using intermediate variables.

• ***Modeling of stopwatches by timers:***

Let us consider that the automaton given in Figure 6.1 follows the behavior given in Figure 7. T_i and T_j are the timers corresponding to stopwatches x_i and y_i. We express the dynamics $\dot{x}_i = 1$ and $\dot{y}_i = 1$ in the location l_2 by associating to step 2 the impulse actions $\uparrow C_{T_i}$ and $\uparrow C_{T_j}$. These actions will activate T_i and T_j as soon as $\uparrow X_2 = 1$. In a similar way, we express the dynamic $\dot{x}_i = 1$ in l_3. We will now give the method to represent the behavior of x_i and y_i whose values are 0 at the entry of l_2. Note that the value of x_i in a given location l_2 or l_3 is the sum of: the value of x_i when the system reaches this location and the passed time from the reaching instant to actual one.

The latter item corresponds to the value of timer T_i which is activated when the system reaches the step corresponding to the given location. For the for-

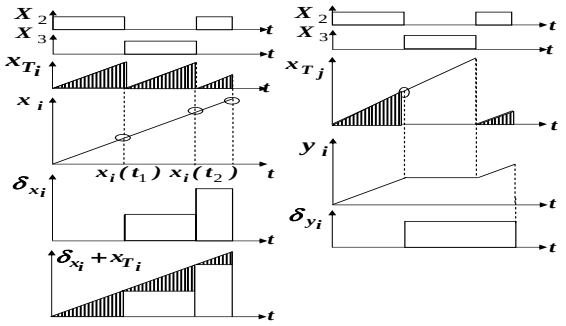


Figure 7: Representing behavior of stopwatches in grafcet.

mer item, an intermediate variable denoted by δ_{x_i} and called as *shifting variable* is used. δ_{x_i} is initialized when the automaton resets to 0 the stopwatch x_i. The value of δ_{s_i} corresponding to $x_i(t_1)$ in Figure 7 can be obtained by associating to the step 2 (Fig. 6.2) the impulse action $\downarrow \delta_{x_i} := \delta_{x_i} + x_{T_i}$ (Shifting action). It adds to δ_{s_i} whose initially has the value 0, the value of x_{T_i} representing the duration that the grafcet stays in step 2. The value of δ_{x_i} corresponding to $x_i(t_2)$ in Figure 7 can be obtained by associating to step 3 the same action. It adds to previous value of δ_{x_i} the duration that the system rests in step 3. The resulting values of δ_{x_i} are shown in Figure 7. They correspond to that of x_i at the instants of reaching l_2 and l_3 after each commutation between these two locations. As a result, $\delta_{x_i} + x_{T_i}$ is equivalent to that of x_i at any instant during the system dynamics either in l_2 or l_3.

The behavior of stopwatch y_i is different from that of x_i. y_i is suspended when the automaton fires from l_2 to l_3. y_i resumes in location l_2 from the same value when it was suspended, then we associate the action $\downarrow \delta_{y_i} := \delta_{y_i} + x_{T_j}$ to step 2 to memorize this value. δ_{y_i} is initialized when the automaton resets to 0 the stopwatch y_i. The describing exactly the given part of automaton is given in Figure 6.2.

In Figure 6.1, x_i and y_i are initialized by firing the arc $l_2 \rightarrow l_4$. Our grafcet does this resetting by allocating to zero the variables δ_{x_i} and δ_{y_i} after the firing from step 2 to 4. The action resetting the shifting variables will be associated to the step 4. The initial step of is associated by an impulse action resetting all the shifting variables used in the grafcet.

The grafcet monitor checks permanently the time space associated to the actual step. The faulty step is reached when the system violates this time range. This fact can be represented in the grafcet model by using the concept of hierarchy. It is easy to imagine that a grafcet G_1 has an influence on anther grafcet G_2. G_1 is the Grafcet resulting from structural translation described above (Fig. 6.2). G_2 has two steps: initial and faulty steps (Fig. 6.3). The activation of initial step of G_2 expresses that the system's behavior is acceptable. G_2 evolves to faulty step when the time space is violated. Let E_1,..E_i,..,E_n be the time subspace in the locations l_1,..,l_i,..,l_n permitting to evolve to the faulty state. The corresponding steps in grafcet G_1 are 1,..,*i*,..,*n*. The receptivity of t_{21} in Figure 6.3 is: $[X_1.\overline{E_1} + .. + X_i.\overline{E_i} + .. + X_n.\overline{E_n}]$. In Figure 6.3, the event $\uparrow m$ represents the reparation operation.

4 APPLICATION

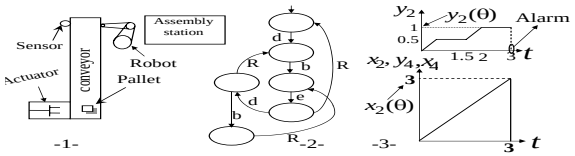


Figure 8: -1- Workshop -2- Working specification -3- A scenario of working.

Figure 8 shows a manufacturing system and its working specification. In this system, when the control system gives the order d, the actuator puts down a pallet on the conveyor. When the sensor B detects the transferred pallet (*event b*), and if the robot is not busy (*event e*), it transfers the pallet to the assembly station. The actuator comes back to its initial state and waits again d. When the robot finishes its task (*event R*), it returns to its initial state. The information concerning the interruptible tasks is given in the following table. $t.u$ is the abbreviation for *"time units"*.

Task name	Conveyor task	Robot task
$[\alpha_i,\beta_i]$ (t.u)	[3,4]	[2,3]
$[\alpha_i,\gamma_i)$ (t.u)	[3,5)	[2,4)
Used stopwatches	x_2 and y_2	x_4 and y_4
Monitoring signals	s_2 and r_2	s_4 and r_4

In Figure 9.1, we give the monitoring automaton of the considered system composed of 12 locations and focalize to a part of it in Figure 9.2. The time spaces in the locations have been calculated by using the model-checker *PHAVer* (Frehse, 2005).

Figure 8.3 shows a scenario of working where the robot and conveyor start their tasks simultaneously. This situation is represented by location L_7 as the stopwatches dynamic's show. In this scenario, the conveyor is interrupted 2 $t.u$. Then, the system fires to L_8. The inequality in bold in L_8 detects a fault in the considered behavior at the instant $x_2(\theta) = 3$. The corresponding value of y_2 is $y_2(\theta) = 1$. This result can be explained as follows: to finish the conveyor task correctly, one needs to have at least the duration $\alpha_2 - y_2(\theta) = 3 - 1 = 2\ t.u$. The corresponding value of x_2 will be $x_2 = x_2(\theta) + (\alpha_2 - y_2(\theta)) = 3 + 2 = 5$. This value exceeds the maximum permitted duration of conveyor's task. Figure 10.1 shows the monitoring

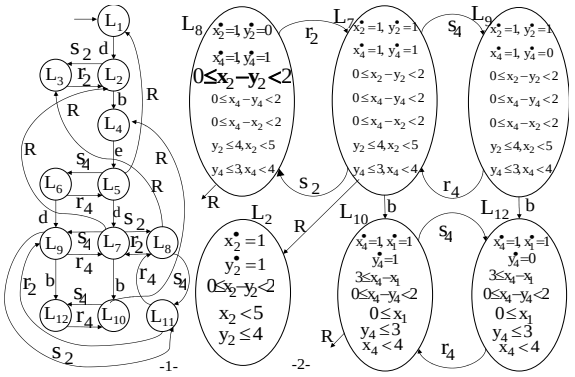


Figure 9: 1- Automaton $\mathbb{A}^*$ 2- Scoped part of $\mathbb{A}^*$.

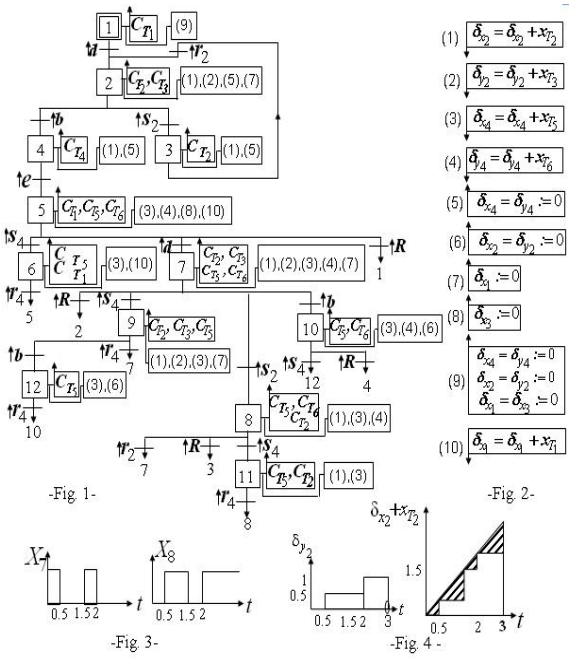


Figure 10: 1- G_1 2- Shifting and initiation actions 3- G_1 evolutions 4- evolution of Grafcet variables.

grafcet G_1 of the system. The timers T_1, T_2, T_3, T_5 and T_6 correspond respectively to stopwatches x_1, x_2, y_2, x_4 and y_4. The used shifting variables are : δ_{x_1}, δ_{x_2}, δ_{y_2}, δ_{x_4}, and δ_{y_4}. Figure 10.3 shows the evolution of G_1 according to the proposed scenario.
The receptivity of transition t_{21} in G_2 (Fig. 6.3) is: $(X_3.\overline{E_3}+X_6.\overline{E_6}+X_8.\overline{E_8}+X_9.\overline{E_9}+X_{11}.\overline{E_{11}}+X_{12}.\overline{E_{12}})$. Its predicate becomes true at the instant $t=3$ because $X_8=1$ and the inequality $(\delta_{x_2}+x_{T_2})-\delta_{y_2}\geq 2$ in $\overline{E_8}$ becomes true at this instant as shown in Figure 10.4.

5 CONCLUSION

Active approach has been carried out to provide solution to specific problem related to the fault detection which is the ability to detect the faults as early as possible. It is based on a stopwatch automaton which provides a formal support to this approach. The link between the design of monitoring system and its implementation in programmable logic controller is provided using grafcet tool. We have shown how the grafcet can be used to describe the monitoring stopwatch automaton's behavior.

REFERENCES

A.allahham and alla, H. (2006). Monitoring of timed discrete events systems: Application to manufacturing systems. In *The 32nd Annual conference of IEEE Industrial Electronics Society*.

Alur, R., Courcoubetis, C., Halbwachs, N., Henzingerd, T., Hod, P., Nicollin, X., Olivero, A., Sifakis, J., and Yovine, S. (1995). The algoritmic analysis of hybrid systems. *Theoretical Computer Science*, 138(1).

Cassez, F. and Larsen, K. (2000). The impressive power of stopwatch. In *11th conference on concurrency theory*, number 1877, pages 138–152.

David, R. (1995). Grafcet: A powerful tool for specification of logic controllers. *IEEE transactions on control, systems technology*, 3(3).

Frehse, G. (2005). Phaver: Algorithmic verification of hybrid systems past hytech. In *The Fifth International Workshop on Hybrid Systems: Computation and Control*, pages 258–273.

Ghazel, M., Toguéni, A., and Bigang, M. (2005). A monitoring approach for discrete events systems based on a timed perti net model. In *Proceedings of 16th IFAC World Congress*.

Huang, Z., Chandra, V., Jiang, S., and Kumar, R. (1996). Modeling discrte event systems with faults using a rules based modeling formalism. *Mathematical Modeling of Systems*, 1(1).

S. H. ZAd, R. H. K. and Wonham, W. M. (2003). Fault diagnosis in discrete-event systems: Framework and model reduction. *IEEE Transactions On Automatic Control*, 48(7):1199–1212.

FAULT DETECTION ALGORITHM USING DCS METHOD COMBINED WITH FILTERS BANK DERIVED FROM THE WAVELET TRANSFORM

Oussama Mustapha [1,2], Mohamad Khalil [2,3], Ghaleb Hoblos [4], Houcine Chafouk [4] and Dimitri Lefebvre [1]

[1] *University Le Havre, GREAH, Le Havre, France*
[2] *Lebanese University, Faculty of Engineering, Section I- El Arz Street, El Kobbe, Lebanon*
[3] *Islamic University of Lebanon, Faculty of engineering, Biomedical Department, Khaldé, Lebanon*
[4] *ESIGELEC, IRSEEM, Saint Etienne de Rouvray, France*
oussama_mustapha@hotmail.com, mkhalil@ieee.org, ghaleb.holos@esigelec.fr
houcine.chafouk@esigelec.fr, dimitri.lefebvre@univ-lehavre.fr

Keywords: Signal, Filters Bank, DCS, Fault, detection, wavelet transform.

Abstract: The aim of this paper is to detect the faults in industrial systems, such as electrical machines and drives, through on-line monitoring. The faults that are concerned correspond to changes in frequency components of the signal. Thus, early fault detection, which reduces the possibility of catastrophic damage, is possible by detecting the changes of characteristic features of the signal. This approach combines the Filters Bank technique, for extracting frequency and energy characteristic features, and the Dynamic Cumulative Sum method (DCS), which is a recursive calculation of the logarithm of the likelihood ratio between two local hypotheses. The main contribution is to derive the filters coefficients from the wavelet in order to use the filters bank as a wavelet transform. The advantage of our approach is that the filters bank can be hardware implemented and can be used for online detection.

1 INTRODUCTION

The fault detection and diagnosis are of particular importance in industry. In fact, the early fault detection in industrial machines can reduce the personal damages and economical losses. Many researchers have performed fault detection by using mechanical conditions such as vibration analysis. Recently the current or voltage signature analysis is used for the detection of electromechanical faults, such as a broken bar in electrical drives (Sottile and Kohler, 1993; Schoen *et al.*, 1995; Kliman *et al.*, 1996). Other researchers use the AI tools (Awadallah and Morcos, 2003) and frequency methods (Benbouzid, *et al.*, 1999).The aim of this paper is to propose a method for the on-line detection of changes in the electric current feeding an induction motor due to a mechanical fault. The method is based on a filters bank, whose coefficients are derived from the wavelet, to decompose the signal in order to explore their frequency and energy components of the signal. Then, the Dynamic Cumulative Sum method is applied to the filtered signals in order to detect any change in the signal. The filters bank is derived from the wavelet transform, by using the Prony method, so the wavelet characteristics are approximately conserved and this allows both filtering and reconstruction of the signal. The main contributions are to derive the filters and to evaluate the error between filters bank and wavelet transform. This study continues our investigation concerning fault detection by means of wavelet transform and filters bank (Mustapha *et al.*, 2006a, 2006b). Extraction and detection will be applied on simulated and real signals. The real signals are issued from long duration experiments, with GREAH, on asynchronous machines of 4kW. These signals are recorded when the machine is properly operating and then when a bar of the same machine is broken.This paper is decomposed as follows. First we present the wavelet transform (WT) and the filters bank technique. In section 3 we detail the derivation of filters from a WT. In section 4, the Cumulative Sum and the Dynamic Cumulative Sum methods are presented. In section 5, some results are discussed. Then, the choice of the suitable filter are discussed in section 6.

2 WT AND FILTERS BANK

The Fourier analysis is the most well known mathematical tool used for transforming the signal from time domain to frequency domain. But it has an important drawback represented by the loss of time information when transforming the signal to the frequency domain. To preserve the temporal aspect of the signals when transforming them to frequency domain, one solution is to use is the WT (Truchetet, 1998) which analyzes non-stationary signals by mapping them into time-scale and time-frequency representation. The Wavelet Transform is similar to the Short Time Fourier Transform but provides, in addition, a multi-resolution analysis with dilated and shifted windows. The multi-resolution analysis consists of decomposing the signal *x(t)* using the wavelet $\psi(t)$ and its scale function $\phi(t)$ (Frandrin, 1993; Krim, 1995):

$$T_x^{\psi}(a,b) = \int_{-\infty}^{+\infty} x(t)\psi_{ab}(t)dt \ , \ \psi_{ab}(t) = \frac{1}{\sqrt{a}}\psi(\frac{t-b}{a}) \quad (1)$$

where a and b are respectively the dilation and translation parameters. The filter associated with the scale function $\phi(t)$ is a low pass filter and the filter associated to the wavelet $\psi(t)$ is a band pass filter. The following formulas can be used to calculate detail and approximation coefficients (Truchetet, 1998):

$$a_x(n,m) = \int_{-\infty}^{+\infty} x(t)\phi_{nm}(t)dt \quad (2)$$

$$d_x(n,m) = \int_{-\infty}^{+\infty} x(t)\psi_{nm}(t)dt \quad (3)$$

where m and n are integers**.**

In this way, the relevant events, to be detected, can be shown as details on specific scale levels. In Discrete Wavelet Transform (DWT), the multi-resolution analysis uses a scaling function and a wavelet to perform successive decomposition of the signal into approximations and details (figure 1: a and b).

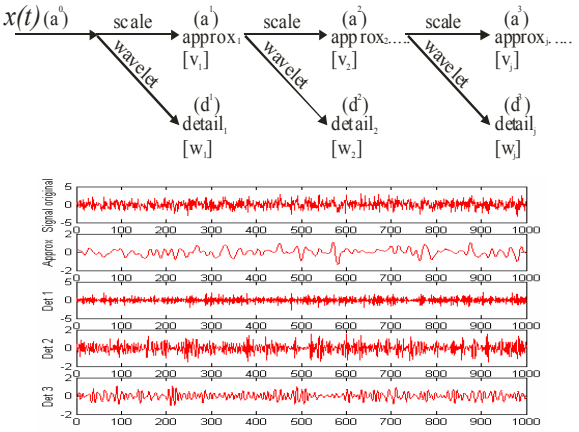


Figure 1: (a) multi-resolution analysis: Successive decomposition into approximations and details (b) multi-resolution analysis of the original signal into an approximation and three detail levels.

At each time t, the signal is first decomposed by using an N-channels band-pass filters bank whose central frequency moves from lowest frequency f_1 up to the highest frequency f_N. Each component $m \in \{1, \ldots, N\}$ is the result of filtering the original signal x by a band-pass filter centered on f_m. The frequency response of the filters bank is shown in (figure 2).

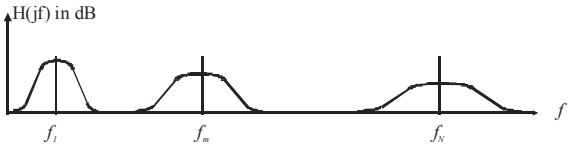


Figure 2: Response curves of the filters bank.

For each component m, the sample $y^{(m)}(t)$, is on-line computed according to the original signal *x(t)* and using the parameters $a_i^{(m)}$ and $b_j^{(m)}$ of the corresponding band-pass filter according to (4):

$$y^{(m)}(t) = \sum_{j=0}^{q} bj^{(m)}.x(t-j) - \sum_{i=1}^{p} a_i^{(m)} y(t-i) \quad (4)$$

where x is the original signal, f_s is the sampling frequency of the original signal x, f_N must satisfy the condition $f_N \leq f_s/2$, N is the number of channels used, p and q are the orders of the filter at level m. The choice of the filters bank depends on the original signal and its frequency band. The number of filters N depends on the details that we have to extract from the signal and on the events that must be distinguished. In our case we will use $N = 3$ filters.

The procedure of decomposing *x(t)* into signals $y^{(m)}(t)$, $m=1 \ldots N$, allows us to explore all frequency components of the signal. $y^{(1)}(t)$ gives the low

frequency components and $y^{(N)}(t)$ gives the high frequency ones. Therefore, the points of change of each component give information about the frequency and energy contents and will be used to detect any changes in frequency and energy in the original signal.

3 PRONY'S METHOD

In the present work, the main objective is to derive the filters coefficients of a filters bank from a wavelet in order to use the filters bank as a WT. The filters bank is derived from the WT, by using the Prony's method, so the wavelet characteristics are approximately conserved and this allows both filtering and reconstruction of the signal.

For a given wavelet, we can use the approximation coefficients of the wavelet function $\psi(t)$ to extract the coefficients a_i and b_i in order to design an IIR filter that behaves as the wavelet. The extraction of the filter coefficients can be done by using the Prony's method. The main advantage of the wavelet-derived filter is that it can be used instead of the wavelet and can be hardware implemented in order to be used for online signal filtration. Figure 3 shows the response curves (h_{wav}) of the wavelet function 'db3' and the response curves (h_{filt}) of the derived filter.

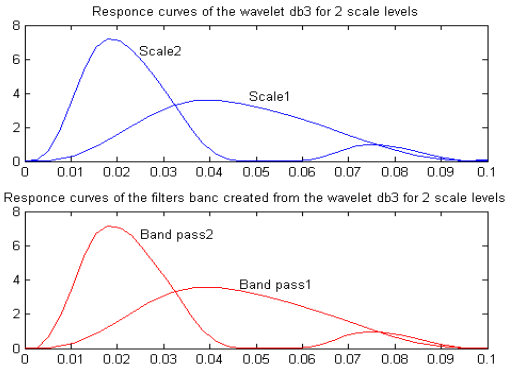


Figure 3: response curves of the wavelet function 'db3' and of the derived filters bank (the filter's order is 30).

Prony's method is an algorithm that can be used to find an IIR filter with a prescribed time domain impulse response. According to the time domain impulse response h_{wav} of the wavelet function $\psi(t)$, the numerator order p and the denominator order q of the desired filter, Prony's method is used to compute the filter's coefficients a_i and b_j, $i=1...p$ and $j=1...q$. If the length of h is less than the largest order (p or q), h is padded with zeros. It is fundamentally based on signal approximation with a linear combination of adjustable exponentials.

The impulse matching problem for modeling an entire causal signal $x(t)$, $t=0,1,...,\infty$, produces an infinite number of equations. The problem is to find the parameters a_i and b_i such that the equation (x) is satisfied:

$$\begin{bmatrix} x(0) & 0 & \cdots & 0 \\ x(1) & x(0) & \cdots & 0 \\ x(2) & x(1) & x(0) & \cdots 0 \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} \stackrel{?}{=} \begin{bmatrix} X_1 \\ \tilde{x} \;\; \tilde{X}_2 \end{bmatrix} \begin{bmatrix} 1 \\ \tilde{a} \end{bmatrix} \stackrel{?}{=} \begin{bmatrix} b \\ 0 \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_q \\ 0 \\ 0 \\ \vdots \end{bmatrix} \quad (5)$$

where X_1 is the top part of matrix X, $\tilde{a}$ is a p dimentional vector of parameters $\{a_i\}$, b is a $q+1$ dimentional vector of parameters $\{b_j\}$, $\tilde{x}$ is the first column of bottom part of matrix X, and $\tilde{X}_2$ is the p last columns of matrix X bottom part.

The equation $\tilde{x} + \tilde{X}_2 \tilde{a} \stackrel{?}{=} 0$ contains an infinite number of equations to be solved for $\tilde{a}$. This linear equation is usually over determined and no exact solution exists. This means that since the vector $\tilde{x}$ can only be approximated by the columns of matrix $\tilde{X}_2$, it's necessary to choose $\tilde{a}$ to minimize the equation error defined by the equation (6):

$$\varepsilon = e^T e = \sum_{t=q+1}^{\infty} e^2(t) = \sum_{t=q+1}^{\infty} \left\{ x(t) + \sum_{i=1}^{p} a_i x(t-i) \right\}^2 \quad (6)$$

with $e = \tilde{x} + \tilde{X}_2 \tilde{a}$. The error is minimized by using partial differentiation with respect to parameters $\{a_i\}$:

$$\tilde{X}_2^T \tilde{X}_2 \tilde{a} = -\tilde{X}_2^T \tilde{x} \quad (7)$$

In order that the orthogonality condition $e_{\min}^T \tilde{X}_2 = 0$ is satisfied, (7) provides a solution for the optimum vector $\tilde{a}$, which can then be used to find the solution to vector b by simple matrix multiplication in (5).

4 CUMSUM AND DCS

The Cumulative Sum algorithm (CUMSUM) algorithm is based on a recursive calculation of the logarithm of the likelihood ratios. This method can

be considered as a sequence of repeated tests around the point of change t_M (figure 4) (Nikiforov, 1986; Basseville and Nikiforov, 1993). For the seek of simplicity *x(t)* will be referred as x_t in the following. Let $x_1,x_2,x_3,...,x_t$ be a sequence of observations. Let us assume that the distribution of the process X depends on parameter θ_0 until time t_M and depends on parameter θ_1 after the time t_M. At each time t we compute the sum of logarithms of the likelihood ratios as follows:

$$S_1^{(t,m)} = \sum_{i=1}^{t} s^{(m)}{}_i = \sum_{i=1}^{t} Ln \frac{f_{\theta_1}(x_t / x_{t-1},...,x_1)}{f_{\theta_0}(x_t / x_{t-1},...,x_1)} \qquad (8)$$

The importance of this sum comes from the fact that its sign changes after the point of change. The detectability (Basseville and Nikiforov, 1993) is due to the fact that the expectation $E_{\theta_0}[s_i] < 0$ and $E_{\theta_1}[s_i] > 0$. We, then, calculate the following detection function $g^{(m)}{}_t = S_1^{(t,m)} - \min_{1 \le i \le t} S_1^{(i,m)}$. This function compares, at any time t, the difference between the value of the sum of the logarithm of the likelihood ratio and its minimal current value. The instant at which the procedure is stopped is $t_a = min\ \{t : g^{(m)}{}_t \ge h\}$, where h is the detection threshold (i.e the change can be detected when the detection function reaches a predetermined threshold h). The point of change can be defined as follows $t_M = max\ \{t : g^{(m)}{}_t = 0\}$.

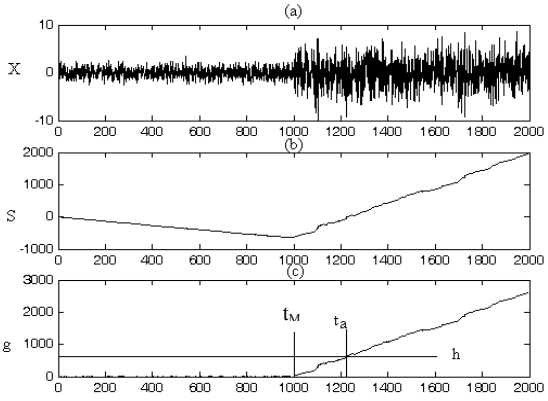


Figure 4: CUNSUM algorithm (a) Signal (b) CUNSUM (c) Detection function.

At any time t and for the observation vector $X = X_t = (x_1,...,x_t)$, suppose that the distribution of the process X depends on parameter θ. A change can affect the frequency distribution of the signal. The Dynamic Cumulative Sum method (DCS) is a repetitive sequence around the point of change t_M. It is based on the local cumulative sum of the likelihood ratios between two local segments estimated at the current time t. These two dynamic segments $S_a^{(t)}$ (« after t ») and $S_b^{(t)}$ (« before t ») are estimated by using two windows of width W (figure 5) before and after the time instant t as follows:

* $S_b^{(t)} : x_i ; \mathrm{i} = \{t-W,...,t-1\}$ follows a probability density function $f_{\theta_0}(x_i)$

* $S_a^{(t)} : x_i ; \mathrm{i} = \{t+1,...,t+W\}$ follows a probability density function $f_{\theta_1}(x_i)$

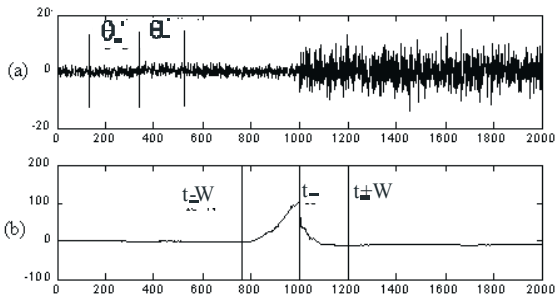


Figure 5: DCS algorithm (a) Signal; (b) Dynamic cumulative sum Cumulative sum; (c) Detection function.

The parameters $\hat{\theta}_b^{(t)}$ of the segment $S_b^{(t)}$, are estimated using W points before the time instant t and the parameters $\hat{\theta}_a^{(t)}$ of the segment $S_a^{(t)}$, are estimated using W points after the time instant t. At a time t, the DCS is defined as the sum of the logarithm of likelihood ratios from the beginning of the signal up to the time t:

$$DCS^{(m)}(S_a^{(t)}, S_b^{(t)}) = \sum_{i=1}^{t} Ln \frac{f^{(i)}_{\hat{\theta}_a}(x_i)}{f^{(i)}_{\hat{\theta}_b}(x_i)} = \sum_{i=1}^{t} \hat{s}_i \qquad (9)$$

(Khalil, 1999) proves that the DCS function reaches its maximum at the point of change t_M. The detection function used to estimate the point of change is

$$g^{(m)}{}_t = \max_{1 \le i \le t} \left[DCS^{(m)}\left(S_a^{(t)}, S_b^{(t)}\right) \right] - DCS\left(S_a^{(t)}, S_b^{(t)}\right) \qquad (10)$$

The instant at which the procedure is stopped is $t_a = inf\ \{t : g^{(m)}{}_t \ge h\}$, where h is the detection threshold. The point of change is estimated as $t_M = max\ \{t>1 : g^{(m)}{}_t = 0\}$. The DCS is a method that

can be used when the parameters of the signal are unknown.

5 RESULTS

The algorithm is first applied to simulated signals and then to real signals (figure 6). The simulated signal is generated by concatenating two random signals of different variances (σ_0=1 et σ_1=3), and two sinusoidal signals of different frequencies (f_0=150Hz et f_1=600Hz). Real and simulated signals are decomposed into 3 scales before applying the DCS method. These scales are computed by using the ARMA coefficients calculated by Prony's method and corresponding to the 'db3' wavelet. The coefficients of the derived filter of order 5 from the wavelet 'db3' for scale level 3 are detailed in the next table:

Table 1: Derived filters coefficients.

ai	1.000	-1.247	0.527	-0.165	0.604	-0.409
bi	0	0.0110	0.0130	0.0144	0.0203	0.0252

The results lead us to determine the point of change of statistical parameters of these signals.

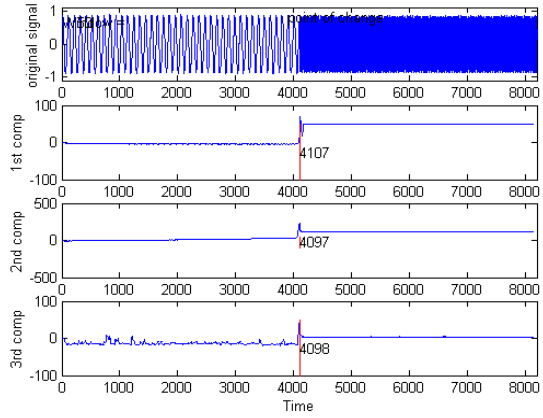


Figure 6: Detection of a real signal.

Table 2: Comparison of the points of change.

	Expected Time of change	1st comp.	2nd comp.	3rd comp.
1st simulated signal	1000	1006	1005	1002
2nd simulated signal	2000	2012	2003	2001
Real signal	4000	4107	4097	4098

Note that the third component, which is filtered by a highest central frequency band-pass filter, presents the closest point of change to the real one as shown in the table 2.

6 FILTER'S ORDER

In the wavelet theory, the choice of the wavelet is a critical problem. To extract the specific events in a signal, the choice of the wavelet is important to be adapted to the event to be detected. Many researchers have performed the detection by using the wavelet in different domains of application: in image edge detection (Mallat, 2000), for compression (Benbouzid *et al.*,1999), for signal de-noising in speech processing (Misiti *et al.*). In biomedical applications, the quadratic spline wavelet is used by Li (Li *et al.*,1995) and the complex wavelet is used by Shenhadji (Shenhadji *et al.*,1995) to process the ECG signal.
In our work, filters derived from many wavelets such as the Daubechies, the coiflet and the symlet wavelets are tested and according to the results obtained in figure 7, the filter derived from the wavelet 'db3' at level 3 has been used because it presents the minimum error and then it is chosen.

Note that the error is defined as follows:

$$error = \sum_{i=1}^{k} (h_{wav} - h_{filt})^2 \qquad (11)$$

Where, h_{wav} and h_{filt} are the impulse responses of the wavelet and the derived filter respectively.

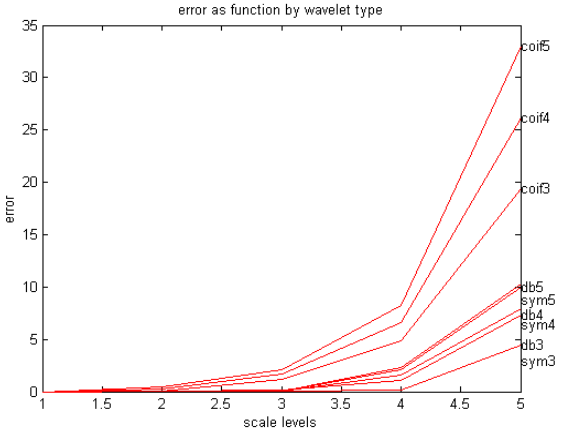


Figure 7: Error due to the use of different types of wavelets.

The orders of the filter (p and q) are very important parameters and can affect the error due to the application of Prony's method to extract the filter coefficients from the wavelet. As shown in figure 8, we can see that if the order of the filter becomes 30 and above, the error due to the derivation becomes negligible for filter derived from db3.

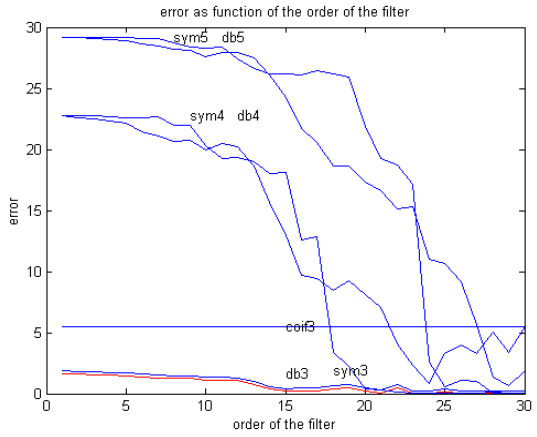


Figure 8: Error due to the order of the filter derived from different types of wavelets (scale 2 and p=q=30).

7 CONCLUSIONS

This article has proposed a method to detect the point of change of statistical parameters in signals issued from industrial machines. This method uses a band-pass filters bank, derived from a wavelet transform, to decompose the signal and the DCS algorithm to characterize and classify the parameters of a signal in order to detect any variation of the statistical parameters due to any change in frequency and energy. The main contribution of the work is to find a filters bank that approximates a wavelet. The filters bank derivation is done by using the Prony's method. After the calculation of the resulting error, between the derived filters bank and the correspondent wavelet, the wavelet 'db3' has been selected. In order to reduce the error due to the order of the derived filter, the order is taken to be beyond 30. This on-line algorithm is developed and tested and it gives good results for the detection of changes in the signals. It is necessary to test the algorithm with other types of wavelets, to explain the error depending on the scale levels, and to implement the whole algorithm in a DSP. The detectability of DCS must be proved after decomposing the signal, especially after using the ARMA decomposition. Another perspective is to complete the filters design by determining the optimum orders *p* and *q*.

REFERENCES

Sottile J, Kohler J. An on-line method to detect incipient failure of turn insulation in random wound motors. IEEE Trans Energy Conver 1993;8(4):762–8.

Schoen RR, Habetler TG, Kamran F, Bartheld RG. Motor bearing damage detection using stator current monitoring. IEEE Trans Ind Appl 1995;31(6):1274–9.

Kliman GB, Premerlani WJ, Koegl RA, Hoeweler D. A new approach to on-line turn fault detection in AC motors. In: Proceedings of IEEE-IAS Annual Meeting, 1996:687–93.

Awadallah M.A,Morcos M.M., Application of AI tools in faults diagnosis of electrical machines and drives – an verview, Trans. IEEE Energy Conversion, vol. 18, no. 2, pp. 245-251, june 2003.

Benbouzid M., Vieira M., Theys C., "Induction motor's faults detection and localization using stator current advanced signal processing techniques" IEEE Transaction on Power Electronics, Vol. 14, N° 1, pp 14 – 22, January1999.

Truchetet T. Ondelettes pour le signal numérique, collection traitement du signal, HERMES, Paris, 1998.

Flandrin P. Temps fréquence, HERMES, Paris, 1993.

Krim H., Pesquet J.C. Multiresolution analysis of a class of non stationnary processes. IEEE transaction on information theory, 1995, vol. 41, No 4,pp 1011-1020.

Mallat S., Une exploration des signaux en ondelettes, les éditions de l'école polytechnique, Paris, juillet 2000.

Nikiforov I. Sequential detection of changes in stochastic systems. Lecture notes in Control and information Sciences, NY, USA, 1986, pp. 216-228.

Basseville M., Nikiforov I. Detection of Abrupt Changes: Theory and Application. Prentice-Hall, Englewood Cliffs, NJ, 1993.

Khalil M. Une approche pour la détection fondée sur une somme cumulée dynamique associée à une décomposition multiéchelle. Application à l'EMG utérin. 17ème Colloque GRETSI sur le traitement du signal et des images, Vannes, France,1999.

Li C., Zheng C., Tai C. Detection of ECG characteristic points using wavelet transform. IEEE transaction, BME, 1995, vol.42, No 1, pp 21- 29.

Shenhadji L., Bellanger J.J., Carraut G. Détection temps échelle d'événements paroxystiques intercriptiques en électroencéphalogramme, traitement du signal, 1995, vol.12, No 4, pp 357-371.

Misiti M., Misiti Y., Oppenheim G., Poggi J.M.. Wavelet Toolbox for use with MATLAB® Computation Visualization Programming. The MathWorks User's guide version 4.

Mustapha O., Khalil M., Hoblos G., Chafouk H., Ziadeh H., Lefebvre D., On-Line Fault Detection by Using Filters Bank and Artificial Neural Networks IEEE – ICCTA, Damascus, Syria, April 2006.

Mustapha O, Khalil M., Hoblos G, Chafouk H., Ziadeh H., Lefebvre D., On-Line Change Detection by Using Filters Bank/Wavelet Transform and Dynamic Cumulative Sum Method, 4th FAI International Conference, Lefke, Cyprus, December 2006.

SLIDING MODE CONTROL FOR HAMMERSTEIN MODEL BASED ON MPC

Zhiyu Xi and Tim Hesketh
School of Electrical Engineering & Telecommunications
University of New South Wales
Australia

Keywords: Sliding mode, reaching control, equivalent control, MPC, Hammerstein model, nonlinearity.

Abstract: This paper addresses discrete sliding mode control of nonlinear systems. The nonlinear system is identified as a Hammerstein model firstly to isolate the nonlinearity from the sliding surface design. An MPC law is employed to design the sliding surface. Then Utkins method of equivalent control is used. The method illustrates the effect of the nonlinearity on reaching control. The ball and beam system is adopted as an example. Simulation and on-line results are provided.

1 INTRODUCTION

Variable structure systems (VSS) have been extensively used for control of dynamic industrial processes. The essence of variable structure control (VSC) (Raymond et al., 1988) is to use a high speed switching control scheme to drive the nonlinear plant's state trajectory onto a specified and user chosen surface in the state space which is commonly called the sliding surface or switching surface, and then to keep the plant's state trajectory moving along this surface. The surface is chosen to produce specified dynamic behaviour. Once the state trajectory intercepts the sliding surface, it remains on the surface for all subsequent time, sliding along the surface, leading to the term "sliding mode". Sliding mode controller design comprises two stages. The first is the design of sliding surface, while the second forces the state to approach the sliding surface from any other region of the state space, and remain on it.

The ball and beam system is a widely used laboratory process. It reflects typical control problems which include a double integrating factor, nonlinearity, time delay and noise. In the ball and beam system, a conductive ball lies on the beam comprised of two parallel rods, and is free to roll along the beam. A resistive strip, with impedance proportional to length, covers one of the rods. The other rod is conductive. The position of the ball can be determined by introducing a small current through the rods and measuring the resulting voltage, which varies with impedance as the ball moves. One end of the beam is fixed and the other is mounted on the output shaft of a DC servo motor so the beam is tilted as the motor shaft rotates. The control task is to regulate the position of the ball by altering the angular shaft position of the DC motor.

Design from an identified model has potential advantages in nonlinear control for the ball and beam, and more generally. It relies on mathematical tools and algorithms that build dynamical models from measured data. Relatively simple structures of nonlinearity may be used to describe complex nonlinear systems or ones for which models are difficult to derive. In this paper, discrete sliding mode control of a system described by a Hammerstein model will be addressed. This provides a simple method to deal with nonlinear systems using VSC. The ball and beam system will be used as an example to illustrate the design procedure.

The control of a Hammerstein model has been addressed in the past by several authors (cite15,cite16). Satisfying performance has been derived. In (Hwang and Hsu), Hwang and Hsu talked about nonlinear control profile based on Hammerstein model in case of model uncertainty. They also introduced an inverse block into the system. Meanwhile, they spent a lot effort on designing an observer.

2 HAMMERSTEIN MODEL

Hammerstein models are amongst those most commonly used for nonlinear identification. They are capable of providing simple nonlinear models for a wide range of engineering problems. The model is characterized by a static nonlinearity followed by a linear time invariant (LTI) block. A typical Hammerstein model for a process is shown in Figure 1:

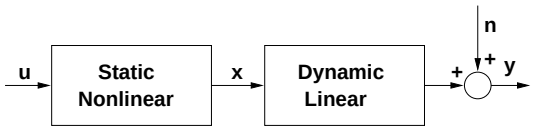


Figure 1: Typical Hammerstein model.

$$y(t) = Tx(t) + n(t) \quad (1)$$
$$x(t) = f(u(t)) \quad (2)$$

where $x(t)$ and $y(t)$ are the inputs and outputs respectively, $n(t)$ is additive noise, f is the nonlinear mapping, T is the transfer function of linear part which can be written as

$$T = \frac{b_0 + b_1 q^{-1} + b_2 q^{-2} + \cdots + b_m q^{-m}}{1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_n q^{-n}}$$

with q^{-1} representing the unit delay operator.

In this way, the nonlinearity of system is separated from the linear block. This leads to the possibility of ignoring the nonlinearity during key steps in the controller design. Also, the nonlinear block in the Hammerstein model is a polynomial, which is a relatively simple form. This reduces problems introduced by complex nonlinearities such as exponentials and sinusoids.

3 DISCRETE SLIDING MODE CONTROL DESIGN

3.1 Sliding Surface Design

Suppose the state space model of the above Hammerstein model is:

$$z(t) = Az(t-1) + Bx(t) \quad (3)$$
$$y(t) = Cz(t) + \bar{n}(t) \quad (4)$$

where $x(t) = f(u(t))$.

Performing a similarity transformation defined by an orthogonal matrix P:

$$z_l = Pz = [z_1 : z_2]^T, A_l = PAP^T, B_l = PB = \begin{bmatrix} 0 \\ B_2 \end{bmatrix}, \quad (5)$$

where z_1 does not have direct dependence on the input nonlinearity. Sliding surface design may be undertaken considering only z_1, treating z_2 as an "input" to the partitioned equations. In this way, the nonlinearity may be ignored while determining the sliding surface, which is linear.

The partitioned state equations corresponding to (3) and (4) may now be expressed in the following way:

$$z_{l1}(t+1) = A_{l11} z_{l1}(t) + A_{l12} z_{l2}(t) \quad (6)$$
$$z_{l2}(t+1) = A_{l21} z_{l1}(t) + A_{l22} z_{l2}(t) + B_{l2} x(t) \quad (7)$$

Suppose

$$sz_l(t) = \begin{bmatrix} s_1 & s_2 & \cdots & s_v \end{bmatrix} z_l(t) = w_1 z_{l1}(t) + w_2 z_{l2}(t)$$

in which v is the dimension of the corresponding state vector, and s is the sliding surface, then the sliding condition is

$$w_1 z_{l1}(t) + w_2 z_{l2}(t) = 0,$$

which yields

$$z_{l2}(t) = -w_2^{-1} w_1 z_{l1}(t). \quad (8)$$

Substitute (8) into (6) then we have,

$$z_{l1}(t+1) = A_{l11} z_{l1}(t) - A_{l12} w_2^{-1} w_1 z_{l1}(t) \quad (9)$$
$$= (A_{l11} - A_{l12} w_2^{-1} w_1) z_{l1}(t). \quad (10)$$

Any standard design algorithm which produces a linear state feedback controller for a linear dynamic system can be used to determine $(A_{l11} - A_{l12} w_2^{-1} w_1)$ and achieve desired performance through selection of sliding mode dynamics (Spurgeon, 1992). Pole placement is an obvious way of assigning closed loop eigenvalues, but for systems of higher order the method has attendant difficulties.

MPC is a widely-used method for calculating closed-loop feedback controller gains. It is suitable for systems with high order. It is employed to determine the sliding surface in this paper. Considering $z_{l1}(t+1) = A_{l11} z_{l1}(t) + A_{l12} z_{l2}(t)$, $z_{l2}(t)$ can be viewed as the input to a new system the state vector of which is $z_{l1}(t)$. An MPC criterion minimizes the cost function which is defined to be:

$$\mathbf{J} = M_{t+1}^\top M_{t+1} + \lambda \mathbf{U}_t^\top \mathbf{U}_t. \quad (11)$$

where

$$M_{t+1} = \begin{bmatrix} z_{l1}(t+1) \\ z_{l1}(t+2) \\ \cdots \\ z_{l1}(t+N) \end{bmatrix}, U_t = \begin{bmatrix} z_{l2}(t) \\ z_{l2}(t+1) \\ \cdots \\ z_{l2}(t+N-1) \end{bmatrix}.$$

The goal is to fix the relationship between $z_{l2}(t)$ and $z_{l1}(t)$ to prescribe desirable performance for the nominal sliding mode dynamics. The controller gain derived is:

$$z_{l2}(t) = -k z_{l1}(t) \quad (12)$$

which means that

$$\sigma(\mathbf{z}_l(t)) = \begin{bmatrix} k & \vdots & I \end{bmatrix} z_l(t) \tag{13}$$

Note that inversion of the similarity transformation (using P) is needed to recover $z(t)$ from $\mathbf{z}_l(t)$. Then $sz(t)$ is the sliding surface.

3.2 Sliding Mode Controller Design

The reaching law still applies for discrete systems. However, the state trajectory may overshoot the sliding surface repeatedly, so that true sliding does not occur. The switching manifold of a discrete VSC system is called an ideal switching manifold because in all practical situations, switching seldom occurs on it. The size of each successive overshoot is non-increasing and the trajectory stays within a specified band which is called a *quasi-sliding mode* (QSM). The specified band is called *quasi-sliding mode band* (QSMB) (Gao et al., 1995) and is defined by

$$\{x \mid -\Delta < s(x) < \Delta\} \tag{14}$$

where 2Δ is the width of the band.

Consider the single input linear system with switching manifold s, a common type of sliding mode controller is:

$$u(t) = u_{eq}(t) + u_2(t) \tag{15}$$

where $u_{eq}(t)$ represents the equivalent control which ensures sliding and $u_2(t)$ drives the state onto the sliding surface, (termed reaching control).

According to the definition of sliding mode, we have

$$\sigma(t+1) = sz(t+1) = sAz(t) + sBu_{eq}(t) = \sigma(t). \tag{16}$$

and

$$\sigma(t) = 0.$$

From the above, the equivalent control can be described as follows:

$$u_{eq}(t) = -(sB)^{-1} sAz(t). \tag{17}$$

Then let us consider the reaching control law. For continuous SMC problem, a simple Lyapunov function $V(\sigma(z)) = 0.5\sigma^T(z)\sigma(z)$ is considered. The corresponding reaching condition is

$$\frac{\partial V}{\partial t} = \sigma^T \dot{\sigma} < 0. \tag{18}$$

In discrete system design, the equivalent form of this condition is

$$[\sigma(t+1) - \sigma(t)]\sigma(t) < 0. \tag{19}$$

Substitute (3), (4), (15) and (17) into (19) then:

$$\begin{aligned} [\sigma(t+1) - \sigma(t)]\sigma(t) &= (sAz(t) + sBu(t) - sz(t))sz(t) \\ &= ((sB)((sB)^{-1} sAz(t) \\ &\quad + u(t)) - sz(t))sz(t) \\ &= (sB(u(t) - u_{eq}(t)) \\ &\quad - sz(t))sz(t). \\ &= (sBu_2(t) - sz(t))sz(t). \end{aligned}$$

$u_2(t)$ should be selected to ensure that:

$$sBu_2(t) < sz(t) \quad when \;\; sz(t) > 0 \tag{20}$$

$$sBu_2(t) > sz(t) \quad when \;\; sz(t) < 0. \tag{21}$$

As mentioned before, in a discrete sliding mode control system, the switching manifold is actually an ideal one. To eliminate the overshoot, the reaching law should be modified. Once the state trajectory enters a specified band around the manifold, the reaching control action ceases and only sliding control applies. The goal is to keep the state trajectory within the specified band.

The modified reaching control law is:

$$sBu_2(t) < sz(t) \text{ when } sz(t) > \Delta \tag{22}$$

$$sBu_2(t) > sz(t) \text{ when } sz(t) < -\Delta. \tag{23}$$

Considering equations (18)-(21) and absolute values of $\sigma(t+1)$ and $\sigma(t)$, if

$$\|\sigma(t+1)\| < \|\sigma(t)\|, \tag{24}$$

it can be concluded that the state trajectory is towards the sliding surface. On the contrary, if

$$\|\sigma(t+1)\| > \|\sigma(t)\|, \tag{25}$$

the trajectory is away from the sliding surface.

Note that (24) is equivalent to

$$\|sBu_2(t)\| < \|\sigma(t)\|, \tag{26}$$

and (25) is equivalent to

$$\|sBu_2(t)\| > \|\sigma(t)\|. \tag{27}$$

The conclusion may be drawn that while the trajectory is outside the ($\varepsilon = \|sBu_2(t)\|$), the trajectory will approach the surface. While the state is within this specified neighborhood, it moves in the direction of leaving the surface (Hui and Zak, 1999). Thus $u_2(t)$ has to be carefully chosen because the value of $\varepsilon = \|sBu_2(t)\|$ is the crucial factor which determines the radius attraction around the sliding surface.

Figure 2. shows the structure of closed loop sliding mode control system: where P is the plant and W represents the reaching controller.

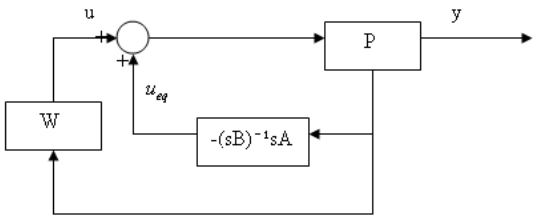


Figure 2: Structure of closed loop sliding mode control system.

3.3 Sliding Mode Control for a Hammerstein Model

In a Hammerstein model, the nonlinearity has been separated from the linear block already. The nonlinear mapping f is a smooth polynomial with respect to its input, hence it is invertible. Thus in the controller design stage, the nonlinearity can be ignored and the controller is only designed based on the linear block and afterwards, the control signal is filtered through an inverse of the nonlinearity before being sent to the plant. This may not be necessary if the nonlinearity is taken into account in the Lyapunov function used for determination of the reaching control.

As far as the description of the linear block is concerned, a non-minimal state space model is employed (refer to (Xi and Hesketh) for details of derivation of non-minimal state space models). The motivation for this is that in a non-minimal state space model, the error signal sequence is contained in state vector (the error signal being the difference between the plant output and the desired trajectory). Sliding mode control results in a regulator, where the state variables will be constrained to move along the sliding surface and eventually reach and stay at zero. Thus the error signal will be regulated to approach zero and stay there if the sliding mode control is based on a non-minimal state space model. This is the aim of tracking control. Figure 3. shows the structure of such a control system.

In the figure, r represents the set point and e is the difference between filtered set point and the output of plant (Xi and Hesketh). For our example, the "filter" is selected simply as q^{-1}. Here both equivalent and reaching controller are related to the tracking error and the set point. The control action force the states to reach the sliding manifold and move along it. This action continues until the plant output is equal to the filtered set point, which is equivalent to the error signal being zero. Then the states will be kept at the origin and the system achieves steady state.

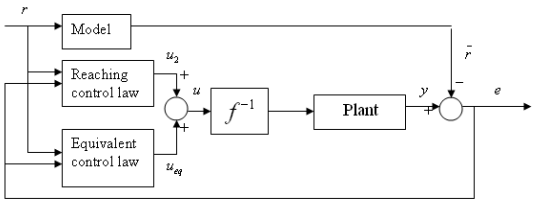


Figure 3: Structure of sliding mode control system based on Hammerstein model.

4 EXAMPLE AND SIMULATION RESULTS

4.1 Identification of Ball and Beam

Deriving an approximate model of the ball and beam system involves determining the transfer function between the input signal (the shaft angle of the motor) and the output signal (the position of the ball). In an identification experiment, a pseudo-random sequence is applied to the input signal, and both input and output signals are sampled (For the ball and beam the sampling interval selected was 1 second). Here Captain Toolbox which is written for Matlab is used to realize the identification ((Young et al., 2001), http://www.es.lancs.ac.uk/cres/captain/). The result of the Hammerstein model identification is:

$$y(t) = 0.962y(t-1) + 0.396u(t-1) - 0.036u^2(t-1)$$

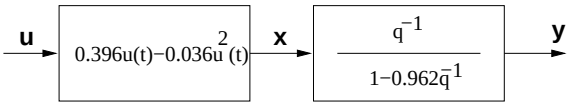


Figure 4: Identification result of Hammerstein model.

The resultant non-minimal state space model of the linear block is:

$$z(t) = Az(t-1) + B\Delta x + Q\Delta r(t) \quad (28)$$

$$y(t) = Cz(t) \quad (29)$$

where

$$A = \begin{bmatrix} 1.962 & -0.962 & -1 & 0.962 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, Q = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, z(t) = \begin{bmatrix} e \\ eq^{-1} \\ \Delta r \\ \Delta r q^{-1} \end{bmatrix}.$$

Suitable differencing is undertaken to introduce $\Delta = 1 - q^{-1}$. Note the way in which the setpoint is introduced within the state vector. This results in feedforward action, achieved with the sliding mode control.

4.2 Controller Design and Simulation

This system is typically unobservable. Performing of observable/unobservable decomposition prevents singularity occurrence later. The system model becomes:

$$\bar{z}(t) = \bar{A}\bar{z}(t-1) + B\Delta\bar{x}(t-1) + Q_{ob}\Delta r(t) \quad (30)$$

$$\bar{y}(t) = \bar{C}\bar{z}(t) \quad (31)$$

where the transformation matrix is T and

$$\bar{A} = \begin{bmatrix} 0 & 0.5698 & 0.4188 & 0.7071 \\ 0 & 0.3374 & 0.2480 & -0.4188 \\ 0 & -0.4591 & -0.3374 & 0.5698 \\ 0 & 0 & -1.6885 & 1.962 \end{bmatrix},$$

$$\bar{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \bar{C} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}.$$

Extracting the observable part we have:

$$z_{ob}(t) = A_{ob}z_{ob}(t-1) + B_{ob}\Delta u_{ob}(t-1) \quad (32)$$

$$y_{ob}(t) = C_{ob}z_{ob}(t) \quad (33)$$

where

$$A_{ob} = \begin{bmatrix} 0.3374 & 0.2480 & -0.4188 \\ -0.4591 & -0.3374 & 0.5698 \\ 0 & -1.6885 & 1.962 \end{bmatrix}, B_{ob} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

In this case, the requirement of equation (5) has already been satisfied so that no further transformation P is needed. Then

$$A_{ob11} = \begin{bmatrix} 0.3374 & 0.248 \\ -0.4591 & -0.3374 \end{bmatrix}, A_{ob12} = \begin{bmatrix} -0.4188 \\ 0.5698 \end{bmatrix},$$

$$A_{ob21} = \begin{bmatrix} 0 & -1.6885 \end{bmatrix}, A_{ob22} = [1.962]$$

The result of optimization is $\begin{bmatrix} k_1 & k_2 \end{bmatrix}$.

Note that an inversion of the observable/unobservable decomposition is to be performed to recover the state vector $z_{ob}(t) = Tz(t)$ after sliding surface design,

$$\sigma(z_{ob}(t)) = \begin{bmatrix} k_1 & k_2 & 1 \end{bmatrix} z_{ob}(t) \quad (34a)$$

$$\sigma(z(t)) = \begin{bmatrix} k_1 & k_2 & 1 \end{bmatrix} Tz(t) = sz(t) \quad (34b)$$

In this case, $u_2(t)$ is chosen to be $-\alpha sgn(\sigma(x(t)))$. Figure 5. shows the performance of the above sliding mode design. The figure shows simulation results, but on-line control is similar.

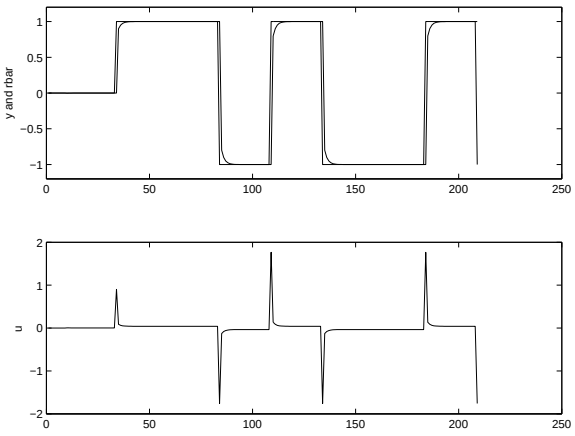


Figure 5: Performance with a linear model.

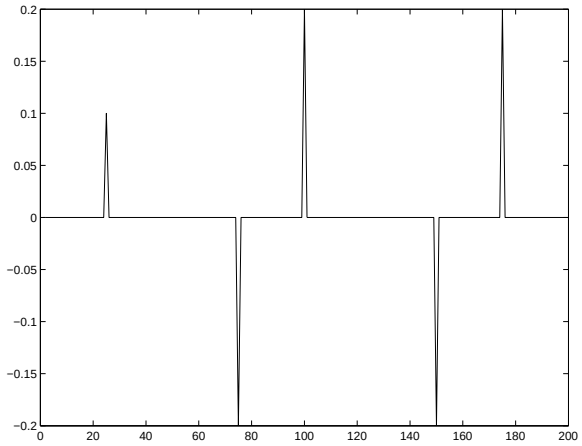


Figure 6: $\sigma(t)$ defined in equation (13).

Figure 6. shows the values of $\sigma(t)$ And Figure 7. shows the state trajectory:

It is shown that the system follows the ideal trajectory. The switching law works while sliding condition is not met. Equivalent control regulates the state to move along the sliding surface until the equilibrium point is achieved.

5 CONCLUSION

In this paper, discrete sliding mode control is applied to a Hammerstein model which results from nonlinear system identification. The nonlinearity is separated during the sliding surface design, so that the switching surface is actually a linear one. The surface is derived by an MPC approach. The nonlinearity is re-

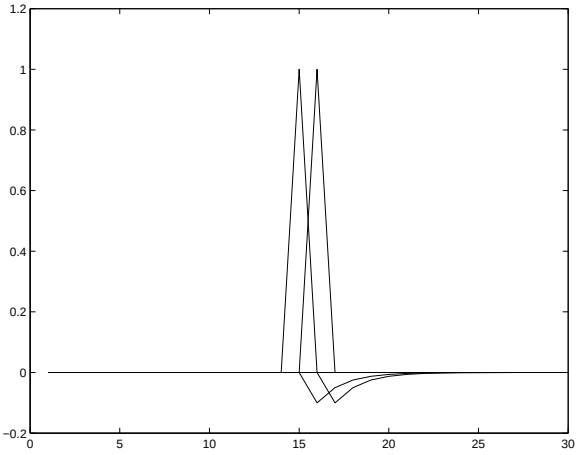


Figure 7: State variables.

considered in design of the reaching control. The ball and beam system is used as an example and simulation results show satisfying performance.

REFERENCES

Bitmead, R. R., Gevers, M., and Wertz, V., 1990, *Adaptive Optimal Control: The Thinking man' s GPC (Prentice Hall).*

Clarke, D. W., Mohtadi, C., and Tuffs, P. S., 1987, *Generalized Predictive Control– Part I. The Basic Algorithm*, Automatica, 23, 137- 148;

Clarke, D. W., Mohtadi, C., and Tuffs, P. S., 1987, *Generalized Predictive Control–Part II. Extensions and Interpretations*, Automatica, 23, 149-160

C. James Taylor, Arun Chotai and Peter C. Young, 2000, *State space control system design based on non-minimal state-variable feedback: further generalization and unification results*, International Journal of Control, 2000, Vol. 73, No. 14, 1329- 1345

Raymond, A DeCarlo, Stanislaw H. Zak, Gregory P.Matthews, *Variable Structure Control of Nonlinear Multivariable Systems: A Tutorial*, Proceedings of The IEEE, Vol. 76, No. 3, March 1988

S. K. Spurgeon, *Temperature Control of Industrial Process using a Variable Structure Design Philosophy*, Trans Inst MC Vol. 14, No. 5, 1992

Stefen Hui, Stanislaw H. Zak, *On discrete-time variable structure sliding mode control*, Systems & Control Letters 38 (1999) 283-288

Weibong Gao, Yufu Wang, Abdollah Homaifa, *Discrete-Time Variable Structure Control Systems*, IEEE Transactions on Industrial Electronics, Vol. 42, No.2, April 1995

Jozef Voros, *System Identification of Discontinuous Hammerstein Systems,* Automatica Vol.33.No.6. pp. 1141-1146, 1997

F. Giri, F. Z. Chaoui, Y. Rochdi, *Parameter identification of a class of Hammerstein plants*, Automatica, 37 (2001) 749-756

Xinghuo Yu and Guanrong Chen, *Discretization Behaviors of Equivalent Control Based Sliding-Mode Control Systems*, IEEE Transaction On Automatic Control, Vol. 48, No. 9, September 2003

Xinghuo Yu, Shuanghe Yu, *Discrete Sliding Mode Control Design With Invariant Sliding Sectors*, Transaction of the ASME, Vol. 122, Page776-782, December 2000.

Zhiyu Xi, Tim Hesketh, *MPC With a NMSS Model*, The Sixth International Conference on Control and Automation.

Peter C. Young, Paul McKenna, John Brunn, *Identification of Nonlinear Stochastic Systems by State Dependent Parameter Estimation*, International Journal of Control, 2001, Vol. 74, No. 18, 1837-1857

C. L. Hwang and J. C. Hsu, *Nonlinear control design for a Hammerstein model system*, IEE Procedings, Control Theory and Applications, Vol 142, Issue 4, p.p 277-285

Zi Ma, Arthur Jutan, Vladimir B. Bajic, *Nonlinear self-tuning controller for Hammerstein plants with application to a pressure tank*. Int. J. Comput. Syst. Signal, 1(2), 221-230, 2000

BICYCLE WHEEL WOBBLE
A Case Study in Dynamics

John V. Ringwood and Ruijuan Feng
Dept. of Electronic Engineering
NUI Maynooth
County Kildare, Ireland
john.ringwood@eeng.nuim.ie, ruijuan.feng@nuim.ie

Keywords: Bicycle, dynamics, wheel wobble, instability.

Abstract: This paper examines reasons why wheel wobble occurs in common production bicycles. In particular, the effects of frame size, rider position and riding style are examined with reference to a range of mathematical models of bicycles which are available in the published literature. Much of the motivation for this work comes from the personal cycling experience of one of the authors and the difficulty in resolving the true cause of wheel wobble from the wide range of advice offered of a variety of cycling experts. It is hoped that recourse to a mathematical analysis will give objective direction as to how wheel wobble can be alleviated through rider intervention.

1 INTRODUCTION

Since 1869 (Rankine, 1869), engineers have been intrigued by the dynamics of the simple bicycle. In recent years, there has been somewhat of a resurgence of interest with the appearance of a number of excellent review papers, such as (Astrom et al., 2005; Limebeer and Sharp, 2006), with reference to almost 170 technical works on the subject. From a dynamics perspective, the study of the bicycle is attractive for many reasons:

- The bicycle is ubiquitous in our lives as a device for commuting, recreation and sport,
- It permits various levels of complexity of analysis, from an interesting lumped-parameter system with non-minimum phase characteristics to a complex system described by a distributed-parameter model,
- Since bicycles are relatively easy to construct, they can be used as mechanical engineering testbeds, with a range of configurations limited only by imagination (Klein, 1989), and
- With the drive towards decreased weight and improved performance, commercial developers in bicycle and motorsport have a great interest in the dynamics of single-track vehicles (Beghi and Frezza, 2006; Hauser and Saccon, 2006).

The motivation for the current study comes from the modest, but important, aspiration of trying to stay upright on a road (racing) bicycle during high-speed descents. While it might be true to say that the aspirant in question (one of the authors on this paper!) is not the most accomplished/confident bike rider, some contribution by the bicycle dynamics to the rider's problem is likely, since:

- The author in question has consistently experienced speed wobble (sometimes called 'shimmy' (Brandt, 2005)) with two well-known production bicycles, but not with a third, and
- The bicycles which exhibited wheel wobble have been successfully ridden by many other cyclists (including Lance Armstrong in his 2003 Tour de France success) with no evidence of wheel wobble.

Wheel wobble is a spontaneous steering oscillation of the front wheel, usually building as the speed approaches a certain threshold. Many theories and myths exist in relation to wheel wobble, including:

1. Wheel wobble is a function of the natural bicycle dynamics, not rider induced i.e. it is inherent to the geometry and elasticity of the bicycle frame (Brandt, 2005).
2. The common rider response of gripping the handlebars tighter only serves to increase the oscillation.
3. Shimmy can be minimised by keeping the knee against the crossbar (increases damping).

4. Shimmy is less prevalent in bicycles which have a longer trail (see model in Section 2).
5. Shimmy is more likely in bicycles with longer frames and higher saddles.
6. Weight distribution has no effect on shimmy although where that weight contacts the frame does (Brandt, 2005).
7. Shimmy is due to loose bearings or eccentrics in the rotational masses (most common reason given, but refuted my many experts).

It is also the experience of the author that shimmy tends to accompanying braking - release of brakes appears to diminish the wobble oscillation amplitude or remove it altogether.

The objective of this study is to examine the range of possible reasons from a pragmatic dynamical systems perspective to see if the root cause of wheel wobble can be identified and to look for any rider interventions (such as weight distribution, braking protocol, etc) which may help to alleviate the symptom.

2 BICYCLE MODEL

The model used in this study is based on that presented in (Astrom et al., 2005). Assumptions under which the model is developed include:

- The bicycle consists of four rigid parts: Frame (including rider), front fork with handlebars and two wheels.
- The influence of other moving parts, such as pedals, chain and brakes on the dynamics is neglected.
- The forward velocity of the bicycle, V, is constant.

Figs.1 and 2 show the representations of the parameters and variables of the system (respectively), with:

O being the centre of mass (including rider)

c is known as the *trail*

λ is the head angle

ϕ is the lean angle

δ is the steering angle

The front fork is key to the ability of a bicycle to self-stabilize and the key relationship is that between ϕ and δ. For small angles, the front fork roll angle is:

$$\phi_f = \phi - \delta cos(\lambda) \tag{1}$$

and the effective front fork steering angle is:

$$\delta_f = \delta sin(\lambda) \tag{2}$$

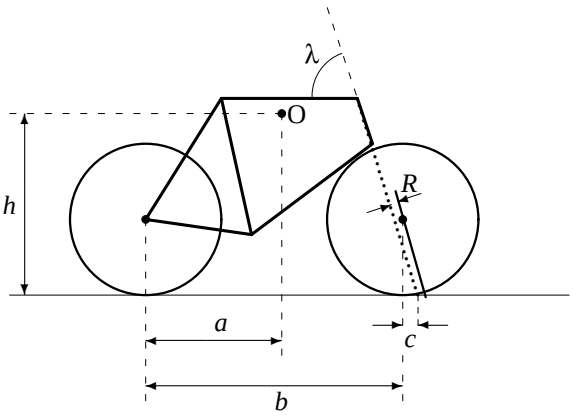


Figure 1: Bicycle parameters.

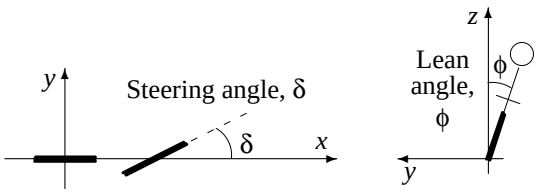


Figure 2: Bicycle variables.

A static torque balance is used to model the front fork, neglecting dynamics and gyroscopic effects. If N_f and F_f are the vertical and horizontal forces acting on the front wheel at the ground contact, then:

$$N_f = a\, m\, g/b \tag{3}$$

where m is the combined mass of rider and frame, and

$$F_f = amV^2\delta_f/b^2 \;=\; \frac{a\, m\, V^2\, sin(\lambda)}{b^2}\,\delta \tag{4}$$

The static torque balance for the front fork, assuming negligible mass of the front fork assembly, is:

$$T - (F_f + N_f\phi_f)c\, sin(\lambda) \tag{5}$$

where T is the external torque applied to the handlebar. Using (1), (2), (3) and (4), (5) reduces to:

$$T - \frac{a\, c\, m\, g\, sin(\lambda)}{b}\phi -$$

$$\frac{a\, c\, m\, sin(\lambda)}{b^2}(V^2 sin(\lambda) - b\, g cos(\lambda))\delta = 0$$

where g is the acceleration due to gravity. The torque balance can be rewritten as:

$$\delta = k_1(V)T - k_2(V)\phi \tag{6}$$

which demonstrates negative feedback (self-stabilization) between ϕ and δ if k_2 is +ve, or

$$V > \sqrt{b\, g\, cot(\lambda)} \tag{7}$$

where:

$$k_1(V) = \frac{b^2}{(V^2 sin(\lambda) - b\,g\,cos(\lambda))m\,a\,c\,sin(\lambda)} \quad (8)$$

and

$$k_2(V) = \frac{b\,g}{V^2 sin(\lambda) - b\,g\,cos(\lambda)} \quad (9)$$

Note that the center of mass of the frame is shifted when the front wheel is turned, giving the torque:

$$T_\delta = -\frac{m\,g\,a\,c\,sin(\lambda)}{b}\delta \quad (10)$$

An overall angular momentum balance for the frame can now be constructed as:

$$J\frac{d^2\phi}{dt^2} - mgh\phi = \frac{DVsin(\lambda)}{b}\frac{d\delta}{dt} + \frac{m(V^2h - acg)sin(\lambda)}{b}\delta \quad (11)$$

where J is the moment of inertia of the bicycle with respect to the x-axis and D is the inertia product with respect to the $x-z$ axes (see Figs.1 and 2). Finally, inserting (6) into the momentum balance gives:

$$\begin{aligned} &J\frac{d^2\phi}{dt^2} + \frac{DVg}{V^2 sin(\lambda) - bgcos(\lambda)}\frac{d\phi}{dt} \\ &\quad + \frac{mg^2(bhcos(\lambda) - acsin(\lambda))}{V^2 sin(\lambda) - bgcos(\lambda)}\phi \\ &= \frac{DVb}{acm(V^2 sin(\lambda) - bgcos(\lambda))}\frac{dT}{dt} \\ &\quad + \frac{b(V^2h - acg)}{ac(V^2 sin(\lambda) - bgcos(\lambda))}T \end{aligned} \quad (12)$$

The system poles can be evaluated as:

$$p_{1,2} = \frac{\frac{-DVg}{K} \pm \sqrt{\frac{d^2V^2g^2}{K_1^2} - \frac{4Jmg^2(K_2)}{K_1}}}{2J} \quad (13)$$

or

$$p_{1,2} = \frac{-mahVg \pm \sqrt{(mahVg)^2 - 4m^2g^2h^2K_1(K_2)}}{2mh^2K_1} \quad (14)$$

where

$$K_1 = V^2 sin(\lambda) - bgcos(\lambda) \quad (15)$$

and

$$K_2 = bhcos(\lambda) - acsin(\lambda) \quad (16)$$

3 MODEL PARAMETERIZATION

In this section, the model parameters will be determined for a (2003) Trek 5200. The manufacturer provides a table of some of the critical model dimensions, as they vary with frame size, as: We can provide a simple transformation of fork rake (given by the manufacturer), R, to trail (used in the model), c, via some simple geometry as:

Table 1: Variation in bicycle parameters with frame size.

Frame size	b(m)	R	λ (deg)
50	0.979	0.0045	72
52	0.981	0.0045	72.5
54	0.985	0.0045	73
56	0.996	0.0045	73.8
58	0.996	0.0045	73.8
60	1.004	0.0045	74
62	1.008	0.0045	74

$$c = \frac{r - R/cos(\lambda)}{tan(\lambda)} \quad (17)$$

where $r = 0.35m$ for a 700c wheel. The remaining model parameters, h and a, which define the vertical and horizontal position (respectively) of the centre of mass, O, must be determined by experiment and are addressed in Sections 3.1 and 3.2 respectively. While Table 1 shows the variation for the Trek 5200, the variation in head angle and rake (which are key determinants of stability) with frame size for some other popular road bikes is shown in Fig.3.

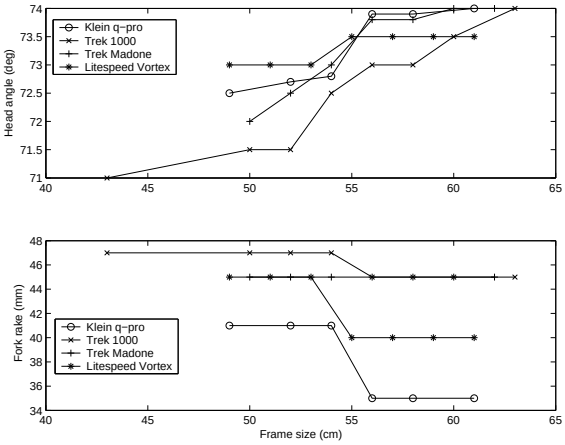


Figure 3: Typical head angle and rake variations.

3.1 Determination of a

The horizontal position of O relative to the centre of the back wheel, a, may be determined by a see-saw balance to find the horizontal position of O. The set-up is as shown in Fig.4. The bicycle, including rider, is moved back and forth in tiny increments until balance is achieved. This arrangement was also used to test the effect of an incline on A, with the following

Figure 4: Experimental determination of *a*.

Table 2: Variation in *a* with incline.

Rear wheel height (m)	*a* (m)
0	0.43
0.045	0.46
0.09	0.499
0.09*	0.501

results: The '*' denotes a condition where, with an 'incline' of 0.09 m, the rider leans forward to simulate a braking condition.

3.2 Determination of *h*

h can also be determined by a slightly more complicated arrangement, as shown in Fig.5. The bicycle, including rider, is induced to act as a pendulum, by suspension of the bicycle from an overhead fixture. For small angles of swing, the period of the pendulum is:

$$T = 2\pi\sqrt{l/g} \tag{18}$$

where l is the pendulum length (to the centre of mass). The movement of the bike/rider combination was measured by bluetooth-enabled MTx motion sensor from Xsens Motion Technologies. This allows the accurate 3 degree-of-freedom tracking of an object in motion. Fig.6 shows a section of the output from the sensor for the degree of freedom most closely aligned with the pendulum motion. The average measured oscillation period, over three trials, is 2.622 secs. This accords well with (average) stopwatch measurements of zero-crossing time of 2.617. From the measurements in Fig.5 and (18), we can determine *h* as 0.867 m. Since the mass of the bicycle + rider is dominated by the rider (80kg Vs 9kg), one would expect *O* to be roughly at the centre of mass of the rider. The mean ratio of centre of mass to height in males has been determined (Elert, 2006) as 0.565, which would put *h* roughly at the hip bone for the rider in question, consistent with the result obtained from the experimental determination. This provides a rough check on the experimental result.

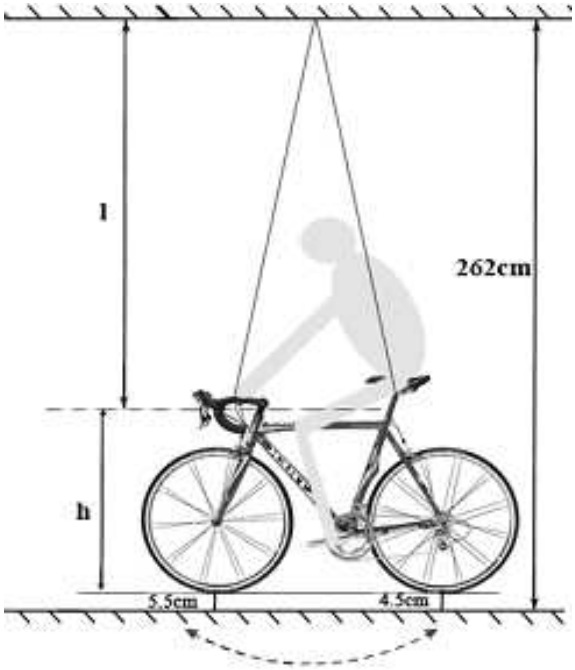


Figure 5: Experimental determination of *h*.

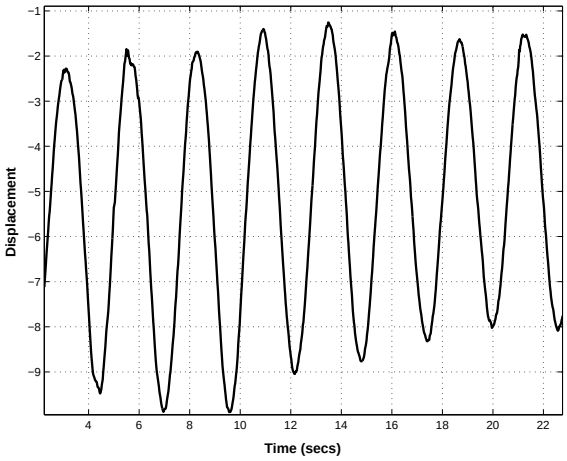


Figure 6: Pendulum period determination.

4 RESULTS

In this section, we examine the stability of the bicycle model, for the parameters determined in Section 3,

with variations in frame size and aspects which might relate to rider physique and position.

4.1 Variation in Poles with Frame Size

Using equation (14) and the data contained in Table 1, the pole variations may be plotted for a baseline condition of $a = 0.43$ as shown in Fig.7. It can be noted

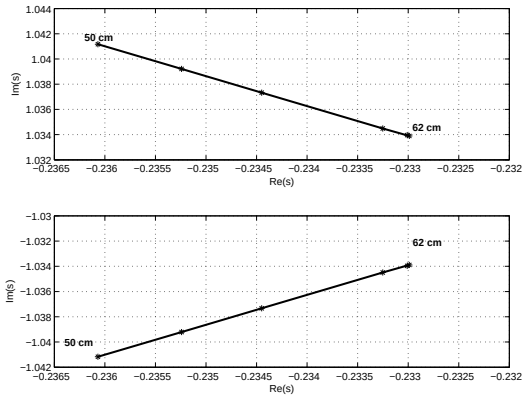


Figure 7: Pole variations with frame size.

that there is a perceptible movement of the poles towards the imaginary axis as the frame size increases. This suggests that, on the basis of frame geometry alone (not taking into account frame flexibility), there is a *decrease* in the stability of the bicycle with increasing frame size.

4.2 Variation in Poles with Velocity

Using equation (14), for a baseline condition of $a = 0.43$ and a 60cm frame, the variation in the poles with changes in velocity are computed as in Fig.8. It is clear that bicycle stability reduces as velocity increases. In the author's experience, the onset of wheel wobble typically occurs above 45 km/h. Note that the condition of basic bicycle self-stabilization in (7) is valid for the range of velocity considered in Fig.8.

4.3 Rider Position Variations

4.3.1 Horizontal Position

Again using equation (14), for a baseline condition of $V = 45km/h$ (equivalent to 12.5 m/s) and a 60cm frame, the pole variations with the relative forward position of the rider, a, can be evaluated, as shown in Fig.9. In this case, Fig.9 suggests that, as the rider moves more forwards on the bicycle, the stability of the bicycle *increases*. This would not have been the

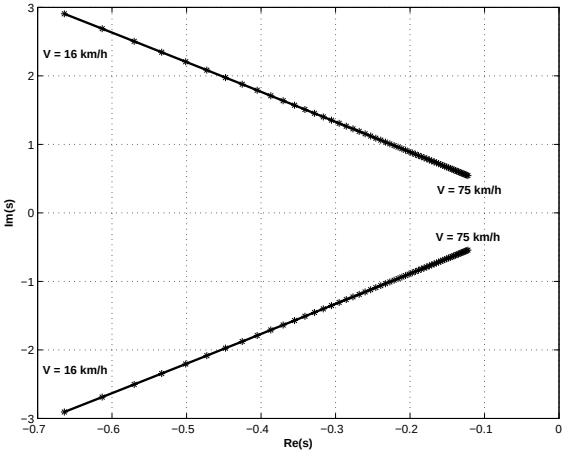


Figure 8: Pole variations with velocity.

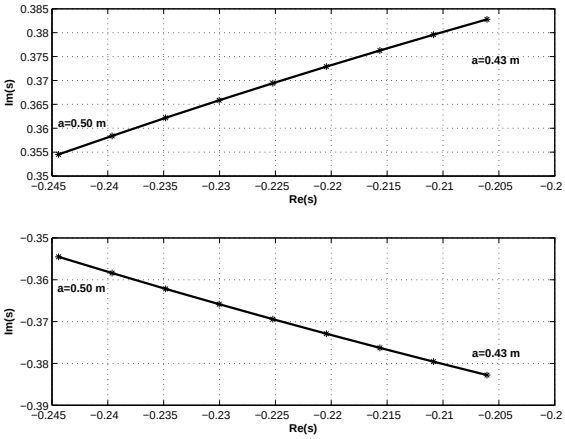


Figure 9: Pole variations with rider horiz. position.

impression of the author, but the result is supported by some anecdotal evidence presented in (Limebeer and Sharp, 2006), which documents a 'floating sensation' experienced by a motorcyclist on a record speed attempt while lying horizontal on the machine. This would, most certainly, have moved the centre of mass of the motorbike/rider combination to the rear, with a consequently relatively smaller value for a.

This result does not support the conclusion that the relative increase in a, due to the rider position moving forward under braking, causes any increased instability.

4.3.2 Vertical Position

Again using equation (14), for a baseline condition of $V = 45km/h$, a 60cm frame and a returned to its nominal value of 0.43, the pole variations with the relative vertical position of the rider, h, can be evaluated, as shown in Fig.10. From Fig.10, it is clear that increas-

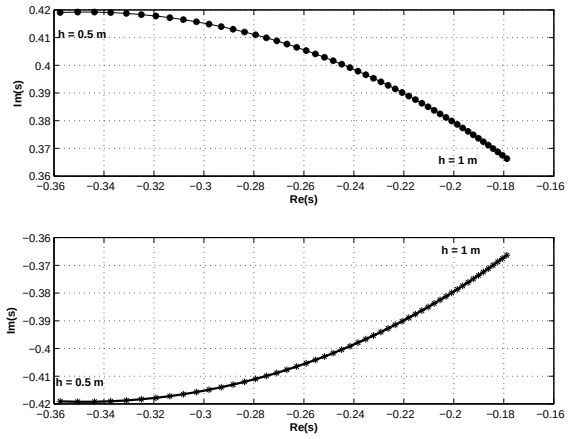


Figure 10: Pole variations with rider vertical position.

ing saddle height or a more upright position of the rider has a *destabilizing* effect on the bicycle, with a migration of the poles towards the imaginary axis.

5 CONCLUSIONS

Though the simple model in Section 2 omits some aspects of bicycle dynamics, such as frame compliance, mass of front fork assembly and rider intervention (via the torque input T), it can help to achieve an understanding of the broad effects that rider position and action has on bicycle stability. Though it is not possible to rely on absolute pole positions returned by the model, the relative pole movement under certain interventions can reveal the type of actions that can help to improve bicycle stability under certain conditions.

Clearly, road (racing) bicycles attempt to achieve a compromise between responsiveness and stability. This is largely dictated by frame geometry and trail (or fork rake). From the analysis in this paper, it appears that this compromise gives poorer stability for larger framed bicycles.

The impact of rider intervention via the torque input, T, deserves further examination. It is believed (Brandt, 2005) that attempting to reduce wheel wobble by rigidly holding the handlebars can, in fact, exaggerate it, due to the spring effect of the arms. One solution offered by an accomplished rider (who also has considerable experience in bicycle design) is to avoid holding the handlebars during fast descents (Brandt, 2006). While this is likely to alleviate stability problems due to frame shortening (as a result of braking) and exaggerated resonance via the arms, it may have it's own particular perils! The effect of rider steering action could be included as a feedback term in the model, though the parameters of such a subsystem may not be trivial to determine.

Further work should also examine the effect of frame compliance, since this is thought to be an important factor leading to wheel wobble and is likely to be more pronounced in bicycles with larger frames. However, some effects which result from component compliance can be examined within the current model structure. In particular, braking (with most of the braking effect coming from the front wheel) is likely to lead to some shortening of the wheelbase, due to flexibility in the (carbon) forks. This could cause a reduction in both b (the wheelbase) and R (the fork rake). The model predicts that a reduction in both these values would have a destabilizing effect, which could more than offset any stabilizing effect resulting from a movement forward in the centre of mass (under downhill braking), examined in Section 4.3.1.

ACKNOWLEDGEMENTS

The authors are grateful to Denis Buckley, John Maloco and Dr. Tomás Ward of the Dept. of Electronic Eng. ay NUI Maynooth for their contribution to the experimental measurements of Section 3.1 and 3.2.

REFERENCES

Astrom, K., Klein, R., and Lennartsson, A. (2005). Bicycle dynamics and control - adapted bicycles for education and research. *IEEE Control Systems Mag.*, 25:26–47.

Beghi, A. and Frezza, R. (2006). Advances in motorcycle design and control. *IEEE Control Systems Mag.*, 26:32–33.

Brandt, J. (2005). Shimmy or speed wobble. *http://www.sheldonbrown.com/brandt/shimmy.html*.

Brandt, J. (2006). Speed wobble. *Private correspondence*.

Elert, G. (2006). Centre of mass of a human. In *The Physics Factbook*. http://hypertextbook.com/facts/2006/centerofmass.shtml.

Hauser, J. and Saccon, A. (2006). Motorcycle modelling for high-performance manouvering. *IEEE Control Systems Mag.*, 26:89–105.

Klein, R. (1989). Using bicycles to teach system dynamics. *IEEE Control Systems Mag.*, 9:4.

Limebeer, D. and Sharp, R. (2006). Bicycles, motorcycles and models - single track vehicle modelling and control. *IEEE Control Systems Mag.*, 26:34–61.

Rankine, W. (1869). On the dynamical properties of the motion of velocipedes. *Engineer*, 28:79–175.

SMART DIFFERENTIAL PRESSURE SENSOR

Michal Pavlik, Jiri Haze, Radimir Vrba and Miroslav Sveda
Brno University of Technology, Udolni 53, CZ-60200 Brno, Czech Republic
pavlik.michal@phd.feec.vutbr.cz, haze@feec.vutbr.cz, vrbar@feec.vutbr.cz, sveda@fit.vutbr.cz

Keywords: Pressure measurement, ultra-low power application, current loop, microcontroller overclocking, non-linear interpolation.

Abstract: This paper presents design and assembly of mixed electronic circuitry for measured signal processing of the capacitive difference pressure sensor, as well as analysis of the obtained results. The smart pressure sensor provides values of measured pressure via 4 - 20 mA current loop output. The loop current is also used for sensor circuitry supplying. This means that current consumption of the whole sensor electronics should be less than 3.5 mA even in extended industrial temperature range from –40 to +125 °C.

1 INTRODUCTION

There are needs in some industrial branches to measure difference between two pressures. The differential measurement system is frequently used for pressure measurements because of its good temperature and time stability. The internal schematic diagram of the differential pressure sensor can be analyzed is a pair of capacitors sensing to differential pressure actual values. These capacities can be up to tens of picofarads. There is no direct measuring of capacities, but capacities of measured capacitors are converted to actual output frequency of a pair of frequency oscillators controlled by measured capacitors. The most important issue is the precision of measurement. Total accuracy is required to be better than that equivalent to 16 binary bits resolution. Therefore frequency of 255 periods of the output signal is averaged. The aim of this paper is the description of the low-power and high-precision measuring system design.

2 ELECTRONICS TOPOLOGY

The proposed electronic circuitry of the pressure sensor can be split into three modular parts. Signal processing of the differential pressure sensor is realized by a pair of oscillators whose output frequencies reflect the value of the measured pressure. Consequently, galvanically separated part including microcontroller converts the output frequency values of the oscillators to digital code values. Besides, embedded microcontroller calculates non-linear correction of the measured values and temperature calibration at the same time. The output quantity of this part of electronic circuitry is a digital calibrated value of pressure. According to the desired extended temperature range from -40 to + 125 °C of the proposed sensor, the outputs of the oscillators are carried by signal transformers.

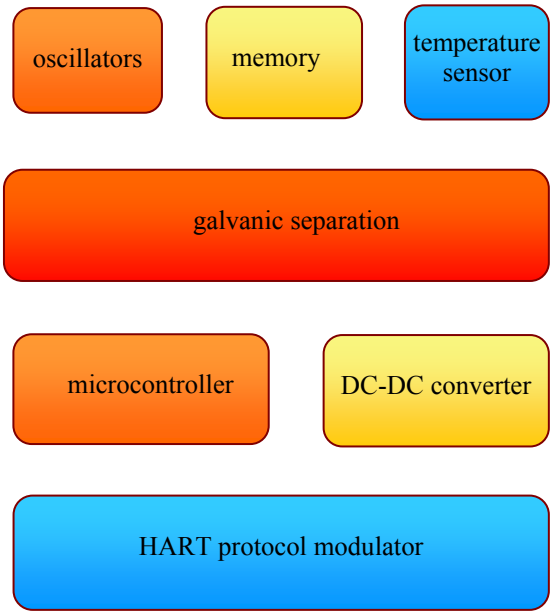


Figure 1: Block diagram of system topology.

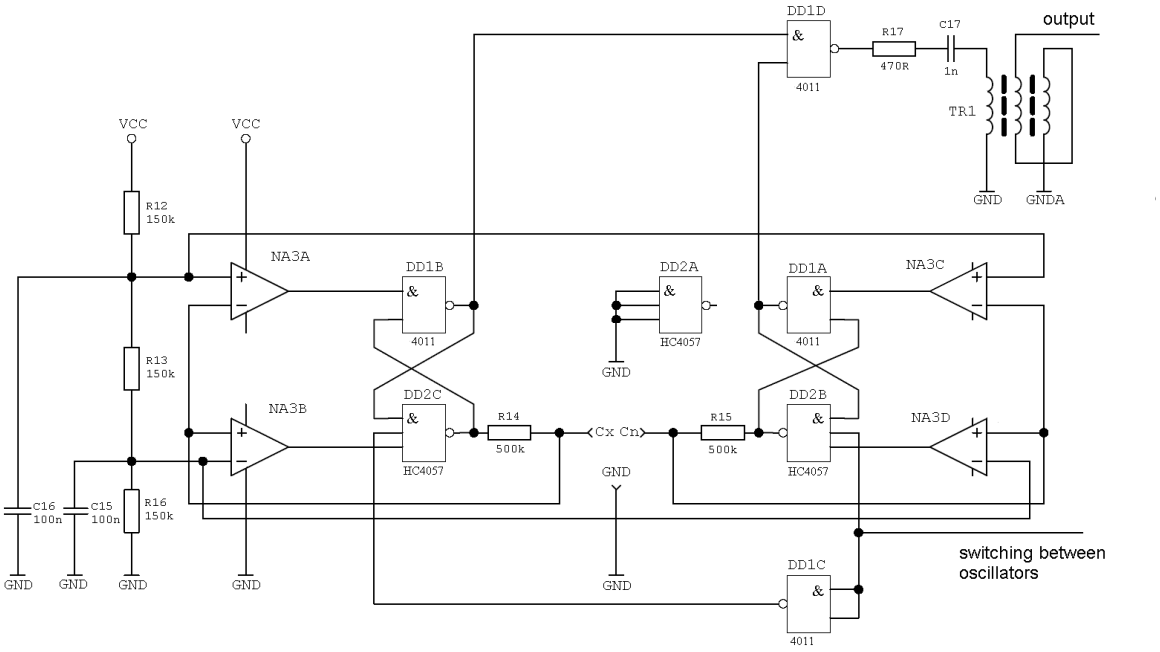


Figure 2: Simplified schematic diagram of galvanically separated oscillators.

The microcontroller controls galvanically separated DC-DC changer that supplies oscillators, too. The block diagram of the electronic circuitry topology is shown in Fig. 1. A microcontroller is included in the third part of electronics. The microcontroller is mainly used for HART modulation of the loop current communication. The second function of the microcontroller is active regulation of the actual current in the current loop by means of sensor consumption supplying current control. Interconnection between the second and third stage of electronics is provided by the SPI bus. These two parts are galvanically connected.

2.1 Oscillators

Even if oscillators are based on the two basic 555 circuits, there are a few circuitry modifications. Only one of oscillators is running during actual running phase of the measurement process. It results in decreasing power consumption to nearly 65% of the original one. The ultra low power and fast comparators MAX939 are used. The crucial parameters of the comparators are slew rate and transfer time delay. The application of these comparators represents the best solution in terms of power consumption and speed ratio. The precision of the measurement mainly depends on the reaction time of the comparators or possibly on the spread of the overshoot from the reference voltage. The simplified schematic diagram of the oscillators is shown in Fig. 2. Output signal of the running oscillator is led via serial combination of the capacitor and resistor to a primary winding of a signal transformer. Serial resistor limits flowing surge current when the logic output is changed. Unfortunately, restriction of an exciting current leads to extension of the rising and falling edge of the transmitted signal. Serial capacity prevents bias direct current from passing the transformer, thus protects the transformer against overloading. The output frequency of the oscillators can be calculated using a simple equation

$$f_{out} = \frac{2.R}{C}, \qquad (1)$$

where R is value of reference resistor 500 kΩ and C represents the measured capacity.

2.2 DC-DC Changer

The DC-DC changer with a transformer was designed to supply the oscillators. The transformer provides galvanic separation. In reality, construction of the switched changer was the only one possible solution and efficiency better than 50 % was achieved. The circuitry of the changer consists of a minimum component and is driven by an embedded microcontroller. Unfortunately, the feedback cannot

be used because it leds to increased power consumption.

2.3 Measurement Principle

The measurement is based on counting of 255 periods of the measured signal. Microcontroller system clock is used as a sampling signal. Quiescent frequency of the oscillators is set to 4.5 kHz. The microcontroller counts 255 periods in 56 ms. Thus total measurement time is 112 ms. These calculations are not correct because the pair of oscillators are not really identical, but even if real measurement can be faster or slower, complete measurement time is constant. This attribute is given by design of the differential pressure sensor. The measurement algorithm is implemented in the microcontroller as follows: Counter/Timer0 (C/T0) is configured as an 8-bit counter (it means 255 period of input signal). The Counter/Timer1 (C/T1) runs as a 16-bit timer with 125 kHz clock before the counting is allowed. The low system frequency of the microcontroller significantly reduces power consumption [3]. But minimal 1 MHz of the system frequency is needed to suppose desired measurement accuracy. Due to when 253 periods are counted the microcontroller is over-clocked to 2 MHz. The value in C/T1 is stored for next processing and C/T1 is cleared. When the 255 periods are counted, the interrupt is called and value in C/T1 is red. This value reflects the measured capacity. With no pressure the counter counts approximately 112 000 pulses from each oscillator. By using equation

$$n = \frac{\log x}{\log 2}, \qquad (2)$$

where x represents numbers of levels and n is a bit resolution, we can calculate that we can measure oscillator frequencies with more than 16-bit resolution.

This accuracy is adequate. For effective processing of the measured values, the working variable *A(p)* is evaluated. Variable *A(p)* represents uncorrected digital pressure

$$A(p) = \frac{f_1 - f_2}{f_1 + f_2}, \qquad (3)$$

where f_1 and f_2 are measured frequencies of oscillator output signals. At next stage the working variable *A(p)* is calibrated using non-linear corrections by hi-order polynomial. The calibration provides linear response of the output value to the pressure. The calibrated output value presents the digital pressure and is set in specified units (bar, kPa, etc.). After all linearization and calibration processes the value is sent via SPI to the second microcontroller which provides transmitting into the current loop.

2.4 Corrections

Two corrections are calculated by the embedded microcontroller. At first, the linearization, offset calibration and gain correction are calculated. Next, the temperature dependence of the measuring electronics is compensated. Fig. 3 shows enumerated dependencies in a 3D graph.

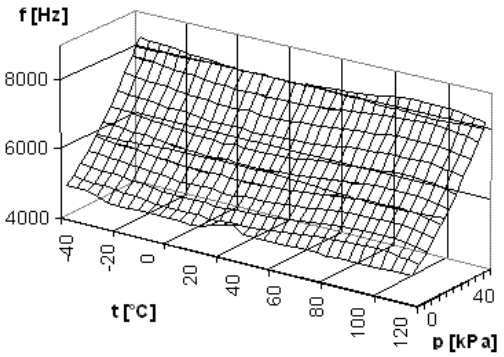


Figure 3: The oscillator output frequency dependence on pressure and temperature.

There are a few calibration methods for example lookup tables but these methods are usually of a high cost and time consuming [2]. The polynomial of fifth to eighth order is used for calibration of the variable *A(p)*. The basic form of the polynomial is

$$y = a_0 + a_1 x + a_2 x^2 + .. + a_n x^n \qquad (4)$$

The Lagrange's polynomial is used for calculation of the calibration constants. The Lagrange's polynomial is the lowest order polynomial which goes through specified values [4]. The Lagrange's polynomial can be calculated by

$$\sum_{i=1}^{n} f(x_i)\lambda_i \qquad (5)$$

where

$$\lambda_i = \frac{(x-x_1)(x-x_2)..(x-x_{i-1})(x-x_{i+1})..(x-x_n)}{(x_i-x_1)(x_i-x_2)..(x_i-x_{i-1})(x_i-x_{i+1})..(x_i-x_n)}. \qquad (6)$$

The calibration data is stored in FRAM embedded on the oscillator board.

2.5 HART Protocol

For communication over the 4 - 20 mA current loop, the HART protocol is used [1]. Signal current modulation is provided by the second microcontroller. Transmitting is done using controlled loading. The regulated loading circuitry is very simple and consists of an NPN type bipolar transistor with a grounded emitter and a driving DA converter. Current consumption is minimized thanks to simplicity of the regulated loading.

3 RESULTS

After design, assembly and programming of the microcontroller real measurements were done. The frequencies of the oscillators, working variable A and digital pressure values were logged. These values were logged for many different pressures over the whole sensor range. From the measured data the bias noise was figured out by equation

$$N_f = \frac{\Delta N_{\max}}{N_{\max} - N_{\min}}, \qquad (7)$$

where $\Delta N_{\max}$ represents the maximal deviation from the mean value of a few samples for a specified pressure in the whole measuring range, $N_{\max}$ is value of the output with maximal pressure and $N_{\min}$ is value of the output with no pressure.

The bias noise in the whole measuring range was only 0.82 ‰. By conversion of the bias noise to the bit resolution the 13.57 effective bit resolution was achieved. The linearity degree of the working variable which determines order of the correction polynomial is very important. The dependence of the variable $A(p)$ on pressure is shown in Fig. 4.

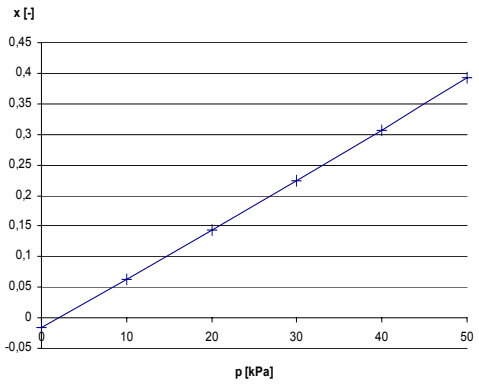


Figure 4: Dependence of the measured output on the pressure.

We can observe deviations of the measured waveform in Fig. 5.

And finally, deviation of the corrected waveform is shown in Fig. 6, after calculating of the Lagrange polynomial constants and their application from the linear waveform.

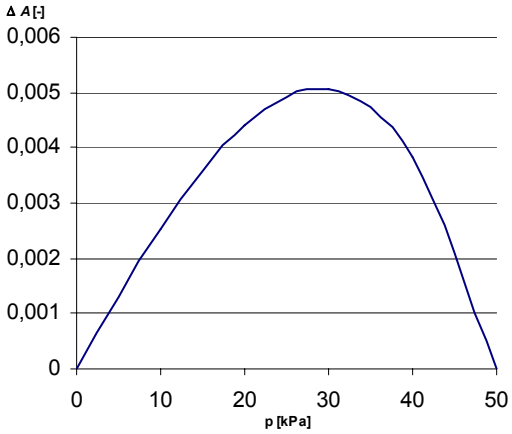


Figure 5: Deviations of the measured waveform.

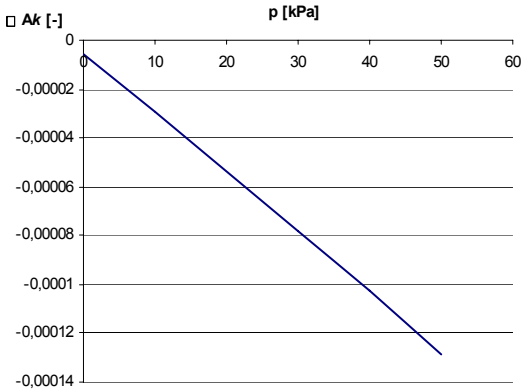


Figure 6: Deviations of the calibrated waveform.

4 CONCLUSIONS

A smart differential capacity pressure sensor was designed and assembled. The system consists of three parts – oscillators, processing microcontroller and HART modulator. Ultra low-power devices and special measuring algorithm in microcontroller were used to reduce power consumption bellow 3.5 mA. The Lagrange polynomials were applied to calculate

the measured values calibration. It improves linearity more than ten times.

ACKNOWLEDGEMENTS

The research has been supported by the Czech Ministry of Education within the framework of the Research Program MSM0021630503 MIKROSYN, by the Czech Grant Agency in projects GACR 102/03/0619 and GACR 102/03/H105, and by the Ministry of Industry and Commerce in projects FF-P/112 and FT-TA/050.

REFERENCES

HART communication foundation (2007) *HART specification*, http://www.hartcomm2.org/hart_protocol/protocol/hart_specifications.html

Kouider, M. Nadi, M. and Kourtiche D. (2003) *Sensors Auto-calibration Method - Using Programmable Interface Circuit Front-end*, SENSORS 2003, ISSN 1424-8220

Holberg, A.M. and Seatre A. (2006) *Innovative Techniques for Extremely Low Power Consumption with 8-bit Microcontrollers*, ATMEL White Paper

Mori, H. and Yamada S. (2003) *Continuation Power Flow with the Nonlinear Predictor of the Lagrange's Polynomial Interpolation Formula*, IEE Japan

A KALMAN FILTERING APPROACH TO ESTIMATE CLAMP FORCE IN BRAKE-BY-WIRE SYSTEMS

Stephen Saric and Alireza Bab-Hadiashar
Faculty of Engineering and Industrial Sciences, Swinburne University of Technology, John Street, Hawthorn, Australia
ssaric@swin.edu.au, abab-hadiashar@swin.edu.au

Keywords: Brake-by-wire, sensor fusion, dynamic stiffness, torque balance, optimisation.

Abstract: Removing a clamp force sensor from brake-by-wire (BBW) system designs has been driven by the need to reduce costs and design complexities. In this paper an improved method is presented to estimate clamp force using other sensory information. The proposed estimator is based on the Kalman filter where the actuator resolver is used in a dynamic stiffness model and the actuator current sensors as well as the resolver are used to give measurement updates in a torque balance model. Experimental results show that the estimator can handle highly dynamic braking scenarios making it suitable for possible use in anti-lock braking system (ABS) controls. A comparison is made with a previous attempt to estimate clamp force in BBW systems and it is shown that the proposed estimator improves the root mean square error (RMSE) of estimation. A training strategy is explained to ensure that the estimator can adequately adapt to parameter variations associated with wear. This paper finally discusses reliability issues associated with the developed clamp force estimator.

1 INTRODUCTION

Drive-by-wire (DBW) technologies are being currently developed and introduced into the automotive industry. One advantage of such technologies is to produce intelligent vehicle control systems that improve performance by benefiting from the integration of electronic systems (Schenk et al. 1995). The subject technology of interest in this paper is BBW systems for disk brakes. Figure 1 shows a schematic diagram of a BBW system as given by Saric et. al. (2007). A pedal feel emulator provides the human-machine interface in a BBW system. This pedal is fitted with sensors whose outputs are processed by an electronic control unit which then controls the actuators.

An electric motor that is coupled to reduction gearing is the general setup used for an electro-mechanical brake (EMB) actuator. The motor is normally of a permanent magnet brushless DC type for the reasons of compactness and enhanced commutation efficiency. A planetary gear-set connected to a ball-screw are generally the components used in the reduction gearing.

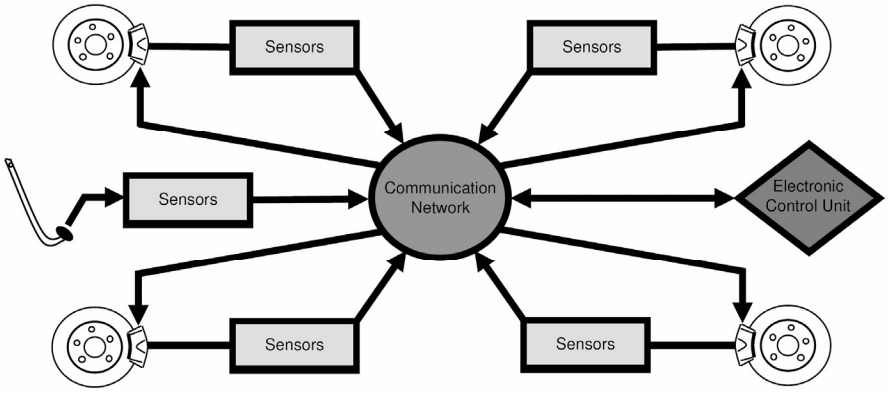


Figure 1: BBW System.

To control EMB caliper clamping force, a clamp force sensor is typically used to close the control loop. A standard motion control architecture (cascaded position, velocity and current control loops) which is slightly altered can be used to control an EMB. Line et al. (2004) exchange the position control loop for a force control loop for EMB control purposes. This architecture is shown in figure 2 as given by Hoseinnezhad et. al. (2006). The control system depicted in figure 2 requires the use of a displacement sensor, normally a resolver, and three motor current sensors for a three phase brushless DC motor.

The adequate implementation of a clamp force sensor in an EMB system can be a difficult thing to achieve. If a clamp force sensor is placed near to a brake pad, it must then be able to mechanically withstand the high temperatures (up to 800 ^{O}C) it will be subject to. Also temperature drifts may need to be compensated for. By embedding a clamp force sensor deep within a caliper, i.e. at the near end of the ball-screw this situation can be avoided. However due to the effects of friction between the embedded clamp force sensor location and an inner pad, a hysteresis effect results which prevents a true clamp force to be sensed. A clamp force sensor is a costly item in an EMB caliper. This is due to a high supplier cost and increased production expenses due to its inclusion. These high production costs result from online calibration for each individual sensor because of performance variability from one unit to another, as well as difficult assembly procedures due to the small tolerances being dealt with.

The elimination of a clamp force sensor from EMB designs is highly desirable because of the cost issues and engineering challenges involved with its use. A way to eliminate this component may be realized via a sensor fusion approach, that is, to estimate clamp force using remaining EMB system sensors.

The introduction given here is followed by a developmental background that briefly explains previous works completed on estimating clamp force in EMB systems. A description of the developmental steps taken to attain our new clamp force estimator is then provided, followed by describing the test rig which we have employed. Validation results are given which then finally leads to conclusive remarks.

2 BBW CLAMP FORCE ESTIMATION REVIEW

Developed torque in an EMB caliper can be determined from motor current sensors which are part of all EMB designs. A simplified model says that the torque induced by a permanent magnet DC motor is linearly related to the current passing through the field coil, that is:

$$T_m = K_m I_m \tag{1}$$

where T_m, I_m and K_m are the motor torque, the field current and the motor torque constant respectively. The latter term is a constant that is experimentally determined. For a brushless permanent magnet DC motor the current (I_m) is the quadrature component of the resultant current space vector as found from the individual phases (Krishnan 2001, p. 527). Since the motor torque input in an EMB caliper causes a clamping force, it is apparent that a relationship must exist between these two variables. To determine an induced clamp force in an EMB caliper using motor current information, a torque balance

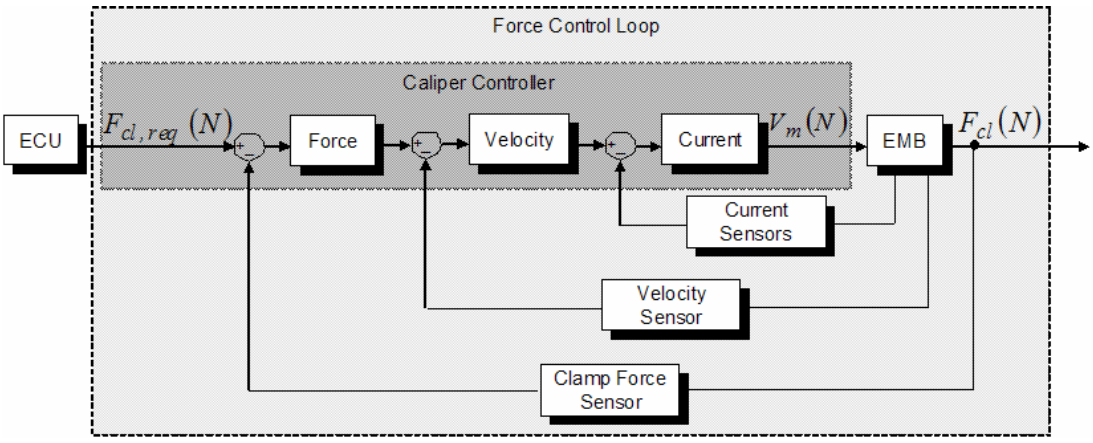


Figure 2: EMB system control architecture.

can be solved as follows:

$$\begin{aligned} T_m &= T_a + T_i + T_f \\ I_m K_m &= \gamma_{tot} F_{cl} + J_{tot}\, d^2\theta_m / dt^2 + T_f \\ F_{cl} &= (I_m K_m - J_{tot}\, d^2\theta_m / dt^2 - T_f) / \gamma_{tot}\,. \end{aligned} \qquad (2)$$

The torque balance says that the torque developed by the motor (T_m) equals the torque required to provide clamping force (T_a), to meet the necessary inertial demands (T_i) and to overcome frictional resistance (T_f). By combining the load ratios from a series connected planetary gear-set and ball-screw, it can be found that this value (γ_{tot}) acts as a gain relating the application torque (T_a) to the clamp force (F_{cl}). The entire caliper inertia (J_{tot}) is lumped and involves both rotational and translational motions. This value is usually attained using empirical data where an energy balance, over a stage of motor acceleration, is formulated to find the lumped inertia (J_{tot}).

Equation (2) shows that the frictional torque (T_f) term is undefined. The reason for this is that as Olsson et al. (1998, p. 176) explain, using theoretical friction models for practical purposes is difficult to achieve in a satisfactory manner. To overcome this problem, theoretical friction models should be merged with experimentally established phenomena unique to a particular system. Friction models of any sort tend to be avoided in trying to estimate clamp force in an EMB caliper because of the problems in trying to account for wear in the reduction gearing. This subject will be continued in later discussion.

A clamp force estimation algorithm was developed by Schwarz et al. (1999) for use on an EMB caliper designed for a disk brake. Equation (2) was involved in part within their algorithm. By employing a differing technique they avoid the need for using a friction model which is explained in more detail as follows. A low amplitude high frequency sinusoid is superimposed on the otherwise normal angular motion from the motor. This forces the motor to pass the same angular position in a finite length of time between a clamping and releasing action. At both these instants the application of (2) yields:

$$T_{m,cl} = \gamma_{tot} F_{cl} + J_{tot}\, d^2\theta_{m,cl} / dt^2 + T_f \qquad (3)$$

$$T_{m,rl} = \gamma_{tot} F_{cl} + J_{tot}\, d^2\theta_{m,rl} / dt^2 - T_f \qquad (4)$$

where the subscripts *cl* and *rl* indicate clamping and releasing respectively. The friction terms in (3) and (4) have approximately the same magnitudes but opposite signs due to the change in course of motor travel. Adding (3) and (4) cancels out the friction terms and after some manipulation the following equation to estimate clamp force (F_{cl}^*) can be found:

$$F_{cl}^{*} = (T_{m,cl} + T_{m,rl} - J_{tot}\, d(\theta_{m,cl} + \theta_{m,rl}) / dt^2) / (2\gamma_{tot}). \qquad (5)$$

Passing the same motor angle via sinusoidal differing becomes a harder and harder task to achieve as the clamp force application rate is increased. Also the requirement of reversing direction in a short period of time during increased clamp force application rates will most likely challenge the dynamic control ability of the EMB system. A means to cope with these problems is proposed by Schwarz et al. (1999). The characteristic curve of an EMB caliper is a relationship between motor angle and applied clamp force where the former is varied in a pseudo-static fashion. Figure 3 displays this curve for an EMB caliper as given by Hoseinnezhad et. al. (2006). Schwarz et al. (1999) put forward the use of a caliper characteristic curve to provide feedback control of applied clamp force. When the opportunity to use (5) arises, it is done so with the intentions of adapting the parameters in the characteristic curve due to pad wear.

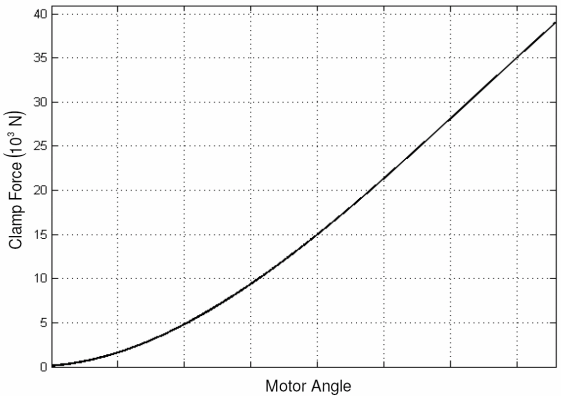


Figure 3: Characteristic curve for an EMB Caliper.

As provided by Hoseinnezhad et al. (2006), figure 4 displays clamp force versus motor angle for a highly dynamic situation where the motor angle is varied in a uniform random fashion with a sample time of 100 ms. It is clear that considerable dynamic exists within the system and that the use of a characteristic curve for clamp force estimation purposes has its limitations for highly dynamic scenarios. The cause of this dynamic is attributed to viscoelastic effects exhibited mainly by the caliper bridge. Hoseinnezhad et al. (2006) developed a dynamic stiffness model to handle such viscoelastic

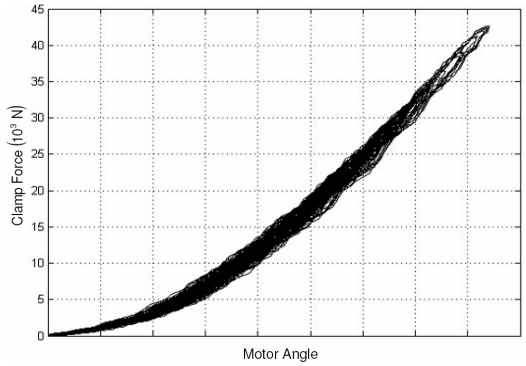


Figure 4: Clamp force versus motor angle for highly dynamic case.

effects. This model is given as follows in discrete time notation:

$$F_{cl}^{*}(k) = \alpha_3\theta_m^3(k) + \alpha_2\theta_m^2(k) + \alpha_1\theta_m(k) + \alpha_0 F_{cl}^{*}(k\text{-}1) \tag{6}$$

where α_4, α_3, α_2, α_1 and α_0 are experimentally determined constants and θ_m is the motor angle.

Saric et al. (2006, 2007) uses (6) as well as a second model to estimate clamp force in a fusion algorithm which optimizes the RMSE of estimation. The second model is based on the torque balance approach where a dynamic Coulomb friction model is used which is dependent on clamp force and is shown below in discrete time notations:

$$F_{cl}^{*}(k) = \frac{T_m(k) - \frac{J_{tot}}{t_s^2}(\theta_m(k) - 2\theta_m(k-1) + \theta_m(k-2)) - A_k \operatorname{sgn}(\theta_m(k) - \theta_m(k-1))}{\gamma_{tot} + \mu_k \operatorname{sgn}(\theta_m(k) - \theta_m(k-1))}. \tag{7}$$

where t_s, μ and A are the sampling time, the coefficient of Coulomb friction and an offset friction term respectively. The two models given by (6) and (7) are fused together by Saric et al. (2006, 2007) using a maximum likelihood estimator to give an optimized estimate of clamp force which is as follows:

$$\hat{F}_{cl}(k) = F_{ds}^{*}(k) + \frac{\sigma_{ds}^2}{\sigma_{ds}^2 + \sigma_{tb}^2}(F_{tb}^{*}(k) - F_{ds}^{*}(k)) \tag{8}$$

where σ is the standard deviation and the subscripts *ds* and *tb* indicate dynamic stiffness and torque balance respectively. Gaussian noises were assumed in the derivation of (8). After having adapted parameters, as detailed by Saric et. al. (2007) and described previously, an improvement in the RMSE of approximately 10% is obtained as a result of fusing via (8). Parameters are adapted in (6) due to stiffness variations because of pad wear, and in (7) because of frictional variations in the caliper reduction gearing.

The fusion algorithm used by Saric et al. (2006, 2007) does not have a recursive nature. Therefore the use of a Kalman filter will further improve estimation accuracy (Kalman 1960, p. 35; Sorenson 1970, p. 63). In this paper we present the use of a Kalman filter to estimate clamp force. The ensuing section details how we setup the Kalman filter for clamp force estimation purposes in a BBW system.

3 KALMAN FILTER SETUP

A Kalman filter is a linear, recursive, discrete time estimation algorithm. It is maximum likelihood in nature in that the RMSE's are minimized. A Kalman filter is implemented widely in control systems to give improved system state estimates. Figure 5 shows a block diagram representation of a Kalman filter in a control system. The use of a Kalman filter is advantageous because of noise influences (Gaussian) which render the true system states unknown. A Kalman filter uses system dynamics as well as other measurement sources to estimate states. Typically the later is attained from direct sensory measurements. The noises which affect both kinds of estimates the Kalman filter receives, view figure 5, are required to be uncorrelated.

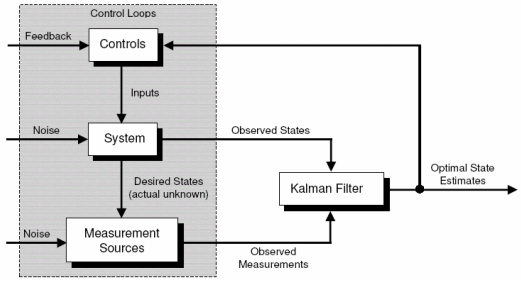


Figure 5: Typical Kalman filter application.

In the case where the system dynamics and/or measurement dynamics (which acts on the system dynamic estimates) is non-linear, an Extended Kalman filter (EKF) can be used which performs a linearization procedure. This is not necessary for our purposes due to the linear nature of the circumstances. Figure 6 shows a block diagram representation of a Kalman filter where:

x - is the system state vector

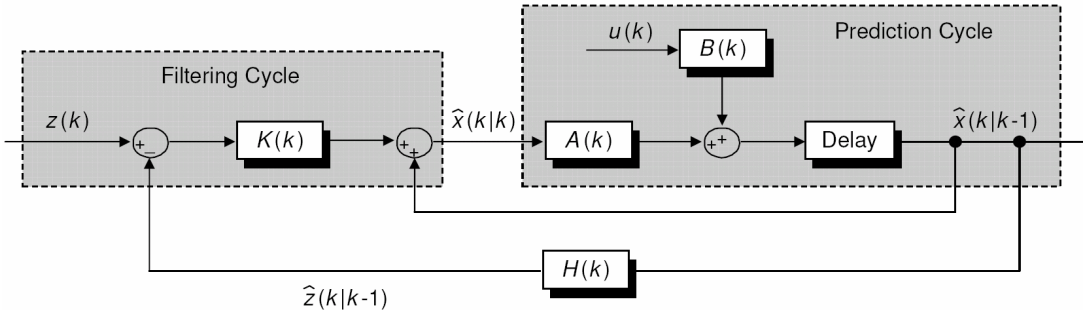


Figure 6: Block diagram representation of a Kalman filter.

u - is the control inputs
z - is the observation vector
K - is the filter gain
A - is the coefficient matrix of the system
B - is the driving matrix
H - is the measurement matrix.

The hat (^) scripts in figure 6 denote the criteria of trying to minimize the RMSE of estimation. The discrete time notation used in figure 6, $k|k$-1, indicates that the estimate at k was determined given knowledge at k-1. The filter gain is defined as follows:

$$K(k) = P(k|k\text{-}1)H^T(k)(H(k)P(k|k\text{-}1)H^T(k) + R)^{-1} \quad (9)$$

where,

P - is the covariance matrix of state estimates
R - is the measurement noise covariance matrix.

The matrices ($P(k|k\text{-}1)$) and ($P(k|k)$) for a Kalman filter are given below as:

$$P(k|k\text{-}1) = A(k)P(k\text{-}1|k\text{-}1)A(k)^T + Q \quad (10)$$
$$P(k|k) = (I - K(k)H(k))P(k|k\text{-}1) \quad (11)$$

where,

Q - is the system noise covariance matrix
I - is the identity matrix.

It should be noted that the system noise covariance (Q) and measurement noise covariance (R) matrices may be time-variant, however we assume them here to be constant. As shown in figure 6, a Kalman filter of any type involves the recursive application of prediction and filtering cycles. Brown and Hwang (1992) give a complete derivation of the Kalman filter algorithm.

To employ a Kalman filter for clamp force estimation in a BBW system, we firstly use (6) as our state space system equation. The constant α_0 from (6) is taken to be equal to $A(k)$ from figure 6. Note that typical matrix notations were not required due to the unit state space dimension of (6). The clamp force in (6) is non-linearly proportional to the motor angle input. This non-linearity does not require the use of an EKF because it is not state dependent. To integrate this non-linearity within the Kalman filtering algorithm given in figure 6 we apply the following equality:

$$B(k)u(k) = \alpha_3\theta_m^3(k) + \alpha_2\theta_m^2(k) + \alpha_1\theta_m(k) \quad (12)$$

We take the equivalent of $\hat{x}(k \mid k\text{-}1)$ from figure 6 to be directly equal to $\hat{z}(k \mid k\text{-}1)$, that is:

$$\hat{z}(k \mid k\text{-}1) = \hat{F}_{cl}(k \mid k\text{-}1). \quad (13)$$

For this situation $H(k)$ is taken to have a constant unit value. We use (7) as our source for measurement updates which is equivalent to $z(k)$ from figure 6. Saric et al. (2007) found that the RMSE's associated with (6) and (7), after having adapted parameters, were 0.35 and 0.61 kN respectively. These values squared are used for assumed constant system (Q) and measurement (R) error variances (the typical term covariance has not been used since (6) has a unit state space dimension).

With the Kalman concept to estimate clamp force in BBW systems defined, the next section briefly describes the test rig required to obtain necessary data for analysis and subsequent validation purposes.

4 EXPERIMENTAL ENVIRONMENT

A test rig was setup for use on a prototype EMB caliper. An external servo motor was used to provide actuation by coupling it to the caliper internal reduction gearing as shown in figure 7. The external motor is of the permanent magnet brushless type, with ratings of 55.5 N.m and 5, 000 rpm and ensures that maximum clamp forces can be achieved. To interface with this motor, the RS232 protocol was utilized. MATLAB's Simulink package along with the xPC block-set provided a real time operating system that was implemented to control the external

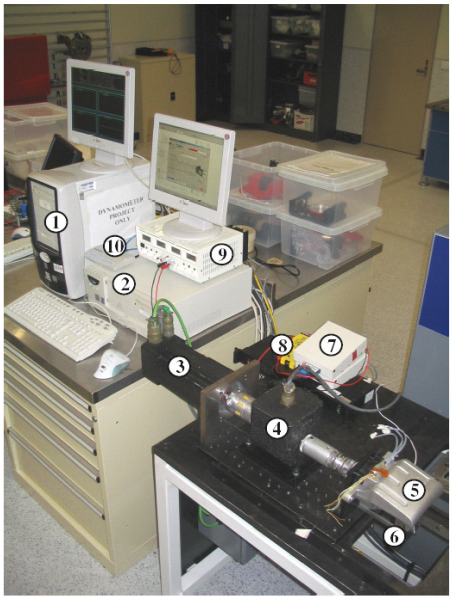


Figure 7: Test rig using a brushless permanent magnet external servo motor.

1. *host PC*
2. *target PC*
3. *brushless permanent magnet external servo motor*
4. *external torque sensor*
5. *EMB caliper*
6. *external clamp force sensor*
7. *National Instruments brake-out boxes*
8. *low pass filter/amplifier for external clamp force sensor*
9. *DC power supply*
10. *ethernet hub*

motor angle. The external motor is controlled by PID controllers within a standard motion control architecture; cascaded position, velocity and current control loops as illustrated in figure 8. Sensory information are logged by uploading the signal data to the host PC from the target PC, marked 1 and 2 respectively in figure 7. The logged data is stamped at 100 μs time-step intervals. Both the host and target PC's have Pentium 4 processors operating at 2.4 GHz. To measure the caliper motor angle, an encoder output is taken from the 1:1 coupled external servo motor. The resolution of this encoder output provides 8, 192 counts per revolution. An external torque sensor is used to sense torque input to the EMB caliper. An external clamp force sensor is used to measure the true load induced by the brake pads.

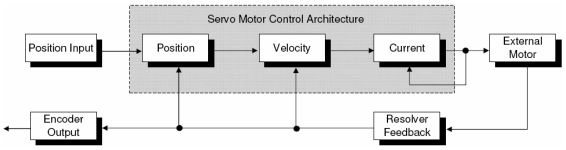


Figure 8: Control scheme used for external servo motor.

For the reasons of clarity the external motor angle and torque data from this test rig is considered to be received from an EMB caliper itself since a resolver and current sensors are available.

5 RESULTS AND DISCUSSIONS

After having adapted parameters in (6) and (7) as detailed by Saric et. al. (2007) and described previously, the Kalman filter setup given earlier for clamp force estimation purposes in a BBW system are applied to uniform random data. The uniform random data involves varying the motor angle in a uniform random manner with a sample time of 100 ms. We use the constant system (Q) error variance to initialize the clamp force estimate error variance at time equal to zero.

Figure 9 shows the performance of our new method to estimate clamp force in a BBW system. This result shows that adaptation to ABS controls is possible seeing as the actuation speeds are comparable. A new RMSE of 0.29 kN results which is an approximately 20 % improvement on the RMSE from the dynamic stiffness model alone. Saric et al. (2007) found that the use of a maximum likelihood estimator with no recursive aspect, as given by (8), gave a RMSE in clamp force estimation of 0.32 kN where the same experimental setup and control input was used as here. Therefore we have demonstrated that the use of an Kalman filter which has a recursive aspect, improves the

RMSE of clamp force estimation by approximately 10 % with regards to the methods used by Saric et. al. (2007).

With in-service pad temperatures being able to possibly reach 800 ^{O}C, it has been found that the stiffness of pads varies depending on the temperature (Schwarz et. al. 1998). Stiffness is a large component within the clamp force estimator developed within this paper. The brake pad temperature was kept constant using the static test rig shown in figure 8, and hence temperature effects on clamp force estimation under practical circumstances should be investigated.

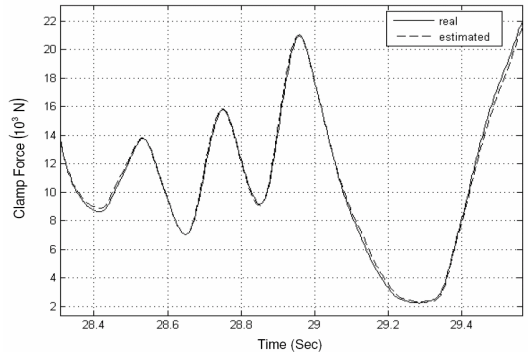


Figure 9: Uniform random data, 100 ms sample time, EKF clamp force estimator validation.

6 CONCLUSIONS

This paper presents the use of a cost effective and design friendly solution for an automotive BBW actuator. The objective of making a clamp force sensor a redundant component in an EMB system is strongly encouraged by the results within this paper. A dynamic stiffness model was used to estimate clamp force which relied on the output from an internal resolver. Based on a torque balance approach, a second model was used to estimate clamp force which relied on the use of internal motor current sensors and an internal resolver. Wear dependent parameters from both models were adapted using an in-service method. The outputs from the two independent models were fused using a Kalman filter to give optimized estimates of clamp force. The developed estimator has been shown via experimental verification to be able to handle highly dynamic braking situations. Also it has been shown that the RMSE of estimation with regards to previous attempts to estimate clamp force in BBW systems has been improved upon. With continued development the possible cost savings inherent with attempting to make a clamp force sensor redundant can be accomplished in future EMB designs.

ACKNOWLEDGEMENTS

The initiative formed by the centre of Research for Advance By-Wire Technologies (RABiT) provided a medium for which this collaborative work was undertaken by Swinburne University of Technology (SUT) and PGT. The authors of this paper would like to thank the engineers from PGT for their kind assistance.

REFERENCES

Brown, R.G., Hwang, P.Y.C., 1992. *Introduction to Random Signals and Applied Kalman Filtering*, John Wiley & Sons. United States, 2nd edition.

Hoseinnezhad, R., Saric, S., Bab-Hadiashar, H., 2006. Estimation of Clamp Force in Brake-by-Wire Systems: A Step-by-Step Identification Approach, *SAE Technical Paper Series*, no. 061154.

Kalman, R.E., 1960. A New Approach to Linear Filtering and Prediction Problems, *ASME Journal of Basic Engineering*, vol. 82, ser. D, pp. 35-45.

Krishnan, R., 2001. *Electric Motor Drives*, Prentice Hall, New Jersey.

Line, C., Manzie, C., Good, M., 2004. Control of an electromechanical brake for automotive brake-by-wire systems with adapted motion control architecture, *SAE Technical Paper Series*, no. 042050.

Olsson, H., Åström, K.J., Wit, CCd., Gäfvert, M., Lischinsky, P., 1998. Friction models and friction compensation, *European Journal of Control*, vol. 4, no. 3, pp. 176-195, 1998.

Saric, S., Bab-Hadiashar, A., Hoseinnezhad, R., 2007. A Sensor Fusion Approach to Estimate Clamp Force in Brake-by-Wire Systems, *IEEE Transactions on Vehicular Technologies*, accepted for publication.

Saric, S., Bab-Hadiashar, A., Hoseinnezhad, R., 2006. A Sensor Fusion Approach to Estimate Clamp Force in Brake-by-Wire Systems, *IEEE Vehicular Technologies Conference*, no. 27448.

Schenk, D.E., Wells, R.L., Miller, J.E., 1995. Intelligent Braking for Current and Future Vehicles, *SAE Technical Paper Series*, no. 950762.

Sorenson, H.W., 1970. Least-squares estimation: from Gauss to Kalman, *IEEE Spectrum*, vol. 7, pp. 63-68.

Schwarz, R., Isermann, R., Böhm, J., Nell, J., Rieth, P., 1999. Clamping force estimation for a brake-by-wire actuator, *SAE Technical Paper Series*, no. 990482.

Schwarz, R., Isermann, R., Böhm, J., Nell, J., Rieth, P., 1998. Modelling and Control of an Electromechanical Disk Brake, *SAE Technical Paper Series*, no. 980600.

ROBUST AND STABLE ROBOTIC FORCE CONTROL

Michael Short[1] and Kevin Burn[2]
[1]*Embedded Systems Laboratory,University of Leicester, UK.* [2]*Control Systems Centre, University of Sunderland, UK*
mjs61@le.ac.uk, kevin.burn@sunderland.ac.uk

Keywords: Robotic force control, model following control, stability.

Abstract: To perform many complex tasks, modern robots often require robust and stable force control. Linear, fixed-gain controllers can only provide adequate performance when they are tuned to specific task requirements, but if the environmental stiffness at the robot/task interface is unknown or varies significantly, performance is degraded. This paper describes the design of a robotic force controller that has a simple architecture yet is robust to bounded uncertainty in the environmental stiffness. Generic stability conditions for the controller are developed and a simple design methodology is formulated. The controller design is tested on an experimental robot, and is shown to perform favourably in the presence of large changes in environmental operating conditions.

1 INTRODUCTION

Traditionally, most industrial robots are designed to allow accurate and repeatable control of the position and velocity of the tooling at the device's end effector. However, if robots are to perform complex tasks in a wider range of applications in the future, it will be essential to accurately control forces and torques at the end effector/task interface. In addition, task constraints sometimes require position control in some degrees-of-freedom (DOF), and force control in others. Thus, to fulfil these extra demands, an important area of robotics research is the implementation of stable and accurate force control. However, this is often difficult to achieve in practice, particularly where robots are operating in unpredictable or disordered environments.

A large number of force control techniques of varying complexity have been proposed over the last twenty years (Zang & Hemami 1997; Whitney 1985). The most basic direct methods simply transform joint-space torques into a Cartesian-space wrench, either in an open-loop fashion (which do not require the explicit measurement of forces and torques) or using inner and outer closed loops for accurate control of joint torques and Cartesian forces, respectively. However, since most industrial robots have position control loops that are not easily modified, indirect methods are often preferred. These involve modifying either joint or Cartesian position demands in order to control forces by deliberately introducing position control errors and using the inherent stiffness of the manipulator in different Cartesian directions. Alternatively, it is possible to add an outer force control loop in systems that have a facility for real-time path modification (Bicker et al. 1994).

Two major problems in the implementation of practical controllers are stability and robustness. Stable force control is particularly difficult to achieve in 'hard' or 'stiff' contact situations, where the control loop sampling rate may be a limiting factor. In an attempt to improve stability various methods have been proposed, the simplest being the addition of compliant devices at the robot wrist (Whitney & Nevins 1979). Another solution is to employ 'active compliance' filters, where force feedback data is digitally filtered to emulate a passive spring/damper arrangement (Kim et al. 1992). However, both methods introduce a potentially unacceptable lag. Robustness is a problem where environmental uncertainty exists, and effective force control can only be achieved by employing an accurate environment stiffness detection technique and smooth switching between controller gains (Ow 1997). This slows down task execution, and can result in unstable contact when the effective stiffness at the robot/environment interface (K_e) varies significantly.

Recent increases in processing power of low-cost computers has led to an increased interest in 'intelligent control' techniques such as those

employing fuzzy logic, artificial neural networks and genetic algorithms (Linkens & Nyongsa 1996).

Where attempts have been made to employ these techniques (specifically fuzzy logic) in explicit robot force controllers, simulation studies have demonstrated good tracking performance despite wide variations in environment stiffness, e.g. (Tarokh & Bailey 1997; Seraji 1998), and for specific contact situations, e.g. deburring (Kiguchi & Fukuda 1997). Improved performance using a hierarchical fuzzy force control strategy has also been demonstrated for various contact situations, such as peg-in-hole insertion (Lin & Huang 1998).

However, fuzzy techniques are not without problems. In addition to problems associated with dimensionality, i.e. large numbers of rules that must be evaluated in the inference process, the performance and stability of fuzzy systems are often difficult to validate analytically (Cao et al. 1998; Wolkenhauer & Edmunds 1997). Additionally, when compared to more 'traditional' control methods such as LQR (Frankin et al. 1994), the resulting fuzzy designs are more complex, have larger memory requirements and larger execution times (Bautista & Pont 2006).

Recent years have seen increased interest in the use of model following control (MFC) techniques. Due to its conceptually simple design and powerful robustness properties, this type of controller has been found to be particularly suited to industrial applications such as robotics and motion control (e.g. Li et al. 1998; Osypiuk et al. 2004). As such, it would seem that MFC-based techniques may prove to be applicable in the force control domain. This idea shall be explored in this paper, and a simple and stable MFC-based technique for force control is presented.

The paper is organised as follows. Section 2 presents a short overview of common difficulties in practical robotic force control. Following this, Section 3 gives a brief description of the MFC-based force controller, and generic stability conditions are developed. In section 4 this technique is applied to a robotic test facility and results are presented. Finally, conclusions and suggestions for further work are outlined.

2 FORCE CONTROL

Prior to examining the robust approach, it is beneficial to outline the force control problem under consideration and describe a conventional solution. A typical conventional force control scheme is shown in Figure 1. The combined stiffness at the end effector/task interface in the direction of the applied force is K_e. This varies between a minimum value, determined by the objects in the environment with which the robot is in contact, and a maximum value, limited by the stiffness of the arm and torque sensor. The latter is dominant when the robot is touching a surface of very high stiffness, i.e. in a hard contact situation. Designing a fixed-gain conventional controller to meet a chosen specification for a specific value of K_e is, in principle, a relatively straightforward task. A problem arises when K_e is unknown or variable, as shown in Figure 2. For example, consider the case where the system is tuned to achieve a specified performance at an upper limit of K_e - at low K_e the system will be overdamped with a relatively high settling time. Conversely, if the system had been tuned for the desired performance at the lower limit of K_e, significant overshoot and oscillatory behaviour would have occurred at higher stiffness values.

In practical robotic systems these effects often have serious consequences, mainly in relation to system stability. In particular, the finite and relatively low sampling rates of many industrial robot control systems can result in unstable behaviour, a situation exacerbated by the presence of noise, non-linearities and other factors. For this reason, force controllers of the type described usually require some form of environment stiffness detection technique to enable the controller gains to be switched accordingly. The main problem with this process is that it is time consuming, often involving 'guarded moves' to contact in order to enable sufficient data to be collected for the algorithm to work. Such methods can also be unreliable in the presence of transducer noise, and are not very effective in situations where K_e is variable or rapidly changing.

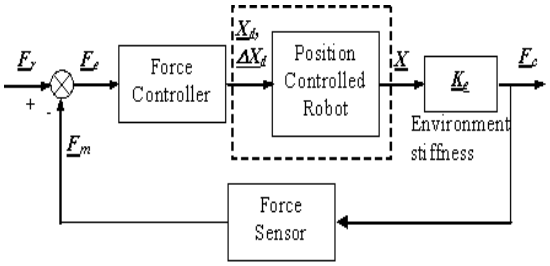


Figure 1: Robot force control.

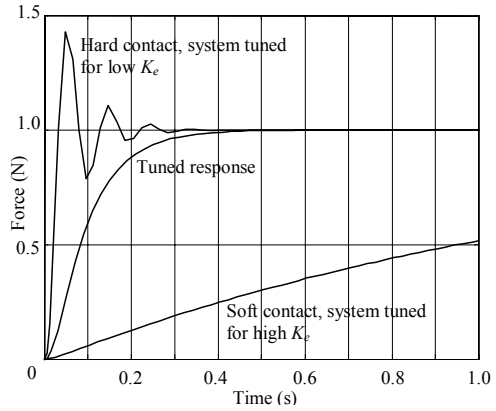


Figure 2: Effect of environmental stiffness.

3 ROBUST FORCE CONTROL

3.1 Principle

In this section we present the proposed robust force controller. It is loosely based around the robust PID strategy discussed in detail by Scokzowski et al. (2005). The original strategy is based upon a two-loop MFC, containing a nominal model of the controlled plant and two PID controllers. The block diagram of a basic MFC controller is shown in Figure 3.

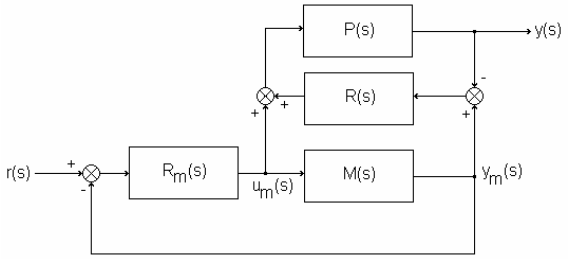


Figure 3: Robust PID based on MFC.

In this type of control, the model compensator $R_m(s)$ is tuned to a nominal model of the plant M(s); the actual plant P(s) contains bounded uncertainties. The auxiliary controller R(s) acts on the difference between the actual process output and the model process output to modify the model control signal $u_m(s)$, which is also fed to the plant.

As shown in Figure 1, when adding an outer force control loop, it is common to use a velocity signal as the input to the robot. In this case the model M(s) is simply the second order motion control loop dynamics augmented by a free integrator, and a known value of environment stiffness. The bounded uncertainty in the plant is then just the environment stiffness K_e, varying between Ke_{max} and Ke_{min}.

If the two loop controllers R(s) and $R_m(s)$ are simple proportional gains, as shown in Figure 4, then the MFC structure is considerably simplified. The model loop gain Kp can be tuned for Ke_{max}, (a relatively trivial task) whilst the auxiliary loop gain Kp' can be tuned to provide an additional control signal should the actual value of K_e be less than Ke_{max}. In the following section we will consider the stability criteria for this controller structure and provide a bound on the maximum value for Kp'.

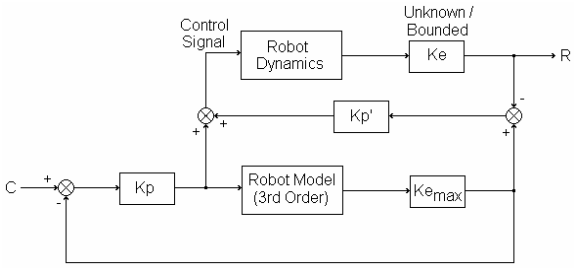


Figure 4: Robust force controller.

3.2 Design for Stability

If the 'model loop' controller $R_m(s)$ is tuned for stability using a nominal design method on the plant P(s) augmented by the maximum environmental stiffness gain Ke_{max}, then we know that the stability of the overall control strategy is restricted by the roots of the equation:

$$1 + R(s)M(s)[1 + \Delta(s)] = 0 \tag{1}$$

Where Δ(s) denotes the model perturbations (uncertainty). The objective is to find for a given plant and bounded uncertainty in the stiffness gain a maximum bound on |R(s)| that will maintain stability. In the case where the uncertainty exclusively resides in the environment stiffness gain K_e, then if the original loop is tuned for Ke_{max} then M(s)[1+Δ(s)] in (1) reduces to:

$$M(s)[1 + \Delta(s)] = P(s) = G(s)K_{e\max} \tag{2}$$

The robot dynamics have the form (due to the free integrator in the forward path):

$$G(s) = \frac{\omega_n{}^2}{s^3 + 2\xi\omega_n s^2 + \omega_n{}^2 s} \tag{3}$$

And the controller R(s) in this case is a single gain, Kp*'*, using (2) and (3) we can re-write equation (1) as follows:

$$s^3 + 2\xi\omega_n s^2 + \omega_n{}^2 s + \omega_n{}^2 Kp' Ke_{\max} = 0 \tag{4}$$

Applying the Routh-Hurwitz stability criterion (Pippard 1997) for a cubic equation, we know that the system is stable if all the co-efficients in the left of (4) are positive, and the following criterion is satisfied:

$$2\xi\omega_n\omega_n{}^2 \geq \omega_n{}^2 Kp' Ke_{\max} \tag{5}$$

Re-arranging (5) gives a stability limit for the controller gain $Kp'_{\max}$ as follows:

$$Kp'_{\max} = \frac{2\xi\omega_n}{Ke_{\max}} \tag{6}$$

Thus if the gain Kp' is chosen between the limits:

$$Kp < Kp' < Kp'_{\max} \tag{7}$$

The controller will be stable for unknown environment gains in the range $0 < K_e < K_{emax}$; as for all gains below Ke_{max}, the stability criteria of (5) holds.

4 EXPERIMENTAL TESTING

4.1 Test Facility

A research facility, previously described in detail (Short 2003), has been developed in the form of a planar robot arm and PC-based open architecture controller. The robot joints are actuated by brushless servomotors (with digital servoamplifiers), and the control loop for each axis is closed via a multitasking DSP embedded in a Delta Tau® Programmable Multi-Axis Controller (PMAC) motion control card, installed into the PC

Each axis has an individual PID controller with feedforward control to enable accurate velocity and position profile following. A six-axis force/torque sensor was developed in-house for the project, and used in this study. The robot arm is shown in Figure 5. For this work, a one-axis version of the system was employed by attaching the sensor to the wrist of the second link, which was then locked at 90° to the first link.

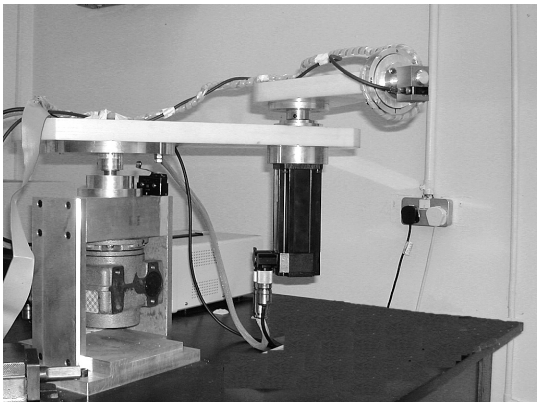


Figure 5: Test facility.

In this paper, we apply the controller proposed in the previous section to this facility. The controller was coded in C and added into the control library. Each experiment involved a contact situation, where the robot first approached a surface then applied a force of 25 N. The contact surface was varied in each experiment, and we used two surfaces; hard (steel) and soft (plastic). In order to reliably detect the contact surface, the end effector was fitted with a Baumer Electric® photoswitch which was calibrated to signal with high accuracy when an object was 5mm away. The robot thus approached the contact surface at a slow jog speed until this signal was made, then switched to force control mode. The sample rate was 200 Hz in each experiment. In the following section we describe the parameters that were used.

4.2 Controller Design

From a previous identification exercise, the parameters of the robot arm model and the environment stiffness limits were determined to be as follows (Short 2003):

$$\omega_n = 244\,rad/s,\ \xi = 1,$$
$$K_{e\max} = 168\ N/mm, \quad K_{e\min} = 11\ N/mm \tag{8}$$

Using these parameters, the nominal loop gain Kp was tuned to a value of 0.02 to give the desired transient performance – a 95% rise time of approximately 2 seconds with minimal (ideally zero) overshoot. Using (6), Kp'_{max} was determined to be 2.9. We therefore chose a value of Kp' = 1.5 for the experiments.

4.3 Experimental Results

Figure 6 shows the response of the system when applying a force to the hard (steel) surface. The very small negative force indicated before contact with the surface was made (at approx 1s) was due to a small drift in the calibration of the force sensor whilst moving in free space. Figure 7 shows the soft (plastic) case. We also show, for completeness, the contact situation for a single loop controller tuned for high K_e in the soft contact case. This is shown in figure 8.

These figures demonstrate the effectiveness of the approach. Comparing Figures 7 and 8, the compensation added by the extra loop can clearly be seen; in Figure 7 we see an almost identical transient to Figure 6. Additionally, in Figure 6 the controller demonstrates no signs of instability as Kp' was kept below the maximum amount. We also measured the integral of time by absolute error ITAE (Franklin et al. 1994) for the responses shown in Figures 6, 7 and 8. This is shown in Table 1. From this the closeness of the proposed robust controller transient responses can be seen (R). The response of the normal (N) controller is also shown in the table. The poor quality of control is clearly highlighted by this vastly increased value.

Table 1: ITAE measures for contact situations.

System	ITAE
(R) Low K_e	23.61
(R) High K_e	23.95
(N) Low K_e	666.5

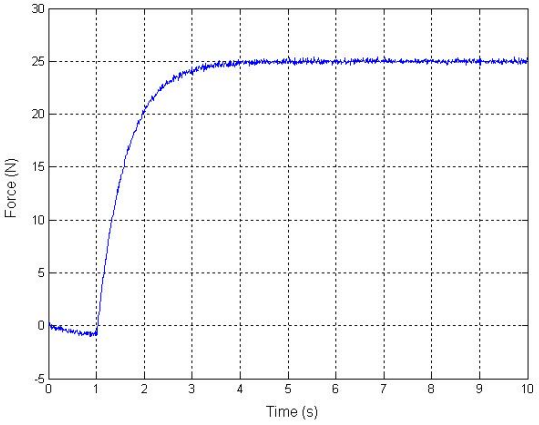


Figure 6: Hard contact situation.

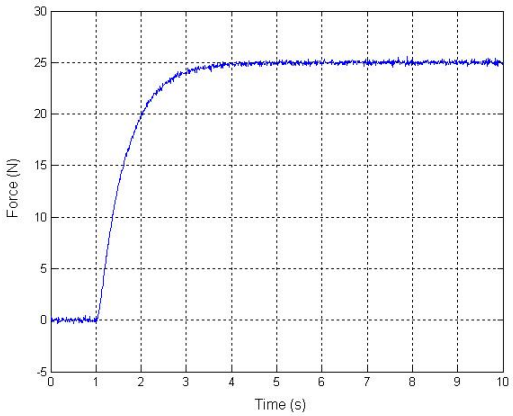


Figure 7: Soft contact situation.

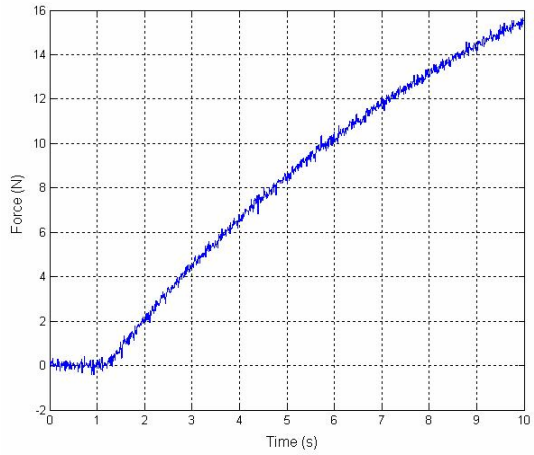


Figure 8: Soft contact situation (normal controller).

5 CONCLUSIONS

In this paper a distinct method for robotic force control has been proposed and tested using an experimental test robot. The method has been shown to improve system performance where a high degree of environmental uncertainty exists, without the need for a stiffness detection routine. The method is conceptually simple and extremely easy to implement; its simplicity also lends itself to easy analytical analysis.

The practical realisation of robotic force control remains a problematic area of research. However, the potential of simple, stable controllers to overcome fundamental difficulties associated with applications where environmental uncertainty exists has been demonstrated.

However, work is required to further validate the control method. This will include analysis of situations where PD controllers are used as the loop compensators, and forces are applied in Cartesian coordinates. We will also consider the effects of model mismatch (which is inevitable if the methodology is to be applied to industrial robots). Further work will also consider implementation on a 6-DOF manipulator to confirm its performance in a range of industrial tasks, and to contrast the approach with other methodologies.

REFERENCES

Bautista, R., Pont, M.J., 2006. Is fuzzy logic a practical choice in resource-constrained embedded control systems implemented using general-purpose microcontrollers? In *Proceedings of the 9th IEEE International Workshop on Advanced Motion Control*, Istanbul, Volume 2, pp.692-697.

Bicker, R., Burn, K., Glennie, D., Ow, S.M., 1994. Application of force control in telerobotics. *Proc Int Conf EURISCON '94*, Malaga, Spain.

Cao, S.G., Rees, N.W., Feng, G., 1998. Lyapunov-like stability theorems for continuous-time fuzzy control systems, *Int J Control*. Vol. 69(1), pp. 49-64.

Franklin, G.F., Powell, J.D., Emani-Naeini, A., 1994. *Feedback Control Of Dynamic Systems*. Addison-Wesley Publishing, Reading Massachusetts, third edition.

Kiguchi, K., Fukuda, T., 1997. Intelligent position/force controller for industrial robot manipulators – application of fuzzy neural networks. *IEEE Trans Industrial Electronics*, Vol. 44(6), pp. 753-761.

Kim, W.S., Hannaford, B., Bejczy, A.K., 1992. Force Reflection and Shared Compliant Control in Operating Telemanipulators with Time Delay. *IEEE Trans on Robotics and Automation*, Vol. 8(2), pp. 176-185.

Li, G., Tsang, K.M., Ho, S.L., 1998. A novel model following scheme with simple structure for electrical position servo systems. *Int. J. Syst. Sci.*, Vol. 29, No. 9, pp. 959–969.

Lin, S.T., Huang, A.K., 1998. Hierarchical Fuzzy Force Control for Industrial Robots. *IEEE Transactions on Industrial Electronics*, Vol. 45, No. 4, pp. 646-653.

Linkens, D.H., Nyongesa, H.O., 1996. Learning systems in intelligent control: an appraisal of fuzzy, neural and genetic algorithm control applications. *IEE Proc Control Theory Appl*, Vol. 143(4), pp. 367-386.

Osypiuk, R., Finkemeyer, B., Wahl, F.M., 2004. Forward-model based control system for robot manipulators. *Robotica*, Vol. 22, No. 2, pp. 155–161.

Ow, S.M., 1997. *Force Control in Telerobotics*. PhD Thesis, University of Newcastle upon Tyne, UK.

Pippard, A.B., 1997. *Response & Stability: An Introduction to the Physical Theory*. Cambridge University Press.

Seraji, H., 1998. Nonlinear and Adaptive Control of Force and Compliance in Manipulators. *Int J Robotics Research*, Vol. 17(5) pp. 467-484.

Short, M., 2003. *A Generic Controller Architecture for Advanced and Intelligent Robots*. PhD. Thesis, University of Sunderland, UK.

Skoczowski, S., Domek, S., Pietrusewicz, K., Broel-Plater, B., 2005. A Method for Improving the Robustness of PID Control. *IEEE Transactions On Industrial Electronics*, Vol. 52, No. 6.

Tarokh, M., Bailey, S., 1997. Adaptive fuzzy force control of manipulators with unknown environment parameters. *J Robotic Sys*, Vol. 14(5), pp. 341-353.

Whitney, D.E., 1985. Historical Perspective and State of the Art in Robot Force Control. *Int J Robotics Res*, Vol. 6(1), pp. 3-14.

Whitney, D.E., Nevins, J.L., 1979. What is the Remote Centre Compliance (RCC) and what can it do? *Proc Int Symp on Industrial Robots*, Washington DC, pp. 135-152.

Wolkenhauer, O., Edmunds, J.M., 1997. A critique of fuzzy logic in control. *Int J Electrical Engineering Education*, Vol. 34(3), pp. 235-242.

Zhang, G., Hemami, A, 1997. An Overview of Robot Force Control. *Robotica*, Vol. 15, pp. 473-482.

PROGRESSES IN CONTINUOUS SPEECH RECOGNITION BASED ON STATISTICAL MODELLING FOR ROMANIAN LANGUAGE

Corneliu Octavian Dumitru[1,2]
[1]*University Politehnica Bucharest, Faculty of Electronics Telecommunications and Information Technology, Romania*
[2]*ARTEMIS Department, GET/INT, Evry, France*
odumitru@alpha.imag.pub.ro

Inge Gavat, Diana Militaru
University Politehnica Bucharest, Faculty of Electronics Telecommunications and Information Technology, Romania
igavat@alpha.imag.pub.ro, diana.militaru@gmail.com

Keywords: MFCC, LPC, PLP, statistical modelling, monophone, triphone.

Abstract: In this paper we will present progresses made in Automatic Speech Recognition (ASR) for Romanian language based on statistical modelling with hidden Markov models (HMMs). The progresses concern enhancement of modelling by taking into account the context in form of triphones, improvement of speaker independence by applying a gender specific training and enlargement of the feature categories used to describe speech sequences derived not only from perceptual cepstral analysis but also from perceptual linear prediction.

1 INTRODUCTION

After years of research and development, the problem of automatic speech recognition and understanding is still an open issue. The end goal of translation into text, accurate and efficient, unaffected by speaker, environment or equipment used is very difficult to achieve and many challenges are to be faced.

Some factors which make difficult the problem of automatic speech recognition (ASR) are voice variations in context or in environment, syntax, the size of vocabulary. How this problems can easy be accommodated by humans, to continue studies in human speech perception can be very important to improve performance in speech recognition by machine. As alternative, human performance can be regarded as guide for ASRs and in this moment neither the best systems built for English language can not reach human performance.

In this paper we will present progresses made in ASR for Romanian language based on statistical modelling with hidden Markov models (HMMs) using the HTK toolkit, developed by the Cambridge University Engineering Department (Woodland, 1994). We started in this domain with a system for continuous speech recognition (Gavat, 2003) based on monophone models, regarding words like phone sequences, neglecting co-articulation. The obtained results challenged us to improvements that addressed especially acoustical context modelling but also speaker independence enhancement and optimization of features choice describing speech.

The remainder of this paper will be also structured as follows: Section 2 presents an overview of the built ASR system. Section 3 describes the context based acoustical modelling, realized with triphones. Section 4 gives an overview of the Rumanian database and the changes made in view of the experiments for speaker independence enhancement. Tests are described and discussed in section 5. Finally, conclusions and future works are given in section 6.

2 SYSTEM OVERWIEV

The functional schema of our speech recognition system is given in figure 1. The system acts in three

main phases: training, testing and evaluation of results.

In the training phase the current parameters of HMMs are established by incrementally refining an initial set of "white" acoustical HMM models.

In the testing phase the test data are verified with the training database using the grammar, the lexicon, the word network and the testing dictionary.

In the evaluation phase, by comparing the transcription of the test speech sequences with the reference transcription, the recognition correctness and accuracy are calculated.

In each phase a data preparation stage is introduced in order to build a set of speech data files and their transcription in the required format. An important step in this stage is feature extraction.

Feature extraction is the lowest level of automatic speech recognition and it lies in the task of extracting the limited amount of useful information from high-dimensional data. The feature extractors maintain much of the characteristics of the original speech and eliminate much of the extraneous information.

After feature extraction, a sequence O of feature vectors are the input data for the testing phase, in order to establish the correct uttered sentence W. We need to estimate therefore the probability of acoustic features given the phonetic model, so that we can recognize the input data for the correct sentence. This probability is referred to as acoustic probability P (O|W).

HMMs are the most common models used in automatic speech recognition systems to model the joint probability distribution of feature vectors for a given utterance model (Gavat, 2000).

Mathematically the problem in continuous speech recognition is to find a sequence of words $\hat{W}$ such that

$$\hat{W} = \arg\max_{W} P(O|W)P(W) \quad (1)$$

The most probable sentence W given the observation sequence O can be computed by taking this product of two probabilities for each sentence and choosing the sentence for which the product is the highest.

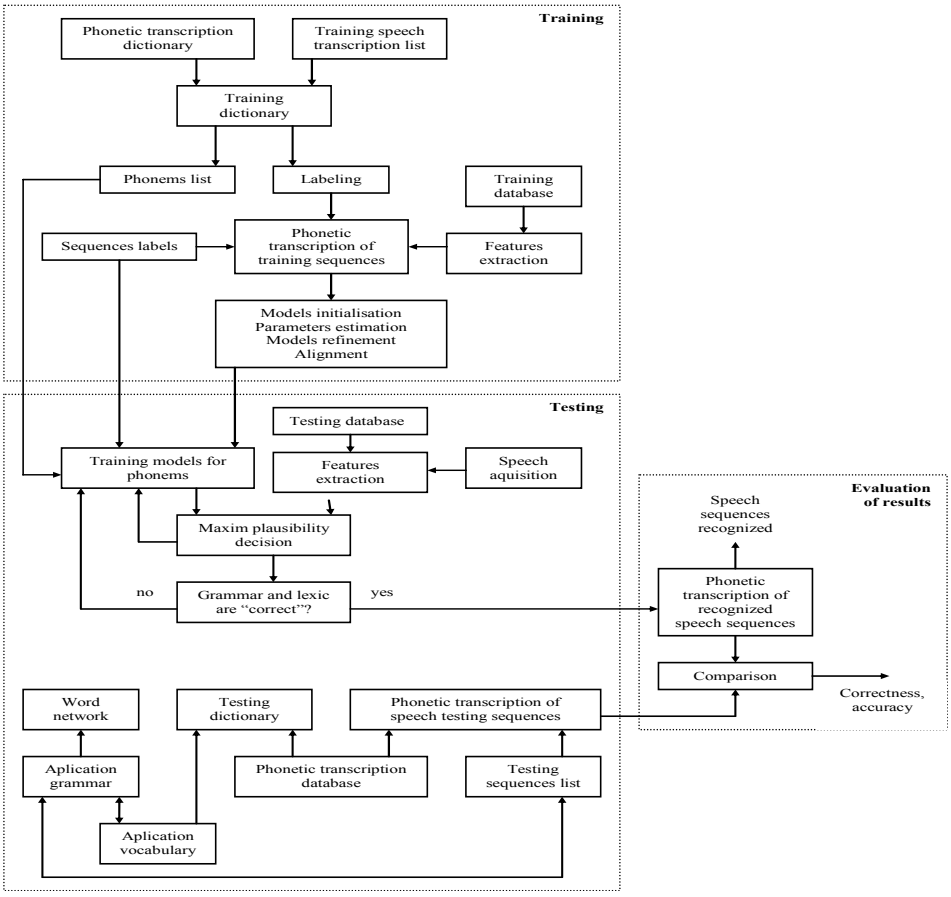


Figure 1: The speech recognition system.

The prior probability P (W) is calculated using a language model appropriate for the recognition task, and the acoustic probability P (O|W), is calculated by concatenating the HMMs of the words in the sequence W and using the Viterbi algorithm for decoding. A silence or a 'short pause' model is usually inserted between the HMMs to be concatenated.

3 TRIPHONE MODELLING

For small vocabulary recognition, word models are widely used, since they are accurate and trainable. In the situation of a specific and limited task they become valid if enough training data are available, but they are typically not generalizable. Therefore, usually for not very limited tasks are preferred phonetic models based on monophones, because the phones, as smallest linguistic units, are easy generalizable and of course also trainable.

Monophones constitute the foundation of any training method, in any language, and we also started with them. But a refinement of this initial step was necessary because in real speech, the words are not simple strings of independent phonemes. The realization of a phoneme is strongly affected by its immediately neighboring phonemes by co-articulation. Because of this, monophone models have been changed in time with triphone models that became the actual state of the art in automatic speech recognition with large vocabularies (Young, 1992), (Young, 1994).

A triphone model is a phonetic model that takes into consideration the left and the right neighbouring phones. This immediate neighbour – phonemes are called respectively the left and the right context; a phoneme constitutes with the left and right context a triphone. For example in the SAMPA (Speech Assessment Methods Phonetic Alphabet) transcription "m - a + j" of the Romanian word "mai", regarded as triphone, the phoneme "m" has as left context "a" and as right context "j".

For each such a triphone a model must be trained: in Romanian that will give a number which equals 40,000 models, situation totally unacceptable for a real world system. In our speech recognition task we have modelled only internal – word triphones and the adopted state tying procedure has conducted to a controllable situation.

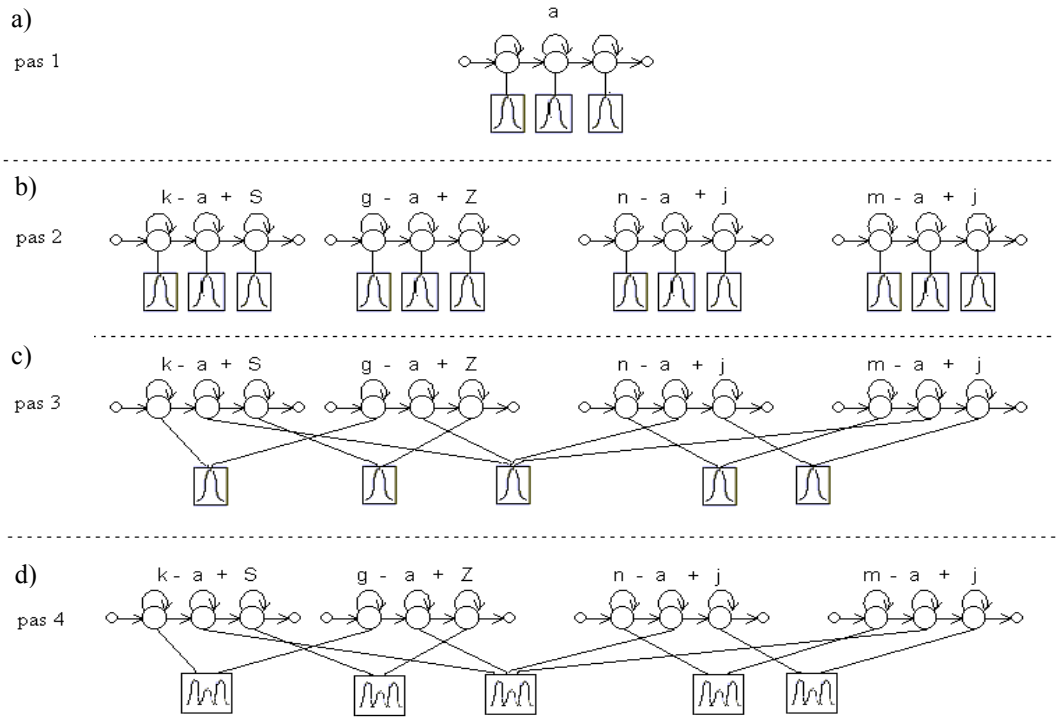


Figure 2: Different models for triphones around the phoneme "a".

If triphones are used in place of monophonemes, the number of needed model increases and it may occur the problem of insufficient training data. To solve this problem, tying of acoustically similar states of the models built for triphones corresponding to each context is an efficient solution. For example, in figure 2b, four models are represented for different contexts of the phoneme "a", namely the triphones "k – a + S", "g – a + z", "n – a + j", "m – a + j". In figure 2c, 2d, there are represented the clusters formed with acoustically similar states of the corresponding HMMs.

The choice of the states and the clustering in phonetic classes are achieved by mean of phonetic decision trees. A phonetic decision tree built as a binary tree, is shown in figure 3 and has in the root node all the training frames to be tied, in other words all the contexts of a phoneme. To each node of the tree, beginning with the parent – nodes, a question is associated concerning the contexts of the phoneme. Possible questions are, for example: is the right context a vowel (R = Consonant?), is the left context a phoneme "a" (L = a?); the first answer designates a large class of phonemes, the second only a single phonetic element. Depending on the answer, yes or no, child nodes are created and the frames are placed in them. New questions are further made for the child nodes, and the frames are divided again.

The questions are chosen in order to increase the log likelihood of the data after splitting. Splitting is stopped when increasing in log likelihood is less than an imposed threshold resulting a leaf node. In such leaf nodes are concentrated all states having the same answer to the question made along the path from the root node and therefore states reaching the same leaf node can be tied as regarded acoustically similar. For each leaf node pair the occupancy must be calculated in order to merge insufficient occupied leaf nodes.

A decision tree is built for each state of each phoneme. The sequential top down construction of the decision trees was realized automatically, with an algorithm selecting the questions to be answered from a large set of 130 questions, established after knowledge about phonetic rules for Romanian language.

4 DATABASE

The data are sampled by 16 kHz, quantified with 16 bits, and recorded in a laboratory environment.

For continuous speech recognition, database for training is constituted by 3300 phrases, uttered by 11 speakers, 7 males and 4 females, each speaker reading 300 phrases.

The databases for testing contained 220 phrases uttered by 11 speakers, each of them reading 20 phrases.

The training database contains over 3200 distinct words; the testing database contains 900 distinct words.

In order to carry out our experiments about speaker independence, the database was reorganized as follows: one database for male speakers (MS), one database for female speakers (FS) and one database for male and female speakers (MS and FS). In all cases we have excluded one MS and one FS from the training and used for testing.

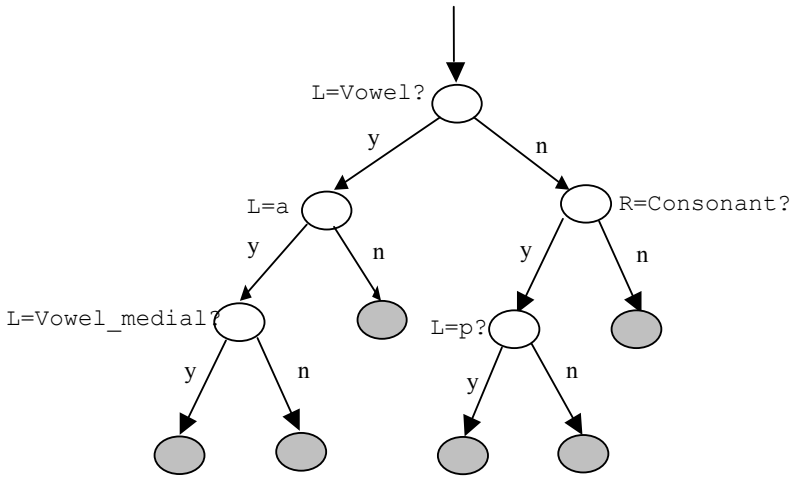


Figure 3: Phonetic tree for phoneme m in state 2.

The speech files from these databases were analysed in order to extract the interesting features. The feature extraction methods used are based on linear predictive coding (LPC), perceptual linear prediction (PLP) (Hermansky, 1990) mel-frequency cepstral coefficients (MFCC).

5 EXPERIMENTAL RESULTS

To assess the progresses made with our ASR system we initiated comparative tests for the performance expressed in word recognition rate (WRR) to establish the values under the new conditions versus the preceding ones. The comparison is made for the following situations:

- Triphone modelling/monophone modelling
- Gender based training/mixed training
- LPC and PLP/MFCC.

The results obtained in the experiments realized under these conditions are summarized in Table 1, Table 2, and Table 3.

Table 1: Word Recognition Rate: Training MS, testing MS or FS.

Training MS	Type	WRR (%)		
		MFCC_ D_A	LPC	PLP
Testing MS	Monophone	56.33	30.85	34.02
	Triphone	81.02	49.73	68.10
Testing FS	Monophone	40.98	23.23	25.12
	Triphone	72.86	47.68	59.00

The WRR are:

- For 12 LPC coefficients the word recognition rates are low: 30.85% (monophone) training and testing with MS and 49.73% (triphone); 31.11% (monophone) training and testing with FS and 61.15% (triphone); 26.10% (monophone) training MS and FS and testing with MS and 51.5% (triphone).
- For 5 PLP coefficients the obtained results are very promising, giving word recognition rates about 58.55% (triphone training and testing FS), 68.10% (triphone training and testing MS) and 70.11% (triphone training MS and FS and testing MS).
- For 36 MFCC_D_A coefficients (mel-cepstral coefficients with first and second order variation) we obtained the best results, as we expected: monophone 56.33% and triphone 81.02%, training and testing with MS; monophone 56.67% and triphone 78.43%, training and testing with FS; monophone 57.44% and triphone 78.24%, training MS and FS and testing with MS.

Table 2: Word Recognition Rate: Training FS, testing MS or FS.

Training FS	Type	WRR (%)		
		MFCC_ D_A	LPC	PLP
Testing MS	Monophone	53.56	26.72	23.78
	Triphone	69.23	49.73	53.02
Testing FS	Monophone	56.67	31.11	34.22
	Triphone	78.43	61.15	58.55

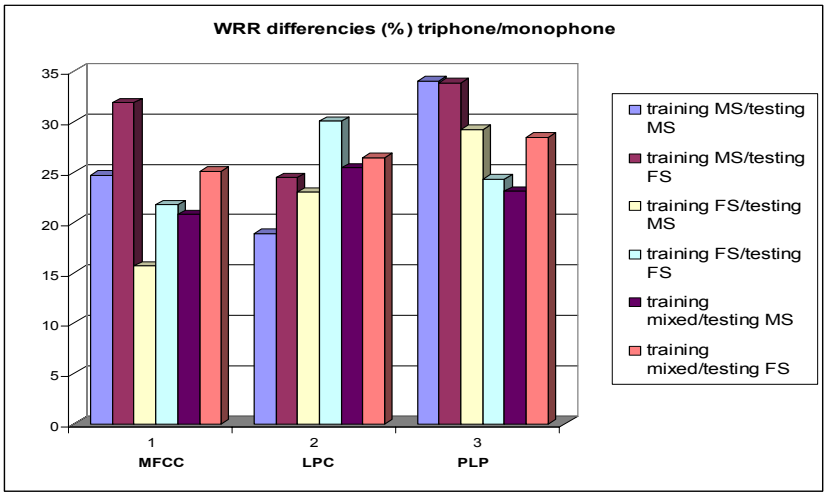
Table 3: Word Recognition Rate: Training MS and FS, testing MS or FS.

Training MS and FS	Type	WRR (%)		
		MFCC_ D_A	LPC	PLP
Testing MS	Monophone	57.44	26.10	47
	Triphone	78.24	51.50	70.11
Testing FS	Monophone	49.89	24.06	41.22
	Triphone	74.95	50.49	69.65

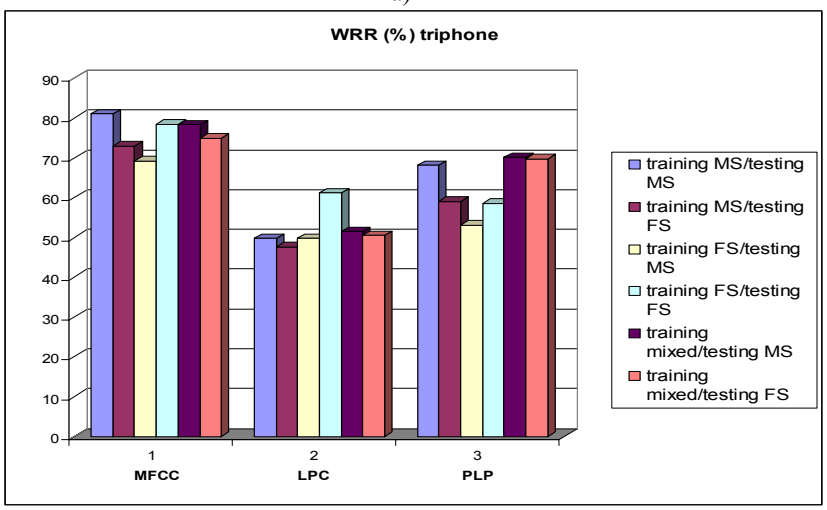
6 CONCLUSION

After the experiments made we have following conclusions:

- The triphone modelling is effective, conducting to increasing in WRR between 15% and 30% versus the monophone modelling. The maximal enhancement exceeds 30% for training MS and testing FS for MFCC_D_A (see figure 4.a).
- A gender based training conduct to good result for test made with speakers from the same gender (training MS / testing MS: 81.02%, testing FS: 72.86%; training FS/testing FS: 78.43%, testing MS: 69.23%); changing gender in testing versus training leads to a decrease in WRR around 10%. For a mixed trained data base changing gender determines only variations around 5% in WRR (see figure 4.b).



a)



b)

Figure 4: a) Chart 1; b) Chart 2.

REFERENCES

Woodland P.C., Odell J.J., Valtchev V., Young S.J, 1994, Large Vocabulary Continuous Speech Recognition Using HTK, *Proceedings of ICASSP 1994,* Adelaide.

Gavat I., Dumitru C.O., Costache G., Militaru D., 2003, Continuous Speech Recognition Based on Statistical Methods, *Proceedings of SPED 2003*, pp. 115-126, Romania.

Young, S.J., 1992, The General Use of Tying in Phoneme-Based HMM Speech Recognizers, *Proceedings of ICASSP 1992*, vol. 1, pp. 569-572.

Young S.J., Odell J.J., Woodland P.C., 1994, Tree Based State Tying for High Accuracy Modelling, *ARPA Workshop on Human Language Technology*.

SAMPA (Speech Assessment Methods Phonetic Alphabet), http://www.phon.ucl.ac.uk/home/sampa/home.htm

Hermansky H., 1990, Perceptual Linear Predictive Analysis of Speech, *J. Acoust. Soc. America*, vol.87, no.4, pp. 1738-1752.

Oancea E., Gavat I., Dumitru C.O., Munteanu D., 2004, Continuous speech recognition for Romanian language based on context-dependent modelling, *Proceedings of COMMUNICATION 2004*, pp. 221-224, Romania.

Dumitru C.O., Gavat I., 2005, Features Extraction, Modelling and Training Strategies in Continuous Speech Recognition for Romanian Language, *Proc. EUROCON 2005*, pp. 1425-1428, Serbia & Montenegro.

Gavat I., Zirra, M., Grigore O., Sabac B, Valsan Z., Cula O., Pascu A., 2000, *Elemente de sinteza si recunoasterea vorbirii*, Ed. Printech, Bucharest, Romania.

Lupu, E., Pop, G. Petre, 2004, *Prelucrarea numerica a semnalului vocal. Elemente de analiza si recunoastere*, Ed. Risoprint , Cluj-Napoca, Romania.

USING NOISE TO IMPROVE MEASUREMENT AND INFORMATION PROCESSING

Solenna Blanchard, David Rousseau and François Chapeau-Blondeau
Laboratoire d'Ingénierie des Systèmes Automatisés (LISA), Université d'Angers
62 avenue Notre Dame du Lac, 49000 Angers, France

Keywords: Noise, Stochastic resonance, Information processing, Measurement, Sensor.

Abstract: This paper proposes a synthetic presentation on the phenomenon of stochastic resonance or improvement through the action of noise. Several situations and mechanisms are reported, demonstrating a constructive role of noise, in the context of measurement and sensors, data and information processing, with examples on digital images.

1 INTRODUCTION

In the context of measurement and information processing, it is progressively realized that noise is not always a nuisance, but that it can sometimes play a beneficial role (Wiesenfeld and Moss, 1995; Andò and Graziani, 2001; Chapeau-Blondeau and Rousseau, 2002; Chapeau-Blondeau and Rousseau, 2004). Such a constructive role of noise can be observed in diverse situations, through different cooperative mechanisms, assessed by various measures of performance to quantify the improvement. The term "stochastic resonance" is used as a common name to unify these situations where a measure of performance is improved to culminate (resonate) at a maximum when the level of noise is raised. In the recent years, many forms of stochastic resonance have been introduced and analyzed. Experimental observations of stochastic resonance have been obtained in many areas, for instance with electronic circuits, optical devices, neuronal processes, nanotechnologies.

In the present paper, we propose an overview of some basic mechanisms of stochastic resonance, including some recent ones, that we organize in a synthetic perspective. We specially focus on possible constructive role of noise for measurement and sensors, data and information processing. For illustration, we provide examples on digital images, a class of signals not so often considered for applying stochastic resonance, which gives us the opportunity of also showing new examples.

2 NOISE-SHAPED SENSORS

In this section, we show a constructive action of the noise that can be used to shape the input–output characteristic of devices and we give an example of application illustrated with saturating imaging sensors.

We consider a device with static nonlinear input–output characteristic $g(\cdot)$. This device is in charge of the transmission or the processing of an information carrying signal s so as to produce the output signal y with

$$y = g(s)\,, \tag{1}$$

where input signal s and output signal y may be function of time or space. Let us assume that the input–output characteristic $g(\cdot)$ of our device is not optimally adapted to transmit or process the input signal s. Therefrom, one may look for another device with a more suitable input–output characteristic shape. As an alternative, we are going to show that it is also sometimes possible to modify the input–output characteristic $g(\cdot)$ of such a device without having to change any physical parameter of the device itself.

We introduce a noise η in the input–output relation of Eq. (1) which becomes $y = g(s+\eta)$. This noise η can be a native noise due to the physics of the device or a noise purposely injected to the input of the device. Then, since the device is no longer deterministic, an effective or average input–output characteristic can be defined as $g_{\rm eff}(\cdot)$ given by the expectation

$$g_{\rm eff}(s) = \mathrm{E}[y] = \int_{-\infty}^{+\infty} g(u)\, f_\eta(u-s)du\,, \tag{2}$$

with $f_\eta(u)$ the probability density function of the noise η. In presence of the noise η, the shape of the device input–output characteristic, $g_{\text{eff}}(\cdot)$ in Eq. (2), is now controlled by $g(\cdot)$ and by the noise probability density function $f_\eta(u)$. Therefore, a modification in the response of a memoryless device can be obtained thanks to the presence of a noise η which makes it possible to shape the input–output characteristic of the device without changing the device itself.

In practice, the modified input–output characteristic $g_{\text{eff}}(\cdot)$ of the device in Eq. (2) is not directly available. Yet, it is possible to have a device presenting an approximation of the response of Eq. (2) by averaging N acquisitions y_i with $i \in \{1,\dots N\}$ to produce

$$y = \frac{1}{N}\sum_{i=1}^{N} y_i = \frac{1}{N}\sum_{i=1}^{N} g(s+\eta_i)\ , \tag{3}$$

where the N noises η_i are white, mutually independent and identically distributed with probability density function $f_\eta(u)$. Practical implementation of the process of Eq. (3) can be obtained, as proposed in (Stocks, 2000) for 1-bit quantizers, via replication of the devices associated in a parallel array where N independent noises are added at the input of each device or, as proposed in (Gammaitoni, 1995) for a constant signal, by collecting the output of a single device at N distinct instants. Similarly to what is found in Eq. (2), the input–output characteristic of the process of Eq. (3) is shaped by the presence of the N noises η_i. Because of Eq. (3), one has first

$$\mathrm{E}[y] = \mathrm{E}[y_i]\ , \tag{4}$$

and also one has for any i

$$\mathrm{E}[y_i] = \int_{-\infty}^{+\infty} g(s+u)\, f_\eta(u) du\ . \tag{5}$$

The N noises η_i bring fluctuations which can be quantified by the nonstationary variance $\mathrm{var}[y] = \mathrm{E}[y^2] - \mathrm{E}[y]^2$, with $\mathrm{E}[y^2] = \mathrm{E}[y_i^2]/N + \mathrm{E}[y]^2(N-1)/N$ and

$$\mathrm{E}[y_i^2] = \int_{-\infty}^{+\infty} g^2(s+u)\, f_\eta(u) du\ . \tag{6}$$

For large values of N, $\mathrm{var}[y]$ tends to zero. In these asymptotic conditions where N tends to infinity, the process constituted by the device $g(\cdot)$, the N noises η_i and the averaging of Eq. (3), becomes a deterministic equivalent device with input–output characteristic given by Eq. (2). For finite values of N, the presence of N noises η_i will play a constructive role if the improvement brought to the transmission or processing of the input signal s by the modification of the device characteristic is greater than the nuisance due to the remaining fluctuations in y.

The process of Eq. (3) delimits a general problem: given a device characteristic $g(\cdot)$ and a number N of averaging samples, how can one choose the probability density function of the noises η_i to obtain a targeted characteristic response. This inverse problem is, in general, difficult to solve. A pragmatic solution, inspired from the studies on stochastic resonance, consists in fixing a probability density function for the noises η_i and to act only on the rms amplitude σ_η of these identical independent noises.

For illustration, we now give an example of application of the process of Eq. (3). We consider devices with input–output characteristic $g(u)$ presenting a linear regime limited by a threshold and a saturation

$$g(u) = \begin{cases} 0 & \text{pour} \quad u \le 0 \\ u & \text{pour} \quad 0 < u < 1 \\ 1 & \text{pour} \quad u \ge 1\ . \end{cases} \tag{7}$$

The possibility of shaping the response of such devices by using the process of Eq. (3) is shown in Figure 1. The noises η_i injected in Eq. (3), arbitrarily chosen Gaussian here, tend to extend the amplitude range upon which the effective input–output characteristic $g_{\text{eff}}(\cdot)$ of Eq. (2) is linear.

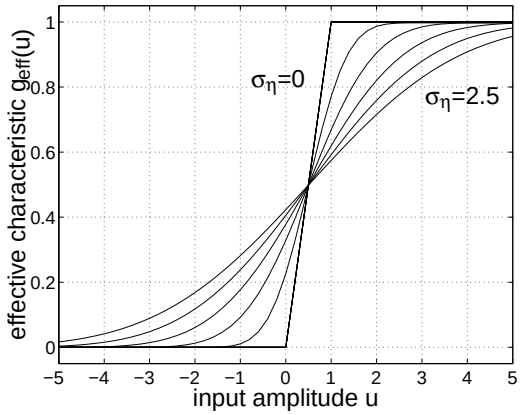


Figure 1: Effective input–output characteristic $g_{\text{eff}}(\cdot)$ of Eq. (2) for the device of Eq. (7) in presence of Gaussian centered noise with various rms amplitude $\sigma_\eta = 0, 0.5, 1, 1.5, 2.5$.

The input–output characteristic $g(u)$ of Eq. (7) can constitute a basic model for measurement sensors. In the domain of instrumentation and measurement, a quasi-linear behavior associated to perfect reconstruction is sought. Nevertheless, sensor devices are usually linear for moderate inputs but can present saturation at large inputs or/and a threshold for small inputs. Such behaviors at large and small inputs induce distortions degrading the quality of the signal transmitted by these sensor devices. Therefore, the linear regime of the input–output characteristic of a saturating sensor usually sets the limit of the signal dynamic to be

transmitted with fidelity. In this measurement framework, the process of Eq. (3) can be used to widen the dynamic of sensors presenting threshold and saturation like in Eq. (7).

For further illustrations, we consider the case where Eq. (7) models the response of an imaging sensor (CCD, retina, or even photographic films). We assume that, although all physical parameters (for example luminance of the scene, time exposure, imaging sensor sensibility) have been adjusted to their best, the image s submitted to Eq. (7) still undergoes saturation by $g(\cdot)$ and is therefore over-exposed. The possibility of a noise–widened dynamic of this imaging sensor can be visually appreciated in Figure 2. Fluctuations due to the presence of noises η_i in Eq. (3) are decreasing with increasing N. Nevertheless, in some cases, as perceptible in Figure 2c, the sensor device can benefit from the presence of the noise even with $N = 1$ in the process of Eq. (3). Also, the noise–widened visual improvement of the transmitted image, can be quantitatively assessed, in Figure 3, by the normalized cross-covariance between the original image and the transmitted image.

Figure 2: Noise–widened dynamic of an imaging sensor with response given by Eq. (7). (a) a 8-bit version of the "lena" image correctly exposed with 256 grey-levels coded between 0 and 1; (b) over-exposed transmitted image in absence of noise. The over-exposure is controlled by an exposure parameter k_{ex}, which is a constant added to all the pixel value of the original image before the process of Eq. (7); (c) transmitted image with the same exposure parameter $k_{ex} = 0.75$ but with the presence of zero-mean Gaussian noise of rms amplitude $\sigma_\eta = 0.3$ with $N = 1$ acquisition averaged; (d),(e),(f) same conditions but with $\sigma_\eta = 0.1, 0.2, 0.4$ with $N = 3, 7, 63$.

3 NOISE-ASSISTED INFORMATION TRANSMISSION

We now move to a higher information-processing level, with statistical quantification of information, and possible connection to pattern recognition tasks, for another example of a constructive role of noise. Consider a binary image with values $s(x_1,x_2) \in \{0,1\}$ at spatial coordinates (x_1,x_2). The detector $g(\cdot)$ is taken as a hard limiter with threshold θ,

$$g(u) = \begin{cases} 0 & \text{for} \quad u \leq \theta \\ 1 & \text{for} \quad u > \theta \,, \end{cases} \tag{8}$$

and delivers the output image $y(x_1,x_2) = g[s(x_1,x_2) + \eta(x_1,x_2)]$, with independent noise $\eta(x_1,x_2)$ at distinct pixels $(x_1,,x_2)$. When the detection threshold θ is high relative to the values of the input image $s(x_1,x_2)$, i.e. when $\theta > 1$, then $s(x_1,x_2)$ in absence of the noise $\eta(x_1,x_2)$ remains undetected as the output image $y(x_1,x_2)$ remains a dark image; thus, with no noise, no information is transmitted from $s(x_1,x_2)$ to $y(x_1,x_2)$. From this situation, addition of the noise $\eta(x_1,x_2)$ allows a cooperative effect where $s(x_1,x_2)$ and $\eta(x_1,x_2)$ cooperate to overcome the detection threshold. This translates into the possibility of increasing and maximizing the information shared between $s(x_1,x_2)$ and $y(x_1,x_2)$ thanks to the action of the noise $\eta(x_1,x_2)$ at a nonzero level which can be optimized. The effect can be precisely quantified by means of a Shannon mutual information $I(s,y)$ between $s(x_1,x_2)$ and $y(x_1,x_2)$, definable as

$$I(s,y) = H(y) - H(y|s) \,. \tag{9}$$

With the function $h(u) = -u\log_2(u)$, the entropies in Eq. (9) are

$$H(y) = h(p_{00}p_0 + p_{01}p_1) + h(p_{10}p_0 + p_{11}p_1) \tag{10}$$

and

$$H(y|s) = p_0[h(p_{00}) + h(p_{10})] + p_1[h(p_{01}) + h(p_{11})] \,. \tag{11}$$

In Eqs. (10)–(11) one has the probabilities $\Pr\{s = 1\} = p_1 = 1 - p_0$ determined by the input image

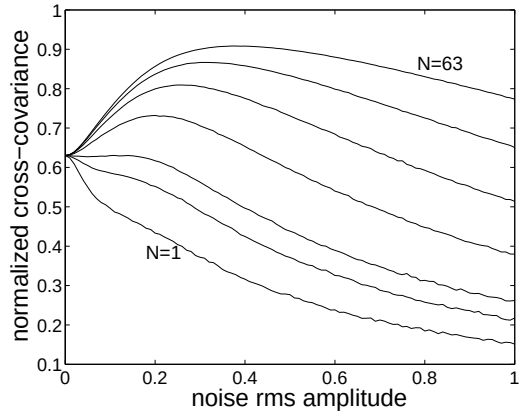


Figure 3: Normalized cross-covariance of original "lena" image correctly exposed with the image transmitted by the process of Eq. (3) as a function of the level of the noise σ_η for various number of acquisitions $N = 1,2,3,7,15,31,63$. The exposure parameter $k_{ex} = 0.75$ is the same as in Figure 2.

$s(x_1,x_2)$, and

$$p_{0s} = 1 - p_{1s} = \Pr\{y = 0 \,|\, s\} = F_\eta(\theta - s) \qquad (12)$$

determined by the noise $\eta(x_1,x_2)$ via its cumulative distribution function $F_\eta(\cdot)$.

For a 410×415 binary image with $p_1 = 0.3$, Figure 4 shows typical evolutions for information $I(s,y)$ as a function of the rms amplitude σ_η of the noise $\eta(x_1,x_2)$ chosen zero-mean Gaussian. In Figure 4, when $0 < \theta < 1$ the noise is felt only as a nuisance, and information $I(s,y)$ is maximum at zero noise and decreases when σ_η grows. Meanwhile, when $\theta > 1$ no information is transmitted in the absence of noise, and it is the increase of the noise level σ_η above zero which authorizes information $I(s,y)$ to grow in order to culminate at a maximum for a nonzero optimal amount of noise maximizing information transmission.

This noise-aided information transmission quantified in Figure 4 can be visually appreciated in Figure 5, with the optimal noise configuration in the middle image.

This example illustrates one basic form of noise-aided information processing, in which the noise has a constructive influence in a binary decision in the presence of a fixed discrimination threshold. This basic form can find applicability in many areas, and can be elaborated upon in many directions. It can be related to the effect of dithering known in imaging at low processing levels (Gammaitoni, 1995). It can also be related to threshold nonlinearities found in biophysical sensory processes, for instance originating in the retina for visual perception (Patel and Kosko, 2005). At high processing levels, the effect is relevant to pattern recognition in human vision (Piana et al., 2000).

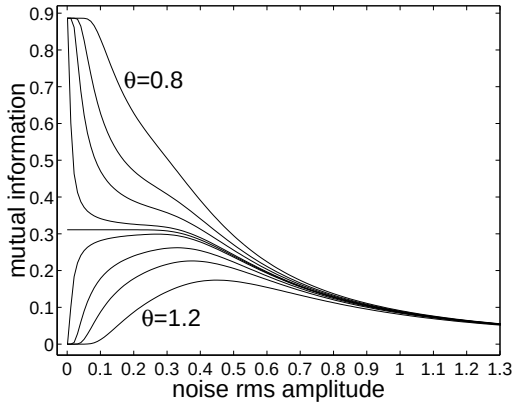


Figure 4: Mutual information $I(s,y)$, as a function of the noise rms amplitude σ_η, in succession for the threshold $\theta = 0.8, 0.9, 0.95, 0.99, 1, 1.01, 1.05, 1.1$ and 1.2.

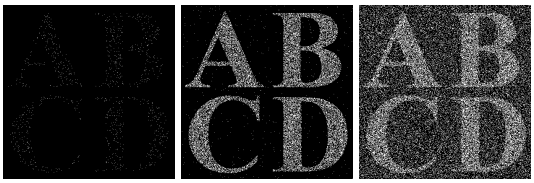


Figure 5: Output image $y(x_1,x_2)$ with threshold $\theta = 1.1$ and noise level $\sigma_\eta = 0.05$ (left), 0.4 (middle) and 1.3 (right).

REFERENCES

Andò, B. and Graziani, S. (2001). Adding noise to improve measurement. *IEEE Instrumentation and Measurement Magazine*, 4:24–30.

Chapeau-Blondeau, F. and Rousseau, D. (2002). Noise improvements in stochastic resonance: From signal amplification to optimal detection. *Fluctuation and Noise Letters*, 2:L221–L233.

Chapeau-Blondeau, F. and Rousseau, D. (2004). Noise-enhanced performance for an optimal Bayesian estimator. *IEEE Transactions on Signal Processing*, 52:1327–1334.

Gammaitoni, L. (1995). Stochastic resonance and the dithering effect in threshold physical systems. *Physical Review E*, 52:4691–4698.

Patel, A. and Kosko, B. (2005). Stochastic resonance in noisy spiking retinal and sensory neuron models. *Neural Networks*, 18:467–478.

Piana, M., Canfora, M., and Riani, M. (2000). Role of noise in image processing by the human perceptive system. *Physical Review E*, 62:1104–1109.

Stocks, N. G. (2000). Suprathreshold stochastic resonance in multilevel threshold systems. *Physical Review Letters*, 84:2310–2313.

Wiesenfeld, K. and Moss, F. (1995). Stochastic resonance and the benefits of noise: From ice ages to crayfish and SQUIDs. *Nature*, 373:33–36.

MULTICHANNEL FILTER FOR ENHANCEMENT OF SPEECH BLOCKS

Ivandro Sanches
Genius Instituto de Tecnologia, Manaus, Amazonas, Brazil
isanches@genius.org.br

Keywords: Speech, noise, microphone array.

Abstract: This work presents the concepts and the achieved results of a proposed microphone array algorithm based on multi-dimensional Wiener filter developed to work on blocks of speech. The inputs to the algorithm are two correlation matrices: the correlation matrix of the background noise affecting the desired signal and the correlation matrix of the signal affected by the noise. Experiments show that improvements of more than 12dB on signal to noise ratio can be achieved when comparing the filtered signals with one of the microphone array channels. In order to save computational load, the input signal is processed in blocks of a specified size and a technique is proposed to reduce blocking effects on the output filtered signal. It will be shown that practically there are no blocking effects. It is also shown that the technique is independent of the array physical configuration.

1 INTRODUCTION

Speech communication or recognition systems on embedded and other kinds of applications are demanding for effective ways of dealing with low signal to noise ratio (SNR) and the mobility of speakers (or even the mobility of applications, in the case of robots). Microphone array techniques play an important role in this scenario. This work presents a multichannel algorithm which significantly increases the SNR, copes with any microphone array geometry and may facilitate user's and application mobility.

Next section introduces the notation and describes the algorithm. Section 3 presents signal enhancement results when the technique is applied to simulated data and, then, data acquired in real conditions. Simulated data were used in order to show and simulate the independency on array physical configuration and to show the absence of blocking effects in the filtered signal.

2 ALGORITHM PRESENTATION

The proposed algorithm has some resemblance to (Florencio and Malvar, 2001) and (Doclo and Moonem, 2001). It differs from both in the sense that the input and output signals are processed in blocks of samples to considerably reduce the computational load. Analysis of the algorithm in hearing aid applications is presented in (Spriet, Moonen, and Wouters, 2005).

The notation used is presented next. It is assumed that speech, **s**, and affecting noise, **n**, are statistically uncorrelated, and that noise is linearly added to speech:

$$\mathbf{x} = \mathbf{s} + \mathbf{n}, \tag{1}$$

where **x** is the output from the N channels of the microphone array for a given frame analysis of L_S samples per channel:

$$\mathbf{x} = \begin{bmatrix} x_1(1) & x_1(2) & \cdots & x_1(L_S) \\ x_2(1) & x_2(2) & \cdots & x_2(L_S) \\ \vdots & \vdots & \ddots & \vdots \\ x_N(1) & x_N(2) & \cdots & x_N(L_S) \end{bmatrix}. \tag{2}$$

Our objective is to estimate the clean signal **s** given **x,** the noise statistics, and the filter order L. In general, we may not need to estimate **s**, but just one of the N rows of **s**. In the approach, without loss of generality, we attempt to estimate s_1, that is, the clean speech signal from channel 1. The algorithm has two correlation matrices as input, the

background noise correlation matrix $\mathbf{R}_N$ and the signal correlation matrix $\mathbf{R}_X$. The former is computed with L_N samples from each channel of the microphone array when there is no speech activity. Note that the bigger L_N is, the more statistics from noise are gathered at the cost of computational load to estimate $\mathbf{R}_N$. The correlation matrix $\mathbf{R}_X$, for a given filter order L, is computed from matrix $\mathbf{X}$ defined as:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \cdots & \mathbf{X}_N \end{bmatrix}, \tag{3}$$

where,

$$\mathbf{X}_i = \begin{bmatrix} x_i(1) & x_i(2) & \cdots & x_i(L) \\ x_i(2) & x_i(3) & \cdots & x_i(L+1) \\ \vdots & \vdots & \ddots & \vdots \\ x_i(L_S - L + 1) & x_i(L_S - L + 2) & \cdots & x_i(L_S) \end{bmatrix}, \quad 1 \le i \le N \tag{4}$$

Then, the correlation matrix $\mathbf{R}_X$, is computed from:

$$\mathbf{R}_X = \frac{\mathbf{X}^T . \mathbf{X}}{L_S - L + 1}, \tag{5}$$

where $\mathbf{X}^T$ is the transpose of $\mathbf{X}$. Matrix $\mathbf{R}_N$ is computed in similar fashion with L_N background noise samples per channel, instead.

The optimal multi-dimensional Wiener filter, $\mathbf{W}_{WF}$, can now be computed:

$$\mathbf{W}_{WF} = \mathbf{R}_X^{-1}(\mathbf{R}_X - \mathbf{R}_N), \tag{6}$$

as presented in (Florencio and Malvar, 2001), matrix $\mathbf{R}_X^{-1}$ above can be replaced by $(\mathbf{R}_X + \rho \mathbf{R}_N)^{-1}$, where $\rho \ge 0$. Increasing ρ improves intelligibility at a cost of increasing signal distortion.

The filtered signal matrix can then be computed from

$$\mathbf{Y} = \mathbf{W}_{WF} . \mathbf{X}^T . \tag{7}$$

It can be seen that matrix $\mathbf{Y}$ is $(NL)\times(L_S\text{-}L\text{+}1)$. Every L rows from $\mathbf{Y}$ correspond to a filtered estimate of a specific channel from the array, and they can be conveniently grouped to form an improved filtered estimate from the specific channel. Grouping L consecutive filtered signals is possible when it is noticed that each one of the L rows is shifted by just one sample from the next row. Equation 8 presents the grouping process resulting in the output filtered signal of length $L_S\text{-}L(N\text{+}1)\text{+}2$ corresponding to the estimation of s_1,

$$y_1(n) = \frac{\sum_{i=1}^{L} \mathbf{Y}[i][NL - i + n]}{L}, \quad 1 \le n \le L_S - L(N+1) + 2, \tag{8}$$

where $\mathbf{Y}[i][j]$ is the $\mathbf{Y}$ element on row i and column j. Figure 1 illustrates the time relative positions among frames and the length of the filtered signals in $\mathbf{Y}$ and in y_1 compared to the original frame length. The algorithm then proceeds taking the next L_S input samples per channel after an input shift of $L_S\text{-}L(N\text{+}1)\text{+}2$ samples.

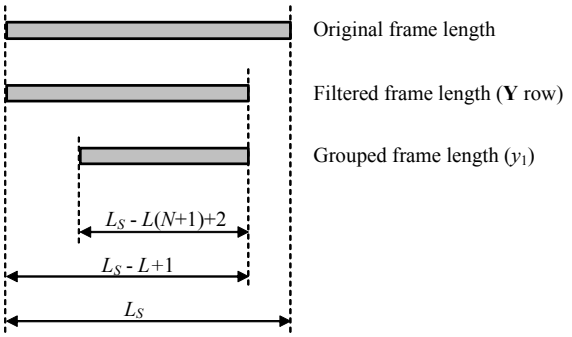


Figure 1: Lengths of the original analysis frame, filtered frame and grouped frame.

As an example, when applying the algorithm in a speech recognition experiment, one may wish that the length of the filtered vector y_1 be around 20ms at a frame rate of 10ms. For that end, assuming sampling frequency f_S kHz, the following must be satisfied:

$$20 f_S = L_S - L(N+1) + 2 . \tag{9}$$

To help with the definitions, one can further assume the constraint that the filtered signal y_1 is half of the original frame length L_S, resulting an L_S corresponding to 40ms. These assumptions and constraints provide a way to determine the value of L, the filter order:

$$L = \mathbf{round}\left(\frac{20 f_S + 2}{N+1}\right). \tag{10}$$

Thus, for instance, when $N = 2$ microphones and $f_S = 8$kHz, the filter order is $L = 54$, and $L_S = 320$ samples.

More generally, equation 8 can be rewritten for channel j, $1 \le j \le N$:

$$y_j(n) = \frac{\sum_{i=1}^{L} \mathbf{Y}[i + (j-1)L][NL - i + n - (j-1)L]}{L}, \quad 1 \le n \le L_S - L(N+1) + 2. \tag{11}$$

3 EXPERIMENTS

This section presents experimental results that show the performance of the proposed algorithm on simulated data as well as data acquired in real conditions.

3.1 Simulated Data

This section presents the algorithm acting on simulated signals in order to explore the algorithm behaviour in respect to blocking effects and independence on the array configuration, that is, it will be shown that the algorithm does not require that the signal be acquired from a perfectly symmetric array. Two experiments will be presented in this section.

The first experiment explores how the algorithm deals with blocking effect. For that end, it was simulated a 4-channel (4 microphones) signal affected by omnidirectional noise at a signal to noise ratio (SNR) of 0dB. Signals sampling frequency is 8kHz. Every channel has an initial period of noise and then a 100Hz sine wave starts. Noise statistics are obtained from the beginning 100ms of the signal (no sine wave present). Sine waves from adjacent channels are shifted by 30 degrees. Analysis frame duration of the input signal is 40ms. Frame duration of the output filtered signal is 20ms, thus blocking effects would happen at this rate (every 2 cycles of the sine wave). The affecting noise is a Gaussian random noise uncorrelated among channels, which is not a condition that happens on real applications, where noise is correlated among channels (the next experiment will show a condition where noise is highly correlated among channels).

Figure 2 presents 60ms of the described signals. There are three plots in this figure. The first plot presents the clean signal. It can be seen that the sine wave period is 10ms, corresponding to 100Hz. The second plot shows the noisy signal, which is formed from the addition of the clean 4-channel sine wave signal to the 4-channel noise signal. The third plot presents the filtered signal corresponding to every channel of the array (see equation 11). The discontinuities at 0.01s on the clean signal, first plot, cause a transition region on the filtered signal, third plot, of about 20ms, after which there is no visual evidence of blocking effect, since the filtered signal is fairly continuous. This was also confirmed analyzing the remaining seconds of the filtered signal. Figure 3 presents in more detail the results for channel 1 only. The first plot compares directly the input clean signal to the filtered signal. The second plot presents channel 1 noisy signal, which is one of the inputs to the algorithm.

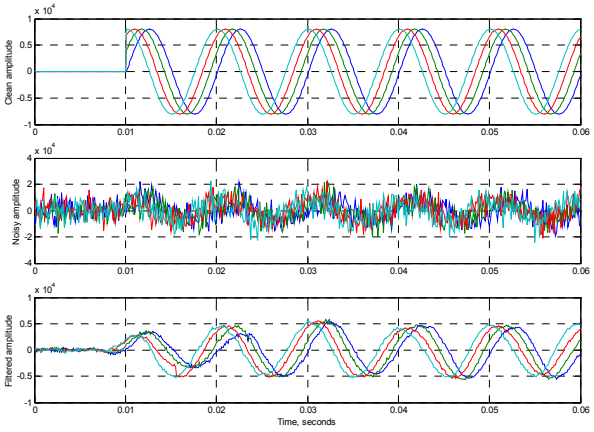


Figure 2: Plot 1 presents a 4-channel 100Hz clean sine wave signal. Every adjacent channel is shifted by 30 degrees. Plot 2 is the result of adding omnidirectional Gaussian noise at 0dB SNR, producing the noisy signal input to the algorithm. Plot 3 is the output filtered signal corresponding to each input noisy channel.

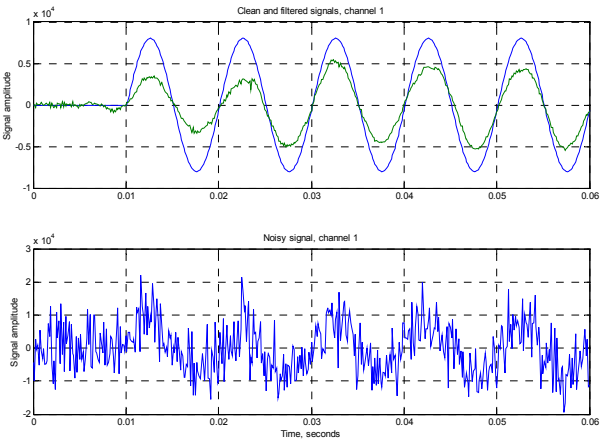


Figure 3: Channel 1 extracted from figure 2. The first plot compares channel 1 clean signal to the corresponding filtered signal. The second plot presents the actual channel 1 input noisy signal.

The second experiment, illustrated in figure 4, aims to observe the behaviour of the algorithm in an eventual asymmetric array configuration. Producing different phase shifts between adjacent channels simulates this. In the example, the clean signal phase shifts from channel 1 are 30, 90 and 180 degrees. Likewise, the noise signal channels have different phase shifts. From channel 1, the phase shifts on the noise channels are –20, -50 and –90 degrees. As before, the clean signal is composed of 100Hz sine waves, while the noise signal is now formed with

500Hz sine waves at 0dB SNR. It can be seen on figure 4 third plot that the algorithm coped conveniently with the different phase shifts imposed on the clean and noise signals. It can be noticed that the phase shift among input channels is preserved among the output filtered channels. And, again, no blocking effect can be detected. Figure 5 presents with more detail channel 1 clean signal directly compared to the filtered channel 1 (first plot) and the input noisy signal (second plot).

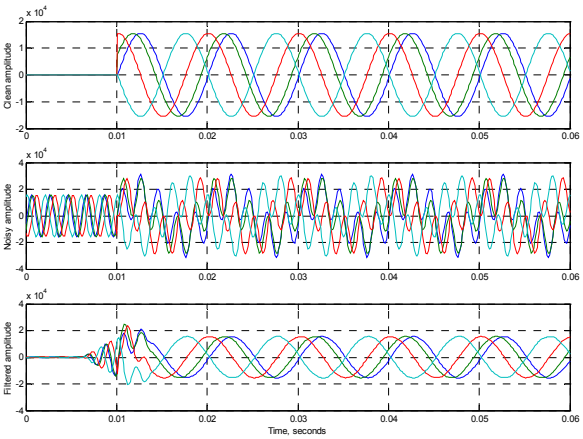


Figure 4: Experiment to show the independence of the algorithm to asymmetries on array configuration.

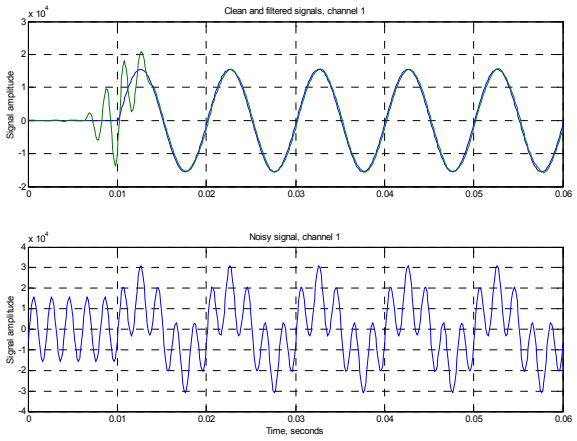


Figure 5: The first plot compares channel 1 clean signal to the corresponding filtered signal from figure 4. Second plot presents the actual channel 1 input noisy signal.

3.2 Real Data

The speech data used in this experiment was acquired from a microphone array with four omnidirectional microphones spaced by 15cm. The signals were acquired at a sampling frequency of 48kHz. In this experiment the signals were decimated to 16kHz. The speaker was about 1m from the microphones. The environment was a room in the speaker's house. An engine background noise can be heard when the corresponding audio file from one of the channels is played. Figure 6 first plot presents the signal from one channel of the microphone array. The SNR at this channel is 4.3dB. Figure 6 second plot shows the output from the proposed algorithm. The SNR at the filtered signal is 32.3dB. Both SNR's were computed by the NIST signal to noise estimation utility (quick method; see References section below). Note that the noise from the first 300ms from the filtered signal is more attenuated than the remaining of the noise portion, since the first 400ms from the noisy input was used to compute the noise correlation matrix, $\mathbf{R}_N$. Input frames of 40ms (L_S=640, L=64) were used to compute the signal correlation matrix, $\mathbf{R}_X$, at every 20ms interval. Filtered output frames of 20ms (320 samples) were produced and concatenated. Listening to this signal, it is realized that the engine background noise was completely removed.

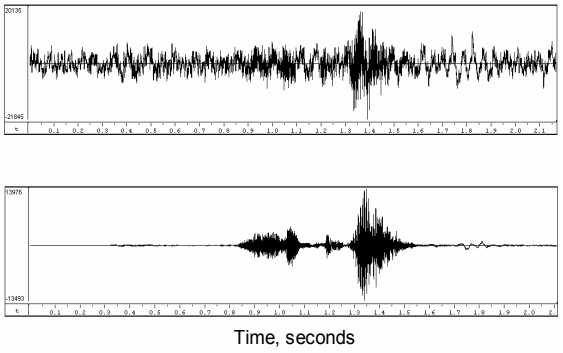


Figure 6: Experiment with real data. The first plot shows one channel from the microphone array. The second plot presents the corresponding algorithm output.

Figure 7 presents in more detail the time interval from 0.8s to 1.2s. This interval corresponds to a sound like '**she**'.

3 CONCLUSION

This work presented a successful algorithm based on multi-dimensional Wiener filter, suitable to work with microphone arrays of any physical configuration. It was shown that, although the algorithm works with blocks of signal, in order to reduce computational load, blocking effects are not perceptible. It is worth mentioning that from the speech recognition point of view, coupling the

microphone array to the speech recognition front-end, blocking effect is not an issue when it is realized that the front-end works with blocks (frames) of speech. If no optimizations are applied, mainly in the solution of equations 5, 6 and 7, algorithm computational complexity is high, about $(NL)^3+(L_S-L)(NL)^2$ flops for each block of output signal (e.g., 4.4Mflops for 20ms of filtered speech with $N = 2$ microphones, $f_S = 8$kHz, $L = 54$, and $L_S = 320$ samples). Future efforts should be focused on this issue, exploring matrices symmetries and positive definiteness. As an example, the computation of $\mathbf{R}_X$ can go from about $(L_S-L)(NL)^2$ to about $N(N+1)(L_SL+3L^2+5L)$ flops. The independence on the array physical configuration coupled with the computation of every channel best estimate may be conveniently applied on speech recognition tasks where microphones are spread in a room environment, and the channel with the best SNR is chosen as input to the speech recognition process, extending speaker's mobility. The next steps will be to investigate the performance of the algorithm on speech recognition experiments.

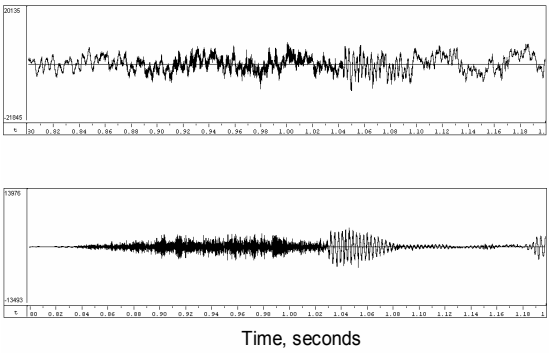


Figure 7: Excerpt from figure 6 signals, between 0.8s and 1.2s. This interval corresponds to a sound like **'she'**.

ACKNOWLEDGEMENTS

This work had the financial support from FUNTTEL and FINEP, respectively the Fund for Technological Development of Telecommunications, of the Brazilian Ministry of Communications, and the Fostering Agency of Studies and Projects, of the Brazilian Ministry of Science and Technology, under the contract 01.02.0066-00.

REFERENCES

Doclo, S. and Moonem, M. (2001). *Microphone Arrays: Signal Processing Techniques and Applications*, chapter GSVD-Based Optimal Filtering for Multi-Microphone Speech Enhancement. Springer-Verlag, Berlin.

Florencio, D., and Malvar, H. (2001). Multichannel filtering for optimum noise reduction microphone arrays. In *Proc. ICASSP, volume 1, pages 197-200.*

NIST, National Institute of Standards and Technology, http://www.nist.gov/speech/tools/index.htm, accessed January 10, 2007.

Spriet, A., Moonen, M., and Wouters, J. (2005). Robustness analysis of multichannel Wiener filtering and generalized sidelobe cancellation for multimicrophone noise reduction in hearing aid applications. In *IEEE Trans. on Speech and Audio Processing, volume 13, pages 487-503.*

ROBUST CONTROL OF HYSTERETIC BASE-ISOLATED STRUCTURES UNDER SEISMIC DISTURBANCES

Francesc Pozo, José Rodellar
CoDAlab, Universitat Politècnica de Catalunya, Comte d'Urgell, 187, 08036 Barcelona, Spain
francesc.pozo@upc.edu, jose.rodellar@upc.edu

Leonardo Acho, Ricardo Guerra
Centro de Investigación y Desarrollo de Tecnología Digital, Instituto Politécnico Nacional
Avenida del Parque, 1310, 22510 Tijuana, Baja California, Mexico
leonardo.acho@upc.edu, inge_guerra@hotmail.com

Keywords: Base-isolated structures, seismic disturbances, active control.

Abstract: The main objective of applying robust active control to base-isolated structures is to protect them in the event of an earthquake. Taking advantage of discontinuous control theory, a static discontinuous active control is developed using as a feedback only the measure of the velocity at the base. Moreover, due to that in many engineering applications, accelerometers are the only devices that provide information available for feedback, our velocity feedback controller could be easily extended by using just acceleration information through a filter. The main contributions of this paper are: (a) a static velocity feedback controller design, and (b) a dynamic acceleration feedback controller design, for seismic attenuation of structures. Robustness performance is analyzed by means of numerical experiments using the 1940 *El Centro* earthquake.

1 INTRODUCTION

Base isolation has been widely considered as an effective technology to protect flexible structures up to eight storeys high against earthquakes. The conceptual objective of the isolator is to produce a dynamic decoupling of the structure from its foundation so that the structure ideally behaves like a rigid body with reduced inter-story drifts, as demanded by safety, and reduced absolute accelerations as related to comfort requirements. Although the response quantities of a fixed-base building are reduced substantially through base isolation, the base displacement may be excessive, particularly during near-field ground motions (Yang and Agrawal, 2002). Applications of hybrid control systems consisting of active or semi-active systems installed in parallel to base-isolation bearings have the capability to reduce response quantities of base-isolated structures more significantly than passive dampers (Ramallo et al., 2002; Yang and Agrawal, 2002).

In this paper, two versions of a decentralized robust active control are developed and applied to a base-isolated structure. The first controller uses the velocity at the base of the structure as feedback information, and it is analyzed via Lyapunov stability techniques as proposed in (Luo et al., 2001). Due to the fact that, in civil engineering applications, accelerometers are the most practically available sensors for feedback control, the second controller is an extension of the first one where just acceleration information is used. Performance of the proposed controllers, for seismic attenuation, are evaluated by numerical simulations using the 1940 El Centro earthquake (California, United States).

This paper is structured as follows. Section 2 describes the problem formulation. The solution to the problem statement using just velocity measurements is described in Section 3, meanwhile the solution employing only acceleration information is stated in Section 4. Numerical simulations to analyze the performance of both proposed controllers are presented in Section 5. Finally, on Section 6 final comments are stated.

2 PROBLEM STATEMENT

Consider a basic forced vibration system governed by:

$$m\ddot{x} + c\dot{x} + \Phi(x,t) = f(t) + u(t), \qquad (1)$$

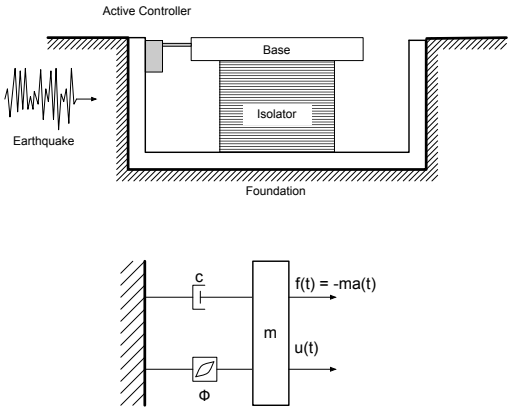


Figure 1: Building structure with hybrid control system (up) and physical model (down).

where m is the mass; c is the damping coefficient; Φ is the restoring force characterizing the hysteretic behavior of the isolator material, which is usually made with inelastic rubber bearings; $f(t)$ is the unknown excitation force; and $u(t)$ is the control force supplied by an appropriate actuator.

In structure systems, $f(t) = -m\ddot{x}_g(t)$ is the excitation force, where $\ddot{x}_g(t)$ is the earthquake ground acceleration. The restoring force Φ can be represented by the Bouc-Wen model (Ikhouane et al., 2005) in the following form:

$$\Phi(x,t) = \alpha Kx(t) + (1-\alpha)DKz(t) \quad (2)$$

$$\dot{z} = D^{-1}\left(A\dot{x} - \beta|\dot{x}||z|^{n-1}z - \lambda\dot{x}|z|^n\right) \quad (3)$$

where $\Phi(x,t)$ can be considered as the superposition of an elastic component αKx and a hysteretic component $(1-\alpha)DKz(t)$, in which the yield constant displacement is $D > 0$ and $\alpha \in [0,1]$ is the post- to pre-yielding stiffness ratio. $n \geq 1$ is a scalar that governs the smoothness of the transition from elastic to plastic response and $K > 0$. The hysteretic part in (2) involves an internal dynamic (3) which is unmeasurable, and thus inaccessible for seismic control design. A schematic description of the base-isolated system structure and its physical model are displayed in Fig. 1.

The following assumptions are stated for system (1)-(3):

Assumption 1 *The acceleration disturbance $f(t) = -m\ddot{x}_g$ is unknown but bounded;* i.e.*, there exists a known constant F such that $|f(t)| \leq F, \quad \forall t \geq 0$.*

Assumption 2 *In the event of an earthquake, it is assumed that $z(0) = 0$ in equation (1) and that the structure is at rest;* i.e.*, $x(0) = \dot{x}(0) = 0$.*

Assumption 3 *There exists a known upper bound on the internal dynamic variable $z(t)$,* i.e.*, $|z(t)| \leq \bar{\rho}_z, \quad \forall t \geq 0$.*

Assumption 1 is standard in control of hysteretic systems or base-isolated structures (Ikhouane et al., 2005). Assumption 2 has a physical meaning because it is assumed that the structure is at rest when the earthquake strikes it. The upper bound in $z(t)$ expressed in Assumption 3 is computable, independently on the boundedness of $x(t)$ by invoking Theorem 1 in (Ikhouane et al., 2005).

Control objective: Our objective is to design a robust controller for system (1) such that, under earthquake attack, the trajectories of the closed-loop remain bounded.

To this end, the theorems in the following sections satisfy this control objective.

3 SEISMIC ATTENUATION USING ONLY VELOCITY FEEDBACK

Theorem 1 *Consider the nonlinear system (1)-(3) subject to Assumptions 1-3. Then, the following control law*

$$u = -\rho sgn(\dot{x}_0) \quad (4)$$

solves the control objective if

$$\rho \geq (1-\alpha)DK\bar{\rho}_z + F. \quad (5)$$

Proof. The closed-loop system (1)-(3) and (4) yields

$$\begin{aligned} m_0\ddot{x}_0 + c_0\dot{x}_0 + k_0x_0 + \Phi(x_0,t) &= -m_0\ddot{x}_g - \rho\mathrm{sgn}(\dot{x}_0) \\ m_0\ddot{x}_0 + c_0\dot{x}_0 + (k_0+\alpha K)x_0 &= -\rho\mathrm{sgn}(\dot{x}_0) + \Delta(z,t) \end{aligned} \quad (6)$$

where

$$\Delta(z,t) = -m_0\ddot{x}_g - (1-\alpha)Dkz.$$

Then

$$\begin{aligned} |\Delta(z,t)| &\leq |f(t)| + |(1-\alpha)DKz| \\ &\leq F + (1-\alpha)DK|z| \\ &\leq F + (1-\alpha)DK\bar{\rho}_z = \rho_1. \end{aligned}$$

Given the Lyapunov function

$$V(x_0,\dot{x}_0) = \frac{k_0+\alpha K}{2}x_0^2 + \frac{m_0}{2}\dot{x}_0^2,$$

its time derivative along the trajectories of the closed-loop system (1)-(3) and (4) yields

$$\begin{aligned}\dot{V}(x_0,\dot{x}_0) &= (k_0+\alpha K)x_0\dot{x}_0 + m_0\dot{x}_0\ddot{x}_0\\ &= \dot{x}_0\left[(k_0+\alpha K)x_0 + m_0\ddot{x}_0\right]\\ &= \dot{x}_0\left[-c_0\dot{x}_0 - \rho\,\mathrm{sgn}(\dot{x}_0) + \Delta(z,t)\right]\\ &= -c_0\dot{x}_0^2 - \rho\dot{x}_0\,\mathrm{sgn}(\dot{x}_0) + \dot{x}_0\Delta(z,t)\\ &= -c_0\dot{x}_0^2 - \rho|\dot{x}_0| + \dot{x}_0\Delta(z,t)\\ &\leq -c_0\dot{x}_0^2 - \rho|\dot{x}_0| + |\dot{x}_0|\rho_1\\ &= -c_0\dot{x}_0^2 + (\rho_1-\rho)|\dot{x}_0|.\end{aligned}$$

The choice of $\rho \geq \rho_1$ makes $\dot{V}$ negative semidefinite, as we wanted to show. ■

Remark 1 (on solution of non-smooth systems) *The closed-loop system (6) has a non-smooth right-hand side, the signum function. Solutions to this non-smooth class of systems in the sense of Filippov has been widely studied (Wu et al., 1998). It is worth noting that non-smooth dynamic systems appear naturally and frequently in many mechanical systems (Wu et al., 1998). Due to the fact that classical solution theories to ordinary differential equations require vector fields to be at least Lipschitz continuous, main difficulties with non-smooth systems are that these systems fail the Lipschitz-continuous requirement. However, if (a) the vector field is measurable and essentially bounded; (b) the solution of the system is absolutely continuous; and (c) the Lyapunov function V is continuous and positive definite and its time derivative $\dot{V}$ along the trajectories of the closed-loop system is continuous and negative semi-definite, then the system under consideration has a solution in the sense of Filippov and it is stable in the sense of Lyapunov (Wu et al., 1998). This is exactly our case.*

Remark 2 *The signum function in the control law in Theorem 1 –common in sliding mode control theory– produces chattering (Utkin, 1982; Edwards and Spurgeon, 1998). One way to avoid chattering is to replace the signum function by a smooth sigmoid-like function such as*

$$\nu_\delta(s) = \frac{s}{|s|+\delta},$$

where δ is a sufficiently small positive scalar (Edwards and Spurgeon, 1998).

Consequently, the following Corollary is stated:

Corollary 1 *Consider the nonlinear system (1)-(3) subject to Assumptions 1-3. Then, the following control law*

$$u = -\rho\frac{\dot{x}_0}{|\dot{x}_0|+\delta} \tag{7}$$

solves the control objective if

$$\rho \geq (1-\alpha)DK\bar{\rho}_z + F$$

and δ is a sufficiently small positive scalar.

Proof. The time derivative of the Lyapunov function

$$V(x_0,\dot{x}_0) = \frac{k_0+\alpha K}{2}x_0^2 + \frac{m_0}{2}\dot{x}_0^2,$$

along the trajectories of the closed-loop system (1)-(3) and (7) yields

$$\begin{aligned}\dot{V} &= (k_0+\alpha K)x_0\dot{x}_0 + m_0\dot{x}_0\ddot{x}_0\\ &= \dot{x}_0\left[(k_0+\alpha K)x_0 + m_0\ddot{x}_0\right]\\ &= \dot{x}_0\left[-c_0\dot{x}_0 - \rho\frac{\dot{x}_0}{|\dot{x}_0|+\delta} + \Delta(z,t)\right]\\ &= -c_0\dot{x}_0^2 - \rho\dot{x}_0\frac{\dot{x}_0}{|\dot{x}_0|+\delta} + \dot{x}_0\Delta(z,t)\\ &= -c_0\dot{x}_0^2 - \rho\frac{\dot{x}_0^2}{|\dot{x}_0|+\delta} + \dot{x}_0\Delta(z,t)\\ &\leq -c_0\dot{x}_0^2 + \rho_1|\dot{x}_0| - \rho\frac{\dot{x}_0^2}{|\dot{x}_0|+\delta}\\ &= -c_0\dot{x}_0^2 - (\rho-\rho_1)|\dot{x}_0| + \rho\left(|\dot{x}_0| - \frac{\dot{x}_0^2}{|\dot{x}_0|+\delta}\right).\end{aligned}$$

The objective of guaranteeing global boundedness of solutions is equivalently expressed as rendering $\dot{V}$ negative outside a compact region. The choice of $\rho \geq \rho_1$ and considering that

$$\lim_{\delta\to 0}\rho\left(|\dot{x}_0| - \frac{\dot{x}_0^2}{|\dot{x}_0|+\delta}\right) = 0$$

guarantees the existence of a small compact region $D \subset \mathbb{R}^2$ (depending on δ) such that $\dot{V}$ is negative outside this set. This implies that all the closed-loop trajectories remain bounded, as we wanted to show. ■

4 SEISMIC ATTENUATION USING ONLY ACCELERATION FEEDBACK

Motivated by the fact that in many civil engineering applications accelerometers are the only devices that provide information available for feedback, Theorem 2 (below) presents a control law based on equation (4) where only acceleration information is required.

Theorem 2 *Consider the nonlinear system (1)-(3) subject to Assumptions 1-3. Then, the following control law*

$$u = -\rho sgn(\upsilon) \tag{8}$$

$$\dot{\upsilon} = \ddot{x}_0 \tag{9}$$

solves the control objective if

$$\rho \geq (1-\alpha)DK\bar{\rho}_z + F.$$

Proof. This proof is straightforward by considering direct integration of equation (9). ■

Remark 3 *In the practical implementation of this control law,* ν *may drift due to unmodeled dynamics, measure errors and disturbance. To avoid this, the following* σ*-modification (Ioannou and Kokotovic, 1983; Koo and Kim, 1994) can be used,*

$$u = -\rho sgn(\upsilon), \tag{10}$$

$$\dot{\upsilon} = -\sigma\nu + \ddot{x}_0, \tag{11}$$

where σ *is a positive constant.*

As in the previous Section, a smooth version of the control law in equations (10)-(11) is considered in the following Corollary.

Corollary 2 *Consider the nonlinear system (1)-(3) subject to Assumptions 1-3. Then, the following control law*

$$u = -\rho \frac{\upsilon}{|\upsilon| + \delta} \tag{12}$$

$$\dot{\upsilon} = -\sigma\upsilon + \ddot{x}_0 \tag{13}$$

solves the control objective if

$$\rho \geq (1-\alpha)DK\bar{\rho}_z + F,$$

where $\sigma > 0$ *and* δ *are sufficiently small positive scalar.*

5 NUMERICAL SIMULATIONS

In order to investigate the efficiency of the proposed controllers, we set $m = 156 \times 10^3$ kg, $K = 6 \times 10^6$ N/m, $c = 2 \times 10^4$ Ns/m, $\alpha = 0.6$, $D = 0.6$ m, $\lambda = 0.5$, $\beta = 0.1$, $n = 3$, and $A = 1$ (Ikhouane et al., 2005). A set of numerical experiments was performed on the system using information recorded during the 1940 El Centro earthquake. Figure 2 shows the ground acceleration for this earthquake. The open loop base displacement can also be seen in Figure 2. It can be seen that $\rho = 2 \cdot 10^5$ is an upper bound for the expression $(1-\alpha)DK\bar{\rho}_z + F$ in equation (5).

Figures 3, 4 and 5 display the time histories of the motion of the base and the control signal force for different values of ρ and δ, when the control law in equation (7) is used. In an equivalent manner, the time histories of the motion of the base and the control signal force when the control law in equations (12)-(13) is used are depicted in Figures 6, 7 and 8. In both cases, the controlled base displacements are significantly reduced compared to the uncontrolled case. It is worth noting that, when $\sigma = 0.1$ in Figure 8, the results are similar to those in Figure 3.

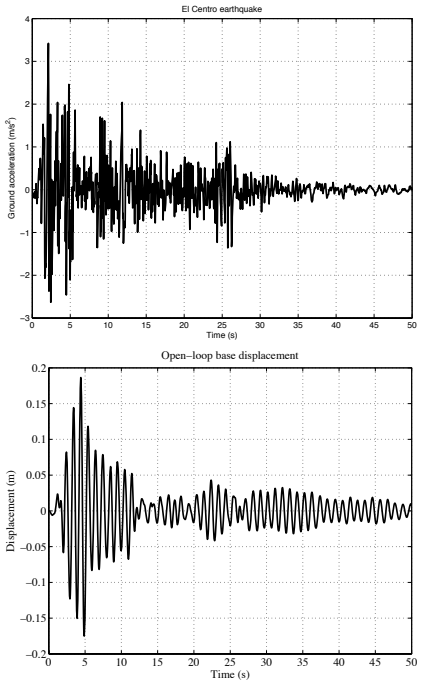


Figure 2: 1940 El Centro earthquake, ground acceleration (top); open loop base displacement (bottom).

6 CONCLUSION

A robust control scheme to attenuate the consequences of seismic events on base-isolated structures has been proposed. It has been shown that a simple controller can fulfill the control objectives, using just velocity measurements or just acceleration information. Simulation results showed the good performance of the controllers. In civil engineering, the controller that just uses acceleration information is of a great interest, due to the fact that accelerometers are easily available. Also, the simplicity of the proposed controllers makes them attractive for a real implementation.

REFERENCES

Edwards, C. and Spurgeon, S. K. (1998). *Sliding Mode Control. Theory and Applications*. Taylor and Francis, London.

Ikhouane, F., Mañosa, V., and Rodellar, J. (2005). Adaptive control of a hysteretic structural system. *Automatica*, 41(2):225–231.

Ioannou, P. and Kokotovic, P. (1983). *Adaptive System with Reduced Models*. Springer-Verlag, New York.

Koo, K.-M. and Kim, J.-H. (1994). Robust control of robot manipulators with parametric uncertainty. *IEEE*

Transactions on Automatic Control, 39(6):1230–1233.

Li, H. and Ou, J. (2006). A design approach for semi-active and smart base-isolated buildings. *Structural Control and Health Monitoring*, 13(2):660–681.

Luo, N., Rodellar, J., Vehí, J., and De la Sen, M. (2001). Composite semiactive control of a class of seismically excited structures. *Journal of The Franklin Institute*, 338:225–240.

Makris, N. (1997). Rigidity–plasticity–viscosity: can electrorheological dampers protect base-isolated structures from near-source ground motions? *Earthquake Engineering and Structural Dynamics*, 26(5):571–591.

Pozo, F., Ikhouane, F., Pujol, G., and Rodellar, J. (2006). Adaptive backstepping control of hysteretic base-isolated structures. *Journal of Vibration and Control*, 12(4):373–394.

Ramallo, J., Yoshioka, M., and Spencer, B. (2004). A two-step identification technique for semiactive control systems. *Structural Control and Health Monitoring*, 11(4):273–279.

Ramallo, J. C., Johnson, E. A., and Spencer, B. F. (2002). Smart base isolation systems. *Journal of Engineering Mechanics*, 128(10):1088–1099.

Smyth, A., Masri, S., Kosmatopoulos, E., Chassiakos, A., and Caughey, T. (2002). Development of adaptive modeling techniques for non-linear hysteretic systems. *International Journal of Non-Linear Mechanics*, 37(8):1435–1451.

Spencer, B., Dyke, S., Sain, M., and Carlson, J. (1997). Phenomenological model for magnetorheological dampers. *Journal of Engineering Mechanics - Proceedings of the ASCE*, 123(3):230–238.

Spencer, B. and Sain, M. (1997). Controlling buildings: a new frontier in feedback. *IEEE Control Systems Magazine*, 17(6):19–35.

Spooner, J., Ordóñez, R., Maggiore, M., and Passino, K. (2001). *Stable Adaptive Control and Estimation for Nonlinear Systems: Neural and Fuzzy Approximation Techniques*. John Wiley & Sons, Inc., New York, NY, USA.

Utkin, V. (1982). *Sliding Modes in Control and Optimization*. Springer-Verlag, Berlin.

Wu, Q., Onyshko, S., Sepheri, N., and Thornton-Trump, A. B. (1998). On construction of smooth lyapunov functions for non-smooth systems. *International Journal of Control*, 69(3):443–457.

Yang, J. N. and Agrawal, A. K. (2002). Semi-active hybrid control systems for nonlinear buildings against near-field earthquakes. *Engineering Structures*, 24(3):271–280.

Yoshioka, H., Ramallo, J., and Spencer, B. (2002). Smart base isolation strategies employing magnetorheological dampers. *Journal of Engineering Mechanics - Proceedings of the ASCE*, 128(5):540–551.

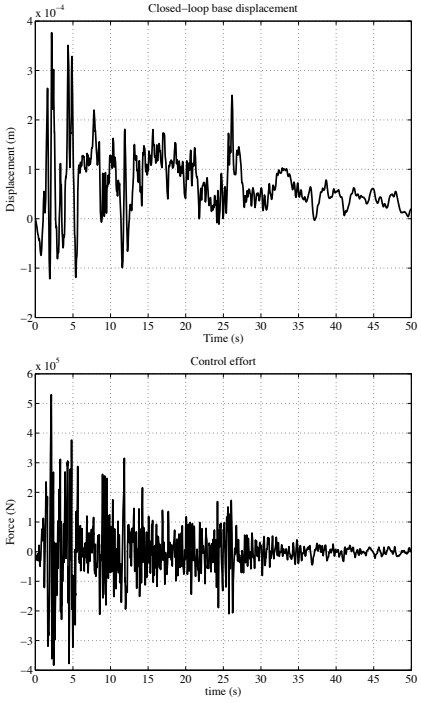


Figure 3: Closed loop base displacement (top) and control signal force (bottom) with control law in equation (7) and parameters $\rho = 2 \cdot 10^6$ and $\delta = 0.01$.

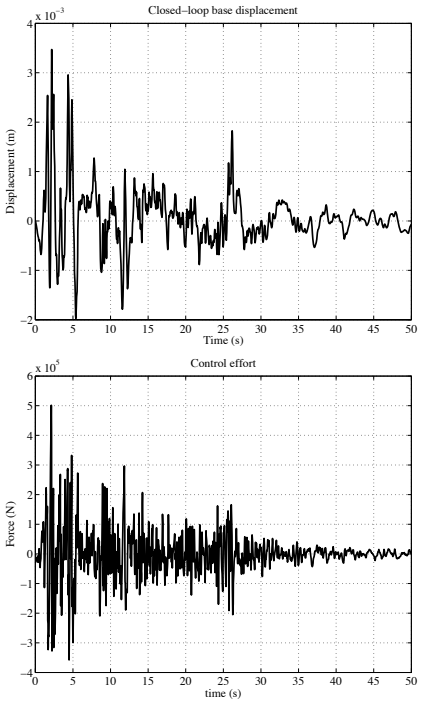


Figure 4: Closed loop base displacement (top) and control signal force (bottom) with control law in equation (7) and parameters $\rho = 2 \cdot 10^6$ and $\delta = 0.1$.

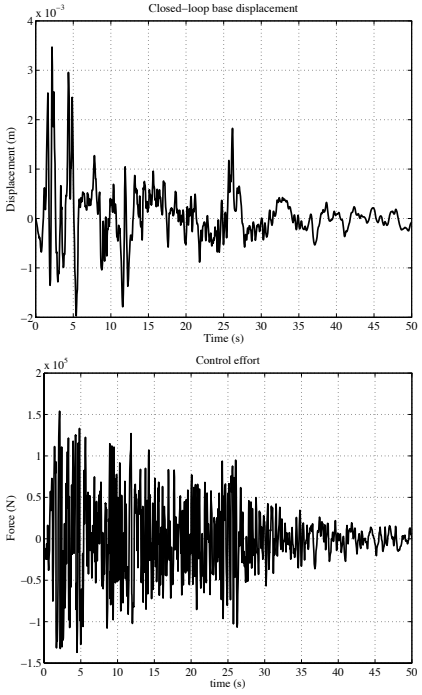


Figure 5: Closed loop base displacement (top) and control signal force (bottom) with control law in equation (7) and parameters $\rho = 2 \cdot 10^5$ and $\delta = 0.01$.

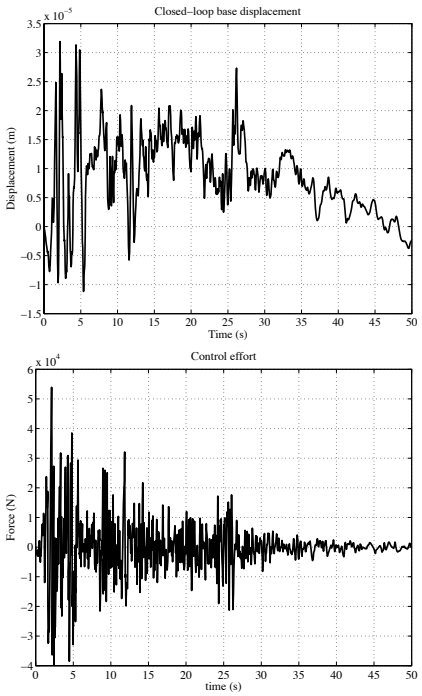


Figure 6: Closed loop base displacement (top) and control signal force (bottom) with control law in equations (12)-(13) and parameters $\rho = 2 \cdot 10^6$, $\delta = 0.01$ and $\sigma = 0.1$.

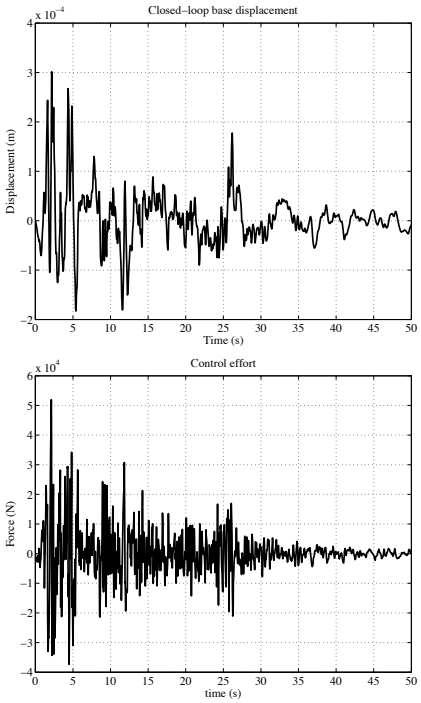


Figure 7: Closed loop base displacement (top) and control signal force (bottom) with control law in equations (12)-(13) and parameters $\rho = 2 \cdot 10^6$, $\delta = 0.1$ and $\sigma = 0.1$.

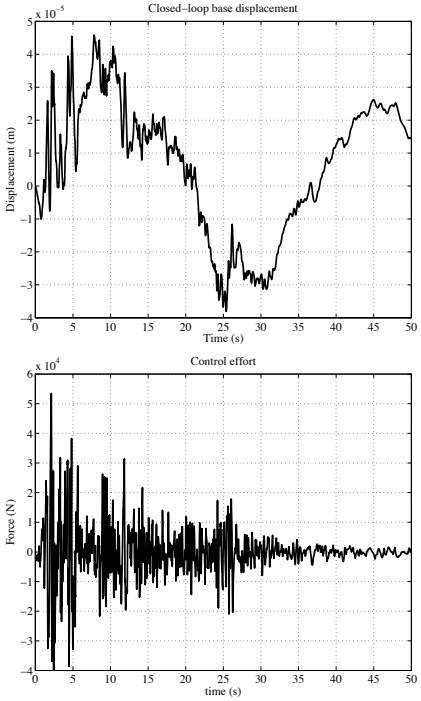


Figure 8: Closed loop base displacement (top) and control signal force (bottom) with control law in equations (12)-(13) and parameters $\rho = 2 \cdot 10^6$, $\delta = 0.01$ and $\sigma = 1$.

IMPROVED ROBUSTNESS OF MULTIVARIABLE MODEL PREDICTIVE CONTROL UNDER MODEL UNCERTAINTIES

Cristina Stoica, Pedro Rodríguez-Ayerbe and Didier Dumur
Department of Automatic Control, Supélec, 3 rue Joliot Curie, F91192 Gif-sur-Yvette, France
cristina.stoica@supelec.fr, pedro.rodriguez@supelec.fr, didier.dumur@supelec.fr

Keywords: Model Predictive Control (MPC), Multivariable Systems, Linear Matrix Inequalities, Robust Control.

Abstract: This paper presents a state-space methodology for enhancing the robustness of multivariable MPC controlled systems through the convex optimization of a multivariable Youla parameter. The procedure starts with the design of an initial stabilizing Model Predictive Controller in the state-space representation, which is then robustified under modeling errors considered as unstructured uncertainties. The resulting robustified MIMO control law is finally applied to the model of a stirred tank reactor to reduce the impact of measurement noise and modelling errors on the system.

1 INTRODUCTION

Model predictive control strategies are widely used in industrial applications, resulting in improved performance, with a practical implementation of the controller which remains simple. However, starting with a controller design based on a 'nominal' model of the system, the question of its robustness towards model uncertainties or disturbances acting on the system always occurs in an industrial environment.

Some methods in the literature deal with robustness maximisation, but in the transfer function formalism (Kouvaritakis et al., 1992), (Yoon and Clarke, 1995), (Dumur and Boucher, 1998), and mainly applied to SISO systems, which makes the generalization to multivariable systems much more complicated.

The purpose of this paper is to present a methodology enhancing the robustness of an initial MIMO predictive controller towards model uncertainties. The state-space design allows the robustification process to be handled in a convenient way. A two-step procedure is followed. An initial MIMO MPC controller is first designed, its robustness is then enhanced via the Youla parametrization, without significantly increasing the complexity of the final control law. The Youla parametrization allows formulating frequency constraints as convex optimization, the entire problem being solved with LMI (Linear Matrix Inequality) techniques.

The paper is organized as follows. Section 2 reminds the main steps leading to the MPC controller in the state-space representation. Section 3 gives the background material required to formulate the robustification strategy, from the Youla parametrization to the robustness criteria under unstructured uncertainties. The elaboration of the robustified controller in state-space representation for this type of uncertainties is further proposed in Section 4. Section 5 provides the application of this control strategy to a stirred tank reactor. Section 6 presents some conclusions and further perspectives.

2 MIMO MPC IN STATE-SPACE FORMULATION

This section focuses on the design of an initial MIMO MPC law. Compared to approaches proposed in the literature based on transfer function formalism, the state-space representation framework chosen here (Camacho and Bordons, 2004) leads to a simplified formulation and reduced computation efforts for MIMO systems. Consider the following discrete time MIMO LTI system:

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\,\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \\ \mathbf{y}(k) = \mathbf{C}\,\mathbf{x}(k) \end{cases} \quad (1)$$

where $\mathbf{A} \in \mathbf{R}^{n\times n}$, $\mathbf{B} \in \mathbf{R}^{n\times m}$, $\mathbf{C} \in \mathbf{R}^{p\times n}$ are the system state-space matrices, $\mathbf{x} \in \mathbf{R}^{n\times 1}$ describes the MIMO system states, $\mathbf{u} \in \mathbf{R}^{m\times 1}$ is the input vector and $\mathbf{y} \in \mathbf{R}^{p\times 1}$ is the output vector.

Next step is to add an integral action to this state-space representation which will guarantee cancellation of steady-state errors:

$$\mathbf{u}(k) = \mathbf{u}(k-1) + \Delta\mathbf{u}(k) \quad (2)$$

This results in an increase of the system states as:

$$\begin{cases}\mathbf{x}_e(k+1)=\mathbf{A}_e\,\mathbf{x}_e(k)+\mathbf{B}_e\,\Delta\mathbf{u}(k)\\ \mathbf{y}(k)=\mathbf{C}_e\,\mathbf{x}_e(k)\end{cases} \tag{3}$$

where the extended state-space representation $\mathbf{x}_e(k)=\left[\mathbf{x}^T(k)\ \ \mathbf{u}^T(k-1)\right]^T$ is characterized by:
$\mathbf{A}_e=\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{0}_{m,n} & \mathbf{I}_m\end{bmatrix}$, $\mathbf{B}_e=\begin{bmatrix}\mathbf{B}\\ \mathbf{I}_m\end{bmatrix}$, $\mathbf{C}_e=\begin{bmatrix}\mathbf{C} & \mathbf{0}_{p,m}\end{bmatrix}$.

The control signal is derived by minimizing the following quadratic objective function:

$$J=\sum_{i=N_1}^{N_2}\left\|\hat{\mathbf{y}}(k+i)-\mathbf{w}(k+i)\right\|^2_{\widetilde{\mathbf{Q}}_J(i)}+\sum_{i=0}^{N_u-1}\left\|\Delta\mathbf{u}(k+i)\right\|^2_{\widetilde{\mathbf{R}}_J(i)} \tag{4}$$

where the future control increments $\Delta\mathbf{u}(k+i)$ are supposed to be zero for $i\geq N_u$. The signal $\mathbf{w}$ represents the setpoint. It is assumed in further developments that the same output prediction horizons (N_1, N_2) and the same control horizon N_u is applied for all input/output transfer functions. $\widetilde{\mathbf{Q}}_J$ and $\widetilde{\mathbf{R}}_J$ are weighting matrices. The predicted output vector has the following form:

$$\hat{\mathbf{y}}(k+i)=\mathbf{C}\,\mathbf{A}^i\hat{\mathbf{x}}(k)+\sum_{j=0}^{i-1}\mathbf{C}\,\mathbf{A}^{i-j-1}\mathbf{B}\mathbf{u}(k+j) \tag{5}$$

where the input vector can be written as:

$$\mathbf{u}(k+j)=\mathbf{u}(k-1)+\sum_{l=0}^{j}\Delta\mathbf{u}(k+l) \tag{6}$$

The state estimate is derived from the observer:

$$\hat{\mathbf{x}}_e(k+1)=\mathbf{A}_e\,\hat{\mathbf{x}}_e(k)+\mathbf{B}_e\,\Delta\mathbf{u}(k)+\mathbf{K}[\mathbf{y}(k)-\mathbf{C}_e\,\hat{\mathbf{x}}_e(k)] \tag{7}$$

The multivariable observer gain $\mathbf{K}$ is designed through a classical method of eigenvectors, arbitrarily placing the eigenvalues of $\mathbf{A}_e-\mathbf{K}\mathbf{C}_e$ in a stable region, as detailed in (Magni, 2002). The observer gain $\mathbf{K}$ is obtained from the extended state-space description and will be used for further mathematical calculation in the robustification procedure. However this design aspect is not crucial since the convex robustification method should lead to an optimal set of these eigenvalues. Moreover the input/output transfer function is not influenced by the eigenvalues placement used to find $\mathbf{K}$ (Boyd and Barratt, 1991).

The objective function can be rewritten in the matrix formalism (Maciejowski, 2001):

$$J=\left\|\mathbf{Y}(k)-\mathbf{W}(k)\right\|^2_{\mathbf{Q}_J}+\left\|\Delta\mathbf{U}(k)\right\|^2_{\mathbf{R}_J} \tag{8}$$

where $\mathbf{Q}_J=diag(\widetilde{\mathbf{Q}}_J(N_1),\cdots,\widetilde{\mathbf{Q}}_J(N_2))$,
$\mathbf{R}_J=diag(\widetilde{\mathbf{R}}_J(0),\cdots,\widetilde{\mathbf{R}}_J(N_u-1))$,
$\mathbf{Y}(k)=\left[\hat{\mathbf{y}}^T(k+N_1)\ \cdots\ \hat{\mathbf{y}}^T(k+N_2)\right]^T$,
$\mathbf{W}(k)=\left[\mathbf{w}^T(k+N_1)\ \cdots\ \mathbf{w}^T(k+N_2)\right]^T$,
$\Delta\mathbf{U}(k)=\left[\Delta\mathbf{u}^T(k)\ \cdots\ \Delta\mathbf{u}^T(k+N_u-1)\right]^T$.

Using these notations, the output vector $\mathbf{Y}(k)$ can be written in the following matrix form, with the definition of the vector $\mathbf{\Theta}(k)$ as a tracking error:

$$\mathbf{Y}(k)=\mathbf{\Psi}\,\hat{\mathbf{x}}(k)+\mathbf{\Phi}\,\mathbf{u}(k-1)+\mathbf{\Phi}_\Delta\Delta\mathbf{U}(k) \tag{9}$$

$$\mathbf{\Theta}(k)=\mathbf{W}(k)-\mathbf{\Psi}\,\hat{\mathbf{x}}(k)-\mathbf{\Phi}\,\mathbf{u}(k-1) \tag{10}$$

with $\mathbf{\Psi}=\left[(\mathbf{C}\mathbf{A}^{N_1})^T\ \cdots\ (\mathbf{C}\mathbf{A}^{N_2})^T\right]^T$,

$\mathbf{\Phi}=\left[\mathbf{\Sigma}^T_{N_1-1}\ \cdots\ \mathbf{\Sigma}^T_{N_2-1}\right]^T$, $\mathbf{\Sigma}_i=\mathbf{C}\sum_{j=0}^{i}\mathbf{A}^{i-j}\mathbf{B}$, $\mathbf{\Sigma}_i^T=(\mathbf{\Sigma}_i)^T$,

$$\mathbf{\Phi}_\Delta=\begin{bmatrix}\mathbf{\Sigma}_{N_1-1} & \cdots & \mathbf{\Sigma}_0 & 0 & \cdots & 0\\ \vdots & \cdots & \vdots & \vdots & \ddots & \vdots\\ \mathbf{\Sigma}_{N_2-1} & \cdots & \mathbf{\Sigma}_{N_2-N_1} & \mathbf{\Sigma}_{N_2-N_1-1} & \cdots & \mathbf{\Sigma}_{N_2-N_u}\end{bmatrix}.$$

The objective function is now given by:

$$J=\left\|\mathbf{\Phi}_\Delta\Delta\mathbf{U}(k)-\mathbf{\Theta}(k)\right\|^2_{\mathbf{Q}_J}+\left\|\Delta\mathbf{U}(k)\right\|^2_{\mathbf{R}_J} \tag{11}$$

which analytical minimization provides:

$$\Delta\mathbf{U}(k)=(\mathbf{R}_J+\mathbf{\Phi}_\Delta^{\mathrm{T}}\mathbf{Q}_J\mathbf{\Phi}_\Delta)^{-1}\mathbf{\Phi}_\Delta^{\mathrm{T}}\,\mathbf{Q}_J\,\mathbf{\Theta}(k) \tag{12}$$

Applying the receding horizon principle, only the first component of each future control sequence is applied to the system, meaning that the first m lines of $\Delta\mathbf{U}(k)$ are used:

$$\Delta\mathbf{u}(k)=\boldsymbol{\mu}\,\mathbf{\Theta}(k) \tag{13}$$

with $\boldsymbol{\mu}=\left[\mathbf{I}_m\ \ \mathbf{0}_{m,m(N_u-1)}\right](\mathbf{R}_J+\mathbf{\Phi}_\Delta^{\mathrm{T}}\,\mathbf{Q}_J\,\mathbf{\Phi}_\Delta)^{-1}\mathbf{\Phi}_\Delta^{\mathrm{T}}\,\mathbf{Q}_J$.

The system model, the observer and the predictive control can be represented in the state-space formulation according to Figure 1. The control signal depends on the control gain $\mathbf{L}=\left[\mathbf{L}_1\ \ \mathbf{L}_2\right]$ and the setpoint filter $\mathbf{F}_w$:

$$\Delta\mathbf{u}(k)=\mathbf{F}_w\mathbf{w}(k)-\mathbf{L}\,\hat{\mathbf{x}}_e(k) \tag{14}$$

with $\mathbf{L}_1=\boldsymbol{\mu}\mathbf{\Psi}$, $\mathbf{L}_2=\boldsymbol{\mu}\mathbf{\Phi}$, $\mathbf{F}_w=diag(\mathbf{F}_{w,1},\cdots,\mathbf{F}_{w,p})$ related to the structure of $\boldsymbol{\mu}$ and $\mathbf{w}(k)$.

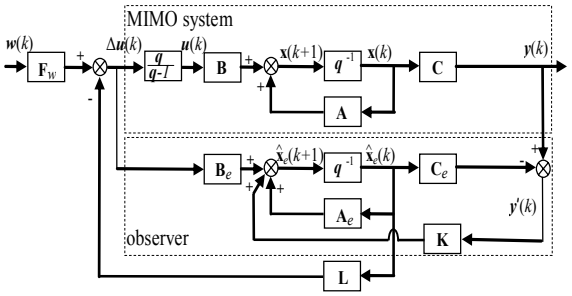


Figure 1: Block diagram of MIMO MPC.

3 ROBUSTNESS USING THE YOULA PARAMETER

This section overviews a technique that improves the robustness of the previous multivariable MPC law in terms of the Youla parameter, also named **Q** parameter. Any stabilizing controller (Boyd and Barratt, 1991), (Maciejowski, 1989) can be represented by a state-space feedback controller coupled with an observer and a Youla parameter. This part focuses on the main steps leading to the multivariable **Q** parameter (here with p inputs and m outputs) that robustifies the MPC law described in Section 2.

3.1 Stabilizing Control Law

The whole class of stabilizing control law can be obtained from an initial stabilizing controller via the Youla parametrization. The first step considers additional inputs $\mathbf{u}'$ and outputs $\mathbf{y}'$ with a zero transfer between them ($\mathbf{T}_{22} = 0$ in Figure 2).

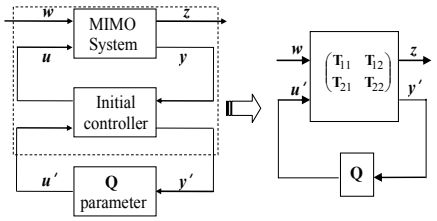


Figure 2: Class of all stabilizing multivariable controllers.

The Youla parameter is then added between $\mathbf{y}'$ and $\mathbf{u}'$ without restricting closed-loop stability. In this case, the transfer from **u** to **y** remains unchanged. As a result, the closed-loop function between **w** and **z** is linearly parametrized by the **Q** parameter, allowing convex specification (Boyd and Barratt, 1991):

$$\mathbf{T_{zw}} = \mathbf{T}_{11\,\mathbf{zw}} + \mathbf{T}_{12\,\mathbf{zw}}\,\mathbf{Q}\,\mathbf{T}_{21\,\mathbf{zw}} \qquad (15)$$

where $\mathbf{T}_{11}, \mathbf{T}_{12}, \mathbf{T}_{21}$ depends on the input vector **w** and output vector **z** considered.

3.2 Robustness Under Frequency Constraints

Practical applications always deal with neglected dynamics and potential disturbances, so that robustness under unstructured uncertainties must be addressed as shown in Figure 3.

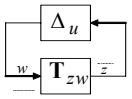


Figure 3: Unstructured uncertainty.

According to the small gain theorem (Maciejowski, 1989), robustness under unstructured uncertainties Δ_u is maximized as:

$$\min_{\mathbf{Q}\in\Re H_\infty} \left\| \mathbf{T_{zw}} \mathbf{W}_T \right\|_\infty \qquad (16)$$

where the weighting term $\mathbf{W}_T$ reflects the frequency range where model uncertainties are more important. For multivariable systems, the H_∞ norm can be calculated as the maximum of the higher singular values. The following theorem formulates the previous H_∞ norm minimization.

Theorem (Clement and Duc, 2000) and (Boyd et al., 1994): A discrete time system given by the state-space representation $(\mathbf{A}_{cl}, \mathbf{B}_{cl}, \mathbf{C}_{cl}, \mathbf{D}_{cl})$ is stable and admits a H_∞ norm lower than γ if and only if:

$$\exists \mathbf{X}_1 = \mathbf{X}_1^{\mathrm{T}} > 0 / \begin{bmatrix} -\mathbf{X}_1^{-1} & \mathbf{A}_{cl} & \mathbf{B}_{cl} & \mathbf{0} \\ \mathbf{A}_{cl}^{\mathrm{T}} & -\mathbf{X}_1 & \mathbf{0} & \mathbf{C}_{cl}^{\mathrm{T}} \\ \mathbf{B}_{cl}^{\mathrm{T}} & \mathbf{0} & -\gamma\mathbf{I} & \mathbf{D}_{cl}^{\mathrm{T}} \\ \mathbf{0} & \mathbf{C}_{cl} & \mathbf{D}_{cl} & -\gamma\mathbf{I} \end{bmatrix} < 0 \qquad (17)$$

This expression can be transformed into a LMI, which variables are $\mathbf{X}_1$, γ and the **Q** parameter included in the closed-loop matrices, as shown in (Clement and Duc, 2000). As a result, the optimization problem is formulated as the minimization of γ under this LMI constraint.

4 ROBUSTIFIED MIMO MPC

The previous robustification strategy based on the Youla parameter is now applied to an initial MIMO state-space MPC calculated as shown in Section 2. The robustness maximization under additive unstructured uncertainties is also equivalent to the minimization of the influence of a measurement noise **b** on the control signal **u** (Figure 4); the

transfer (15) between $\mathbf{w}$ and $\mathbf{z}$ corresponds to the transfer from $\mathbf{b}$ to $\mathbf{u}$. The H_∞ norm of this transfer will be further minimized using LMI tools.

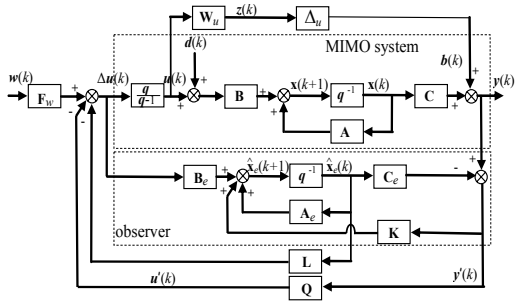


Figure 4: Stabilizing MIMO MPC via **Q** parametrization.

4.1 Stabilizing Control Law

Consider the MIMO linear discrete time system in the state-space representation, including an integral action (3). After adding an auxiliary input vector $\mathbf{u}'$ and output vector $\mathbf{y}'$ (Figure 4), the multivariable control signal is computed as described in Section 2:

$$\Delta\mathbf{u}(k) = \mathbf{F}_w\mathbf{w}(k) - \mathbf{L}\,\hat{\mathbf{x}}_e(k) - \mathbf{u}'(k) \tag{18}$$

with the following observer:

$$\begin{aligned}\hat{\mathbf{x}}_e(k+1) = \mathbf{A}_e\,\hat{\mathbf{x}}_e(k) + \mathbf{B}_e\,\Delta\mathbf{u}(k) + \\ + \mathbf{K}[\mathbf{y}(k) - \mathbf{C}_e\,\hat{\mathbf{x}}_e(k) + \mathbf{b}(k)]\end{aligned} \tag{19}$$

To calculate the closed-loop transfer function, the initial state is increased, adding the prediction error:

$$\boldsymbol{\varepsilon}(k) = \mathbf{x}_e(k) - \hat{\mathbf{x}}_e(k) \tag{20}$$

Considering only the terms related to $\mathbf{b}(k)$ as they are part of the minimization process, the following state-space system is derived:

$$\begin{bmatrix}\mathbf{x}_e(k+1)\\ \boldsymbol{\varepsilon}(k+1)\end{bmatrix} = \begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_3\\ \mathbf{0} & \mathbf{A}_2\end{bmatrix}\begin{bmatrix}\mathbf{x}_e(k)\\ \boldsymbol{\varepsilon}(k)\end{bmatrix} + \begin{bmatrix}\mathbf{0} & -\mathbf{B}_e\\ -\mathbf{K} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{b}(k)\\ \mathbf{u}'(k)\end{bmatrix} \tag{21}$$

$$\mathbf{y}'(k) = \begin{bmatrix}\mathbf{0} & \mathbf{C}_e\end{bmatrix}\begin{bmatrix}\mathbf{x}_e(k)\\ \boldsymbol{\varepsilon}(k)\end{bmatrix} + \begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{b}(k)\\ \mathbf{u}'(k)\end{bmatrix} \tag{22}$$

with $\mathbf{A}_1 = \mathbf{A}_e - \mathbf{B}_e\mathbf{L}$, $\mathbf{A}_2 = \mathbf{A}_e - \mathbf{K}\mathbf{C}_e$, $\mathbf{A}_3 = \mathbf{B}_e\mathbf{L}$.

According to the theory given in Section 3.1, the Youla parameter can be added to robustify the initial controller, since the transfer between $\mathbf{y}'(k)$ and $\mathbf{u}'(k)$ is zero (without measurement noise, the multivariable output $\mathbf{y}'$ depends only on $\boldsymbol{\varepsilon}(k)$, which is independent from $\mathbf{x}_e(k)$ and $\mathbf{u}'(k)$).

4.2 Robustness Under Frequency Constraints

Next step is the definition of the weighting $\mathbf{W}_u$ as a diagonal high-pass filter in state-space formulation:

$$\begin{cases}\mathbf{x}_w(k+1) = \mathbf{A}_w\mathbf{x}_w(k) + \mathbf{B}_w\mathbf{u}(k)\\ \mathbf{z}(k) = \mathbf{C}_w\mathbf{x}_w(k) + \mathbf{D}_w\mathbf{u}(k)\end{cases} \tag{23}$$

Including the $\mathbf{W}_u$ weighting, a new extended state-space description can be emphasized:

$$\begin{bmatrix}\bar{\mathbf{x}}_1(k+1)\\ \boldsymbol{\varepsilon}(k+1)\end{bmatrix} = \begin{bmatrix}\bar{\mathbf{A}}_1 & \bar{\mathbf{A}}_3\\ \mathbf{0} & \mathbf{A}_2\end{bmatrix}\begin{bmatrix}\bar{\mathbf{x}}_1(k)\\ \boldsymbol{\varepsilon}(k)\end{bmatrix} + \begin{bmatrix}\mathbf{0} & -\bar{\mathbf{B}}_{u'}\\ -\mathbf{K} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{b}(k)\\ \mathbf{u}'(k)\end{bmatrix} \tag{24}$$

$$\begin{bmatrix}\mathbf{z}(k)\\ \mathbf{y}'(k)\end{bmatrix} = \begin{bmatrix}\bar{\mathbf{C}}_1 & \bar{\mathbf{C}}_2\\ \mathbf{0} & \mathbf{C}_e\end{bmatrix}\begin{bmatrix}\bar{\mathbf{x}}_1(k)\\ \boldsymbol{\varepsilon}(k)\end{bmatrix} + \begin{bmatrix}\mathbf{0} & -\mathbf{D}_w\\ \mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{b}(k)\\ \mathbf{u}'(k)\end{bmatrix} \tag{25}$$

with $\bar{\mathbf{x}}_1(k) = \left[\mathbf{x}^T(k)\ \mathbf{u}^T(k-1)\ \mathbf{x}_w^T(k)\right]^T$, $\bar{\mathbf{B}}_{u'} = \left[\mathbf{B}^T\ \mathbf{I}\ \mathbf{B}_w^T\right]^T$

$$\bar{\mathbf{A}}_1 = \begin{bmatrix}\mathbf{A}-\mathbf{B}\mathbf{L}_1 & \mathbf{B}-\mathbf{B}\mathbf{L}_2 & \mathbf{0}\\ -\mathbf{L}_1 & \mathbf{I}-\mathbf{L}_2 & \mathbf{0}\\ -\mathbf{B}_w\mathbf{L}_1 & \mathbf{B}_w(\mathbf{I}-\mathbf{L}_2) & \mathbf{A}_w\end{bmatrix},\ \bar{\mathbf{A}}_3 = \begin{bmatrix}\mathbf{B}\mathbf{L}\\ \mathbf{L}\\ \mathbf{B}_w\mathbf{L}\end{bmatrix},$$

$$\bar{\mathbf{C}}_1 = \left[-\mathbf{D}_w\mathbf{L}_1\ \ \mathbf{D}_w(\mathbf{I}-\mathbf{L}_2)\ \ \mathbf{C}_w\right],\ \bar{\mathbf{C}}_2 = \mathbf{D}_w\mathbf{L}.$$

As described in Section 3.2, a multivariable Youla parameter $\mathbf{Q} \in \Re H_\infty$ is added for robustification purposes leading to a convex optimization problem. Since this problem leads to a $\mathbf{Q}$ parameter which varies in the infinite-dimensional space $\Re H_\infty$, a sub-optimal solution considers for each input/output pairs (i, j) a finite-dimensional subspace generated by an orthonormal base of discrete stable transfer functions such as a polynomial or FIR filter:

$$Q^{ij} = \sum_{l=0}^{n_Q} q_l^{ij} q^{-l} \tag{26}$$

In the state-space formalism, this MIMO Youla parameter can be obtained using a fixed pair $(\mathbf{A}_Q \in \mathbf{R}^{pn_Q \times pn_Q}, \mathbf{B}_Q \in \mathbf{R}^{pn_Q \times p})$ and designing only the variable pair $(\mathbf{C}_Q \in \mathbf{R}^{m \times pn_Q}, \mathbf{D}_Q \in \mathbf{R}^{m \times p})$:

$$\begin{cases}\mathbf{x}_Q(k+1) = \mathbf{A}_Q\mathbf{x}_Q(k) + \mathbf{B}_Q\mathbf{y}'(k)\\ \mathbf{u}'(k) = \mathbf{C}_Q\mathbf{x}_Q(k) + \mathbf{D}_Q\mathbf{y}'(k)\end{cases} \tag{27}$$

with $\mathbf{a}_Q = \begin{bmatrix}\mathbf{0}_{1,n_Q-1} & 0\\ \mathbf{I}_{n_Q-1} & \mathbf{0}_{n_Q-1,1}\end{bmatrix}$, $\mathbf{b}_Q = \begin{bmatrix}1\\ \mathbf{0}_{n_Q-1,1}\end{bmatrix}$, $\mathbf{c}_Q^{ij} = \begin{bmatrix}q_1^{ij}\\ \vdots\\ q_{n_Q}^{ij}\end{bmatrix}$

$$\mathbf{C}_Q = \begin{bmatrix}\mathbf{c}_Q^{11} & \cdots & \mathbf{c}_Q^{1p}\\ \vdots & \ddots & \vdots\\ \mathbf{c}_Q^{m1} & \cdots & \mathbf{c}_Q^{mp}\end{bmatrix},\ \mathbf{D}_Q = \begin{bmatrix}q_0^{11} & \cdots & q_0^{1p}\\ \vdots & \ddots & \vdots\\ q_0^{m1} & \cdots & q_0^{mp}\end{bmatrix},$$

$\mathbf{A}_Q = diag(\mathbf{a}_Q, \cdots, \mathbf{a}_Q)$, $\mathbf{B}_Q = diag(\mathbf{b}_Q, \cdots, \mathbf{b}_Q)$.

Adding this Youla parameter leads to the following closed-loop state-space description:

$$\begin{cases}\mathbf{x}_{cl}(k+1) = \mathbf{A}_{cl}\mathbf{x}_{cl}(k) + \mathbf{B}_{cl}\mathbf{b}(k)\\ \mathbf{z}(k) = \mathbf{C}_{cl}\mathbf{x}_{cl}(k) + \mathbf{D}_{cl}\mathbf{b}(k)\end{cases} \tag{28}$$

with $\mathbf{x}_{cl} = \left[\bar{\mathbf{x}}_1^T\ \ \boldsymbol{\varepsilon}^T\ \ \mathbf{x}_Q^T\right]^T$,

$$\mathbf{C}_{cl}=\left[\overline{\mathbf{C}}_1\ \ \overline{\mathbf{C}}_2-\mathbf{D}_w\mathbf{D}_Q\mathbf{C}_e\ \ -\mathbf{D}_w\mathbf{C}_Q\right],\ \mathbf{D}_{cl}=-\mathbf{D}_w\mathbf{D}_Q,$$

$$\mathbf{A}_{cl}=\begin{bmatrix}\overline{\mathbf{A}}_1 & \overline{\mathbf{A}}_3-\overline{\mathbf{B}}_{u'}\mathbf{D}_Q\mathbf{C}_e & -\overline{\mathbf{B}}_{u'}\mathbf{C}_Q\\ \mathbf{0} & \mathbf{A}_2 & \mathbf{0}\\ \mathbf{0} & \mathbf{B}_Q\mathbf{C}_e & \mathbf{A}_Q\end{bmatrix},\ \mathbf{B}_{cl}=\begin{bmatrix}-\overline{\mathbf{B}}_{u'}\mathbf{D}_Q\\ -\mathbf{K}\\ \mathbf{B}_Q\end{bmatrix}.$$

This state-space representation is the crucial point of the robustification method. With the result of the theorem in Section 3, the first step to transform (17) into a LMI consists in multiplying it to the right and to the left with positive definite matrices $\mathbf{\Pi}=\mathrm{diag}(\mathbf{X}_1,\mathbf{I},\mathbf{I},\mathbf{I})$ and $\mathbf{\Pi}^{\mathrm{T}}$ as in (Clement and Duc, 2000). This leads to the following inequality:

$$\begin{bmatrix}-\mathbf{X}_1 & \mathbf{X}_1\mathbf{A}_{cl} & \mathbf{X}_1\mathbf{B}_{cl} & \mathbf{0}\\ \mathbf{A}_{cl}^{\mathrm{T}}\mathbf{X}_1 & -\mathbf{X}_1 & \mathbf{0} & \mathbf{C}_{cl}^{\mathrm{T}}\\ \mathbf{B}_{cl}^{\mathrm{T}}\mathbf{X}_1 & \mathbf{0} & -\gamma\mathbf{I} & \mathbf{D}_{cl}^{\mathrm{T}}\\ \mathbf{0} & \mathbf{C}_{cl} & \mathbf{D}_{cl} & -\gamma\mathbf{I}\end{bmatrix}<0 \qquad (29)$$

which is not yet a LMI because terms such as $\mathbf{X}_1\mathbf{A}_{cl}$ and $\mathbf{X}_1\mathbf{B}_{cl}$ are not linear in $\mathbf{X}_1$, $\mathbf{C}_Q$ and $\mathbf{D}_Q$. To overcome this problem, the following bijective substitution is introduced (Clement and Duc, 2000):

$$\begin{cases}\mathbf{R}^{n\times n}\to\mathbf{R}^{n\times n}\\ \mathbf{X}_1=\left[\begin{array}{c|c}\mathbf{W}_1 & \mathbf{Z}_1\\ \hline \mathbf{Z}_1^{\mathrm{T}} & \mathbf{Y}_1\end{array}\right]\mapsto\left[\begin{array}{c|c}\mathbf{R}_1 & \mathbf{S}_1\\ \hline \mathbf{S}_1^{\mathrm{T}} & \mathbf{T}_1\end{array}\right]=\left[\begin{array}{c|cc}\mathbf{R}_1 & \mathbf{S}_{11} & \mathbf{S}_{12}\\ \hline \mathbf{S}_{11}^{T} & \mathbf{T}_{11} & \mathbf{T}_{12}\\ \mathbf{S}_{12}^{T} & \mathbf{T}_{12}^{T} & \mathbf{T}_{22}\end{array}\right]\end{cases} \qquad (30)$$

with $\mathbf{R}_1=\mathbf{W}_1^{-1}$, $\mathbf{S}_1=-\mathbf{W}_1^{-1}\mathbf{Z}_1$, $\mathbf{T}_1=\mathbf{Y}_1-\mathbf{Z}_1^{\mathrm{T}}\mathbf{W}_1^{-1}\mathbf{Z}_1$.

Next step to the LMI is to multiply (29) on the right with $\mathbf{\Gamma}=\mathrm{diag}\left(\begin{bmatrix}\mathbf{R}_1 & \mathbf{0}\\ \mathbf{S}_1^{\mathrm{T}} & \mathbf{I}\end{bmatrix},\begin{bmatrix}\mathbf{R}_1 & \mathbf{0}\\ \mathbf{S}_1^{\mathrm{T}} & \mathbf{I}\end{bmatrix},\mathbf{I},\mathbf{I}\right)$ and on the left with $\mathbf{\Gamma}^{\mathrm{T}}$. After technical manipulations, the following LMI is obtained:

$$\left[\begin{array}{ccc|ccc|c|c}-\mathbf{R}_1 & \mathbf{0} & \mathbf{0} & \overline{\mathbf{A}}_1\mathbf{R}_1 & t_1 & t_4 & t_7 & \mathbf{0}\\ * & -\mathbf{T}_{11} & -\mathbf{T}_{12} & \mathbf{0} & t_2 & t_5 & t_8 & \mathbf{0}\\ * & * & -\mathbf{T}_{22} & \mathbf{0} & t_3 & t_6 & t_9 & \mathbf{0}\\ \hline * & * & * & -\mathbf{R}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & t_{10}\\ * & * & * & * & -\mathbf{T}_{11} & -\mathbf{T}_{12} & \mathbf{0} & t_{11}\\ * & * & * & * & * & -\mathbf{T}_{22} & \mathbf{0} & t_{12}\\ \hline * & * & * & * & * & * & -\gamma\mathbf{I} & t_{13}\\ \hline * & * & * & * & * & * & * & -\gamma\mathbf{I}\end{array}\right]<0 \qquad (31)$$

where $t_1=\overline{\mathbf{A}}_1\mathbf{S}_{11}-\mathbf{S}_{11}\mathbf{A}_2-\mathbf{S}_{12}\mathbf{B}_Q\mathbf{C}_e+\overline{\mathbf{A}}_3-\overline{\mathbf{B}}_{u'}\mathbf{D}_Q\mathbf{C}_e$,

$t_2=\mathbf{T}_{11}\mathbf{A}_2+\mathbf{T}_{12}\mathbf{B}_Q\mathbf{C}_e$, $t_3=\mathbf{T}_{12}^{\mathrm{T}}\mathbf{A}_2+\mathbf{T}_{22}\mathbf{B}_Q\mathbf{C}_e$,

$t_4=\overline{\mathbf{A}}_1\mathbf{S}_{12}-\mathbf{S}_{12}\mathbf{A}_Q-\overline{\mathbf{B}}_{u'}\mathbf{C}_Q$, $t_5=\mathbf{T}_{12}\mathbf{A}_Q$, $t_6=\mathbf{T}_{22}\mathbf{A}_Q$,

$t_7=-\overline{\mathbf{B}}_{u'}\mathbf{D}_Q+\mathbf{S}_{11}\mathbf{K}-\mathbf{S}_{12}\mathbf{B}_Q$, $t_8=-\mathbf{T}_{11}\mathbf{K}+\mathbf{T}_{12}\mathbf{B}_Q$,

$t_9=-\mathbf{T}_{12}^{\mathrm{T}}\mathbf{K}+\mathbf{T}_{22}\mathbf{B}_Q$, $t_{10}=\mathbf{R}_1\overline{\mathbf{C}}_1^{\mathrm{T}}$, $t_{13}=-\mathbf{D}_Q^{\mathrm{T}}\mathbf{D}_w^{\mathrm{T}}$,

$t_{11}=\mathbf{S}_{11}^{\mathrm{T}}\overline{\mathbf{C}}_1^{\mathrm{T}}+\overline{\mathbf{C}}_2^{\mathrm{T}}-\mathbf{C}_e^{T}\mathbf{D}_Q^{T}\mathbf{D}_w^{\mathrm{T}}$, $t_{12}=\mathbf{S}_{12}^{\mathrm{T}}\overline{\mathbf{C}}_1^{\mathrm{T}}-\mathbf{C}_Q^{\mathrm{T}}\mathbf{D}_w^{\mathrm{T}}$.

The whole problem results in the minimization of γ subject to the *LMI* constraint (31):

$$\min_{LMI}\gamma \qquad (32)$$

5 APPLICATION TO A STIRRED TANK REACTOR

The previous robustification methodology is applied now to the simplified MIMO model of a stirred tank reactor presented in the transfer function formalism in (Camacho and Bordons, 2004):

$$\begin{bmatrix}Y_1(s)\\ Y_2(s)\end{bmatrix}=\begin{bmatrix}1/(1+0.7s) & 5/(1+0.3s)\\ 1/(1+0.5s) & 2/(1+0.4s)\end{bmatrix}\begin{bmatrix}U_1(s)\\ U_2(s)\end{bmatrix} \qquad (33)$$

where Y_1 and Y_2 are the effluent concentration and the reactor temperature, U_1 and U_2 are the feed flow rate and the coolant flow, respectively.

Starting from the state-space representation of this 2 inputs/ 2 outputs model discretized for a sampling time $T_e=0.03$ min, an integral action is added leading to an extended state-space model. For simplicity reasons of multivariable MPC, the same prediction horizons $N_1=1$, $N_2=3$ and $N_u=2$ were used for all outputs and control signals, and the same weights as in (Camacho and Bordons, 2004) $\widetilde{R}_J=0.05\mathbf{I}_{N_u}$ and $\widetilde{Q}_J=\mathbf{I}_{N_2-N_1+1}$.

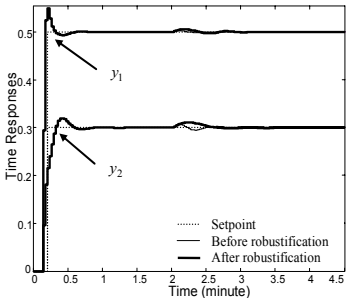


Figure 5: y_1 and y_2 before and after robustification.

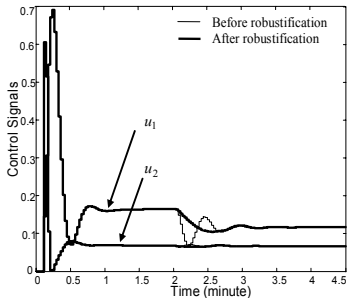


Figure 6: u_1 and u_2 before and after robustification.

Figures 5 shows the time responses obtained for a step reference of 0.5 for y_1, and 0.3 for y_2, and the disturbance rejection for a step disturbance of 0.05 applied to u_1 at $t = 2$ min. Figure 6 shows the control signals u_1 and u_2.

For robustness under additive uncertainties at high frequency, a high-pass filter is used for each control signal, as described in Section 4.2 which transfer form is $\mathbf{W_u} = \mathbf{I}_2 (1 - 0.7q^{-1}) / 0.3$. Using the optimization procedure based on LMIs gives a multivariable Youla parameter as a 2×2 matrix of polynomials of order $n_Q = 20$.

Figure 7 shows the singular values analysis of transfer from **b** to control signals **u** (from Figure 4). The greatest value of maximal singular values represents the H_∞ norm. We can remark that this H_∞ norm has been reduced. In this way the stability robustness is improved with respect to high-frequency additive unstructured uncertainties.

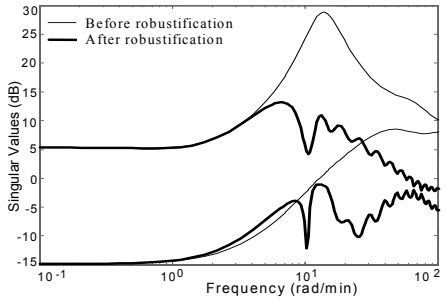


Figure 7: Singular values before and after robustification.

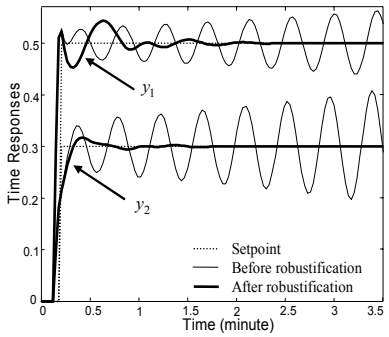


Figure 8: y_1 and y_2 before and after robustification.

Figures 5 and 6 show that after robustification the input/output behaviour is unchanged, but the disturbance is rejected more slowly by the robustified controller. In fact, the robustified controller has a slower disturbance rejection, but a higher robust stability. To support this, a high frequency neglected dynamics of the actuator u_1 has been considered. Thus the transfer between y_1 / u_1 corresponds to $1/(1+0.7s)(1+0.07s)$. Figure 8 illustrates that the initial controller behaviour is destabilized by this uncertainty, but the robustified controller remains stable; it also shows the influence of the considered unstructured uncertainty to y_2.

6 CONCLUSIONS

This paper has presented a new MIMO complete methodology which enables robustifing an initial multivariable MPC controller in state-space formalism using the Youla parameter framework. In order to improve robustness towards unstructured uncertainties, a H_∞ convex optimization problem was solved using the LMIs techniques. The major advantage of the developed structure is the state-space formulation of this MPC robustification problem for MIMO systems with a reduced computational effort compared to the transfer function formalism. This method can also be applied to non square systems, which otherwise are more difficult to control. This technique enables also the use of time-domain templates to manage the compromise between stability robustness and nominal performance.

REFERENCES

Boyd, S., Barratt, C., 1991. *Linear controller design. Limits of performance*, Prentice Hall.

Boyd, S., Ghaoui, L.El., Feron, E., Balakrishnan, V., 1994. *Linear matrix inequalities in system and control theory*, SIAM Publications, Philadelphia.

Camacho, E.F., Bordons, C., 2004. *Model predictive control,* Springer-Verlag. London, 2nd edition.

Clement, B., Duc, G., 2000. A multiobjective control via Youla parameterization and LMI optimization: application to a flexible arm, IFAC Symposium on Robust Control and Design, Prague.

Dumur, D., Boucher, P., 1998. *A Review introduction to linear GPC and applications*, Journal A, 39(4), pp. 21-35.

Kouvaritakis, B., Rossiter, J.A., Chang, A.O.T., 1992. *Stable generalized predictive control: an algorithm with guaranteed stability*, IEE Proceedings-D, 139(4), pp. 349-362.

Maciejowski, J.M., 1989. *Multivariable feedback design*, Addison-Wesley Publishing Company, Wokingham, England.

Maciejowski, J.M., 2001. *Predictive control with constrains*, Prentice Hall.

Magni, J.F., 2002. *Robust modal control with a toolbox for use with MATLAB*, Springer.

Yoon, T.W., Clarke, D.W. 1995. *Observer design in receding-horizon predictive control*, International Journal of Control, 61(1), pp. 171-191.

A MULTI-MODEL APPROACH FOR BILINEAR GENERALIZED PREDICTIVE CONTROL

Anderson Luiz de Oliveira Cavalcanti
Informatic and Industry Academic Department, CEFET, Natal/RN, Brazil
anderson@cefetrn.br

André Laurindo Maitelli
Department of Computation and Automation, UFRN, Campus Universitário S/N, Natal/RN, Brazil
maitelli@dca.ufrn.br

Adhemar de Barros Fontes
Department of Electrical Engineering,UFBA, Rua Aristides Novis, 04, Salvador/BA, Brazil
adhemar@ufba.br

Keywords: Model Predictive Control, Multi-Model, Distillation Column.

Abstract: This paper presents a contribution in multivariable predictive control. A new approach of multi-model based control is presented. The controller used is the quasilinear multivariable generalized predictive control (QMGPC). A metric based in 2-norm is presented in order to build a global model using local models. Simulation results in a distillation column, with a comparative analysis, are presented.

1 INTRODUCTION

The multi-model approach has been presented as an alternative method to be applied is systems that operate in a long range (Aslan *et al.*, 2004). When a process operates in a long range, due to non-linearities, usually the parametric variation of its models is large. For this reason, usually, a controller based in just one model has poor performance in these kind of process.

The basic idea of multi-model approach is to identify a set of models (one for each operating regime in a chosen trajectory) and to interpolate these models (through an interpolation function). Other approach calculates a suitable control effort as a wheighting sum of each control effort (in each designed controller for each operating regime).

Some approches use space state models like (Azimadeh *et al.*, 1998) and (Foss *et al.*, 1995). In (Azimadeh *et al.*, 1998) a set of linear space state models is chosen in a given trajectory. In (Foss *et al.*, 1995) a set on nonlinear space state models is chosen (and a nonlinear predictive controller is designed).

A closed loop metric, that guarantee the global stability, is proposed in (Aslan *et al.*, 2004). In that case, a set of PI controllers is projected and, for each instant, the distance from the current point in a chosen trajectory to a tabled operating regime is calculated.

In this paper, a similar idea to (Foss *et al.*, 1995) is proposed. In this case, a set of local bilinear models is identified. The global model is build with a wheigthing sum of the identified local models. The wheigthing factor is calculated based in a proposed metric. This metric consists of use a 2-norm to measure the distance from the current point (in a chosen monotonic trajectory) and a tabled operating regime. A case study in a debutanized distillation column is presented in order to show an application of the proposed controller.

The next step of this research is the stability and robustness analisys (to presents a stable algorithm proposal).

2 MULTIVARIABLE MULTI-MODEL

The designed controller is based in quasilinear multivariable generalized predictive control (QMGPC). This controller is based in multivariable bilinear NARIMAX (Non Linear, Auto-Regressive, Moving Average, with exogenous input) models.

The basic idea of QMGPC algorithm is calculate a control effort sequence, based in the minimization of a multi-step objective function, in a defined prediction horizon.

2.1 Multivariable Multi-Model

The multivariable multi-model bilinear NARIMAX model with p-inputs and q-outputs is given by:

$$\begin{aligned} &A^{(k)}(q^{-1})\Delta_q(q^{-1})y(k) = B^{(k)}(q^{-1})\Delta_p(q^{-1})u(k-1) + \\ &D_e^{(k)}(q^{-1})D[u(k-1)]D_d^{(k)}(q^{-1})\Delta_q(q^{-1})y(k-1) + \\ &C^{(k)}(q^{-1})e(k) \end{aligned} \quad (1)$$

where $y(k) \in R^q$ is the process output vector, $u(k) \in R^p$ is the process input vector and $e(k) \in R^q$ is the gaussian white noise with zero mean and covariance $diag(\sigma^2)$. The matrices $A^{(k)}(q^{-1})$, $B^{(k)}(q^{-1})$ and $C^{(k)}(q^{-1})$ are polynomials matrices in shift operator q^{-1} and are defined by:

$$A^{(k)}(q^{-1}) = I_{q\times q} + A_1^{(k)}q^{-1} + \cdots + A_{na}^{(k)}q^{-na} \quad (2)$$

$$B^{(k)}(q^{-1}) = B_0^{(k)} + B_1^{(k)}q^{-1} + \cdots + B_{nb}^{(k)}q^{-nb} \quad (3)$$

$$C^{(k)}(q^{-1}) = I_{p\times p} + C_1^{(k)}q^{-1} + \cdots + C_{nc}^{(k)}q^{-nc} \quad (4)$$

$$D_d^{(k)}(q^{-1}) = D_{d,0}^{(k)} + D_{d,1}^{(k)}q^{-1} + \cdots + D_{d,nd_d}^{(k)}q^{-nd_d} \quad (5)$$

$$D_e^{(k)}(q^{-1}) = D_{e,0}^{(k)} + D_{e,1}^{(k)}q^{-1} + \cdots + D_{e,nd_e}^{(k)}q^{-nd_e} \quad (6)$$

where:
$A^{(k)}(q^{-1}) \in R^{q\times q}$, $B^{(k)}(q^{-1}) \in R^{q\times p}$, $C^{(k)}(q^{-1}) \in R^{q\times q}$, $D_e^{(k)}(q^{-1}) \in R^{q\times p}$ and $D_d^{(k)}(q^{-1}) \in R^{p\times q}$. The matrix $D[u(k-1)]$ is defined as:

$$D[u(k-1)] = \begin{bmatrix} u_1(k-1) & 0 & \cdots & 0 \\ 0 & u_2(k-1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_p(k-1) \end{bmatrix} \quad (7)$$

The generic polynomial matrix $P^{(k)}(q^{-1})$ in (1) represents this matrix in instant *k*.

The first step to build (1) is decompose the system's operating range into a number of operating regimes that completely cover the chosen trajectory. Second, for each operating regime, a local model structure must be developed as showed in (Foss *et al.*, 1995). In this case, the model structure is chosen by using the Akaike criterion.

The last step is to identify the parameter's model for each local model. The estimation algorithm is the Multivariable Recursive Least Squares (MRLS).

2.2 Building the Global Model

The global model is built as a weighting sum of the bilinear models in each chosen operating regime. The generic polynomial matrix $P^{(k)}(q^{-1})$ is built as:

$$P^{(k)}(q^{-1}) = \sum_{i=1}^{NOR} P_{(i)}(q^{-1})w_{i,k} \quad (8)$$

where $P_{(i)}(q^{-1})$ is the polynomial matrix in i^{th} operating regime, $w_{i,k}$ is the i^{th} weighting factor calculated in instant *k*, *NOR* is the number of operating regimes. The computation of $w_{i,k}$ is showed in the section 3 of this paper.

2.3 The Quasilinear Multivariable Multi-Model

The nonlinear model presented in (1) is quasilinearized to be used in QMGPC (Quasilinear Multivariable Generalized Predictive Control). The multivariable quasilinear multi-model must be obtained by rewriting the expression (1) of the following form:

$$\begin{aligned} &A^{(k)}(q^{-1},u)\Delta_q(q^{-1})y(k) = \\ &B(q^{-1})^{(k)}\Delta_p(q^{-1})u(k-1) + C^{(k)}(q^{-1})e(k) \end{aligned} \quad (9)$$

where:

$$\begin{aligned} &A^{(k)}(q^{-1},u) = A^{(k)}(q^{-1}) - \\ &D_e^{(k)}(q^{-1})D[u(k-1)]D_d^{(k)}(q^{-1}) \end{aligned} \quad (10)$$

The polynomial matrix $A^{(k)}(q^{-1},u)$ is calculated considering its parameters as constant in prediction

horizon. The polynomial matrix $A^{(k)}(q^{-1})$ is considered diagonal in this paper.

2.4 The Predictor

The output prediction *i-step* ahead may be calculated multiplying the expression (1) for q^i as in the following expression:

$$\begin{aligned}&\widetilde{A}^{(k)}(q^{-1},u)y(k+i)=\\&B^{(k)}(q^{-1})\Delta_p(q^{-1})u(k+i-1)+C^{(k)}(q^{-1})e(k+i)\end{aligned} \tag{11}$$

where $\widetilde{A}^{(k)}(q^{-1},u)=A^{(k)}(q^{-1},u)\Delta_q(q^{-1})$.

In this case, the polynomial matrix $C(q^{-1})=I_{p\times p}$ is uncorrelated (white noise). Considering the following Diophantine equation:

$$I_{p\times p}=E_i^{(k)}(q^{-1},u)\widetilde{A}^{(k)}(q^{-1},u)+q^{-i}F_i^{(k)}(q^{-1},u) \tag{12}$$

where:

$$E_i^{(k)}(q^{-1},u)=E_{i,o}{}^{(k)}(u)+\cdots+E_{i,i-1}{}^{(k)}(u)q^{-(i-1)} \tag{13}$$

$$F_i^{(k)}(q^{-1},u)=F_{i,o}{}^{(k)}(u)+\cdots+F_{i,na}{}^{(k)}(u)q^{-na} \tag{14}$$

Pre-multiplying (11), with $C(q^{-1})=I_{p\times p}$, for $E_i(q^{-1},u)$ we obtain:

$$\begin{aligned}&E_i^{(k)}(q^{-1},u)\widetilde{A}^{(k)}(q^{-1},u)y(k+i)=\\&E_i^{(k)}(q^{-1},u)B^{(k)}(q^{-1})\Delta_p(q^{-1})u(k+i-1)+\\&E_i^{(k)}(q^{-1},u)e(k+i)\end{aligned} \tag{15}$$

Rewriting (12) of the following form:

$$E_i(q^{-1},u)\widetilde{A}(q^{-1},u)=I_{p\times p}-q^{-i}F_i(q^{-1},u) \tag{16}$$

Substituting (16) in (15) we obtain:

$$\begin{aligned}&y(k+i)=F_i^{(k)}(q^{-1},u)y(k)+E_i^{(k)}(q^{-1},u)e(k+i)\\&E_i^{(k)}(q^{-1},u)B^{(k)}(q^{-1})\Delta_p(q^{-1})u(k+i-1)+\end{aligned} \tag{17}$$

As the degree of $E_i^{(k)}(q^{-1},u)$ is $i-1$, then the sub-optimal prediction of $y(k+i)$ is:

$$\begin{aligned}&\hat{y}(k+i)=F_i^{(k)}(q^{-1},u)y(k)+\\&E_i^{(k)}(q^{-1},u)B^{(k)}(q^{-1})\Delta_p(q^{-1})u(k+i-1)\end{aligned} \tag{18}$$

Make:

$$E_i^{(k)}(q^{-1},u)B^{(k)}(q^{-1})=H_i^{(k)}(q^{-1},u)+q^{-i}H_{ipa}{}^{(k)}(q^{-1},u) \tag{19}$$

As the degree of $H_i(q^{-1},u)$ is less than $i-1$, the predictor may be written as:

$$\begin{aligned}&\hat{y}(k+i)=F_i^{(k)}(q^{-1},u)y(k)+H_{ipa}{}^{(k)}(q^{-1},u)\Delta_p(q^{-1})u(k-1)+\\&H_i^{(k)}(q^{-1},u)\Delta_p(q^{-1})u(k+i-1)\end{aligned} \tag{20}$$

The last term of (20) considers the future inputs (forced response) and the two first terms consider only past inputs (free response). Define:

$$\hat{y}(k+i)=H_i{}^{(k)}(q^{-1},u)\Delta_p(q^{-1})u(k+i-1)+Y_{li}{}^{(k)} \tag{21}$$

where:

$$Y_{li}{}^{(k)}=F_i^{(k)}(q^{-1},u)y(k)+H_{ipa}{}^{(k)}(q^{-1},u)\Delta_p(q^{-1})u(k-1) \tag{22}$$

2.5 The Objetive Function

The objective function is given by:

$$J=\sum_{i=N_1}^{NY}\left\|r(k+i)-\hat{y}(k+i)\right\|^2_{R_{(k)}}+\sum_{i=1}^{NU}\left\|\Delta u(k+i-1)\right\|^2_{Q_{(k)}} \tag{23}$$

where N_1 is minimum prediction horizon, NY is prediction horizon, NU is the control horizon, $R_{(k)}$ and $Q_{(k)}$ are weighting matrices of error signal and control effort in instant k in the chosen trajectory, respectively, $\hat{y}(k+i)$ is the sub-optimum i-step ahead predicted output, $r(k+i)$ is the future reference trajectory.

2.6 The Control Law

The control effort is obtained, without constraints, by the minimization of the objective function. This minimization is obtained by the calculation of its gradient (making it equals zero), of the following form:

$$\frac{\partial J}{\partial u}=0 \tag{24}$$

Consider the predictions set:

$$y_{N_{1y}}=H^{(k)}{}_{N_{1yu}}u_{NU}+Y_{lN_1y}{}^{(k)} \tag{25}$$

where:

$$y_{N_{1y}} = \left[\hat{y}(k+N_1) \quad \hat{y}(k+N_1+1) \quad \cdots \quad \hat{y}(k+NY)\right]^T \tag{26}$$

$$H_{N_{1yu}}{}^{(k)} = \begin{bmatrix} H^{(k)}{}_{N_1-1} & H^{(k)}{}_{N_1-2} & \cdots & H^{(k)}{}_{N_1-NU} \\ H^{(k)}{}_{N_1} & H^{(k)}{}_{N_1-1} & \cdots & H^{(k)}{}_{N_1+1-NU} \\ \vdots & \vdots & \ddots & \vdots \\ H^{(k)}{}_{NY-1} & H^{(k)}{}_{NY-2} & \cdots & H^{(k)}{}_{NY-NU} \end{bmatrix} \tag{27}$$

$$u_{NU} = \begin{bmatrix} \Delta_p(q^{-1})u(k) \\ \Delta_p(q^{-1})u(k+1) \\ \vdots \\ \Delta_p(q^{-1})u(k+NU-1) \end{bmatrix} \tag{28}$$

$$y_{lN_{1y}} = \begin{bmatrix} Y_{lN_1} \\ Y_{lN_1+1} \\ \vdots \\ Y_{lNY} \end{bmatrix} \tag{29}$$

The objective function (23) may be rewritten of the following form:

$$J=(H_{N_{1yu}}u_{NU}+y_{lN_{1y}})^T\overline{R}_{(k)}(H_{N_{1yu}}u_{NU}+y_{lN_{1y}})+u_{NU}^T\overline{Q}_{(k)}u_{NU} \tag{30}$$

where:

$$\overline{R}^{(k)} = diag[R_1{}^{(k)},\cdots,R_{q\times NY}{}^{(k)}] \tag{31}$$

$$\overline{Q}^{(k)} = diag[Q_1{}^{(k)},\cdots,Q^{(k)}{}_{p\times NU}] \tag{32}$$

The computation of an element $x_i^{(k)}$ of $\overline{R}^{(k)}$ and $\overline{Q}^{(k)}$ is given by:

$$x_i^{(k)} = \sum_{j=1}^{NOR} x_{i,j}w_{j,k} \tag{33}$$

where $x_{i,j}$ is the i^{th} element of weigthing matrix ($\overline{R}^{(k)}$ or $\overline{Q}^{(k)}$) for the j^{th} operating regime and $w_{i,k}$ is the i^{th} weighting factor calculated in instant k.

The minimization of (30) produces the following control law:

$$u = (H_{N_{1yu}}^T H_{N_{1yu}} + \overline{Q})^{-1} H_{N_{1yu}}^T \overline{R}(r - y_{lN_{1y}}) \tag{34}$$

Because of the receding control horizon, only the first p rows of (34) are computed.

3 THE PROPOSED METRIC

The proposed metric is based in a 2-norm. Norms, in general, gives a notion of distance in a vectorial space. In multivariable case, in a process with p-inputs and q-outputs, the output is $y(k) \in R^q$ and the input is $u(k) \in R^p$. In a known trajectory of process output, the distance of the process's output from the first operating regime to the last operating regime is given by:

$$d_{1,NOR} = \left\| y_{NOR} - y_1 \right\|_2 \tag{35}$$

where y_{NOR} is the process's output in last operating regime and y_1 is the process's output in the first operating regime.

To measure the distance from the current process's output (in instant k) to the i^{th} operating regime, we can use the expression:

$$\delta_{i,k} = \frac{d_{1,NOR}}{\left\| y(k) - y_i \right\|_2}; \quad i = 1,\cdots,NOR \tag{36}$$

where $y(k)$ is the process's output in instant k and y_i is the process's output to the i^{th} operating regime.

The weighting factor for the i^{th} operating regime in instant k is given by:

$$w_{i,k} = \frac{\delta_{i,k}}{\sum_{j=1}^{NOR}\delta_{j,k}}; \quad i = 1,\cdots,NOR \tag{37}$$

4 APPLICATION OF THE MULTI-MODEL APPROACH

4.1 Distillation Column

In this paper, an application in a debutanizer distillation column is showed. Debutaziner distillation column is usually used to remove the light components from the gasoline stream to produce Liquefied Petroleum Gas (LPG). The column is showed in Figure 1.

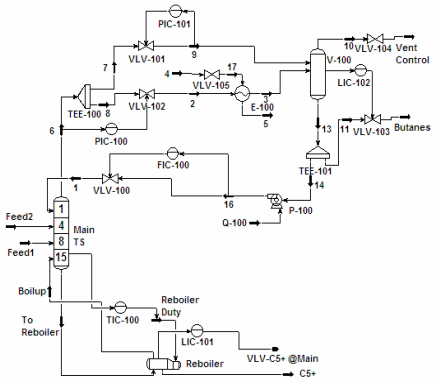


Figure 1: Distillation Column simulated in Hysys Software.

The most common control strategy is to manipulate the reflux flow rate and the temperature in column's bottom and, to control the concentrations of any product in *butanes* stream and in C5+ stream as showed in (Almeida, *et al.*, 2000) and (Fontes, et al., 2006). The chosen process variables are: concentration of i-pentane in butanes stream (y_1) and concentration of i-butene in C5+ stream (y_2).

The reflux flow rate (u_1) is manipulated through the FIC-100 controller and the temperature of column's bottom (u_2) is manipulated through the TIC-100 controller. The reflux flow rate is given in m^3/h and the temperature of column's bottom is given in ^oC.

In this case study, three operation regime were chosen, as showed in Table 1. The identified bilinear models were obtained using the multivariable recursive least squares algorithm and the model's structure has been chosen by using the Akaike criterion. In all models, the chosen sample rate is 4 minutes.

The trajectory of y_1 is monotonically increasing and the trajectory of y_2 is monotonically decreasing.

Table 1: Chosen Operating Regimes.

Operation Regime	Input	Output (Mass Fractions)
1	$u_1 = 40\ m^3/h$	$y_1 = 0.014413$
	$u_2 = 147\ ^oC$	$y_2 = 0.001339$
2	$u_1 = 37\ m^3/h$	$y_1 = 0.017581$
	$u_2 = 147.5\ ^oC$	$y_2 = 0.001161$
3	$u_1 = 34\ m^3/h$	$y_1 = 0.021994$
	$u_2 = 148\ ^oC$	$y_2 = 0.001004$

The operating regimes must be chosen based in a knowledge of the process.

4.2 Results

In this simulation, the process is in the 3rd operating regime and a deviation in reference is applied in the proposed controller. With this reference deviation, the process will come to close to the 1st operating regime. The proposed quasilinear multi-model is compared with quasilinear single-model (using the model of the 3rd operating regime). Figures 2 and 3 show the output comparison.

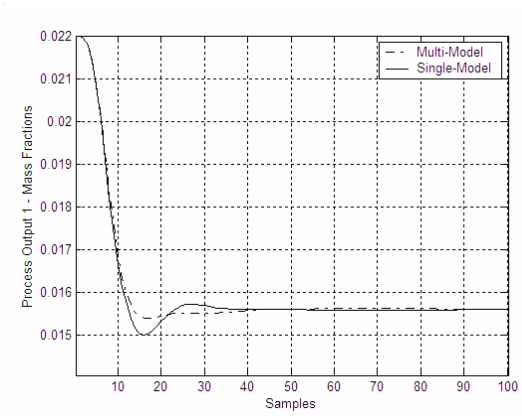


Figure 2: Process Output 1. Comparison between single-model and multi-model approach.

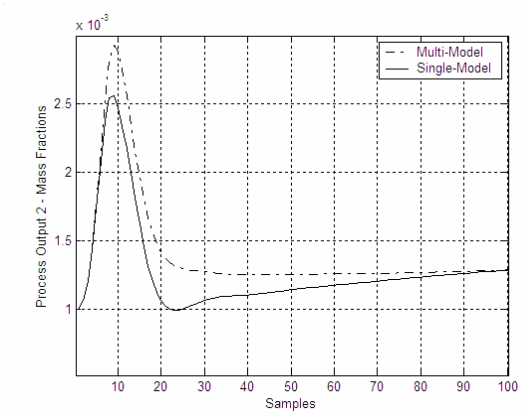


Figure 3: Process Output 2. Comparison between single-model and multi-model approach.

Figures 4 and 5 show the control effort comparison between the quasilinear single-model and multi-model approaches.

The figures show the better performance of multi-model approach when compared with single-model approach.

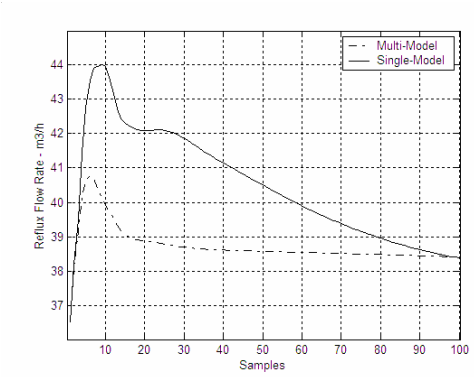


Figure 4: Reflux Flow rate. Comparison between single-model and multi-model approach.

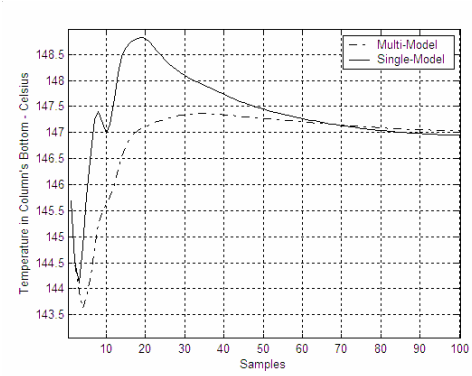


Figure 5: Temperature in column's bottom. Comparison between single-model and multi-model approach.

In order to quantitatively asses the performance of multi-model quasilinear GPC, some indices like showed in (Goodhart, *et al.*, 1994) are calculated. Theses indices may be extended to multivariable case, of the following form:

$$\varepsilon_{1,i} = \sum \left| u_i(k) \right| / N \tag{38}$$

where $i = 1, \cdots, p$ and N is the amount of control effort applied in the process to achieve the desired response. The index showed in (38) is the account of total control effort to achieve a given response. The variance of controlled actuators is:

$$\varepsilon_{2,i} = \sum (u_i(k) - \varepsilon_{1,i})^2 / N \tag{39}$$

The deviation of the process of integral of absolute error (IAE) is:

$$\varepsilon_{3,j} = \sum \left| r_j(k) - y_j \right| / N \tag{40}$$

where $j = 1, \cdots, q$.

The overall measure of effectiveness is defined as:

$$\varepsilon_j = \sum_{i=1}^{p} (\alpha_i \varepsilon_{1,i} + \beta_i \varepsilon_{2,i}) + \rho_j \varepsilon_{3,j} \tag{41}$$

where $j = 1, \cdots, q$. The factors α_i , β_i and ρ_j are weightings chosen to reflect the actual financial cost of energy usage, actuator wear and product quality, respectively. In this case, we consider $\alpha_i = 0.1$, $\beta_i = 0.15$ and $\rho_j = 0.5$.

Table 2: Comparison of Performance indices between Quasilinear single-model and Quasilinear multi-model with *N*=100.

I/O	Model	ε_1	ε_2	ε_3	ε
1	Single	40.47	2.61	499.46	269.00
	Multi	38.72	0.31	486.20	261.80
2	Single	147.38	0.63	242.40	140.47
	Multi	146.88	0.62	197.71	117.56

Table 2 shows the performance of quasilinear multi-model approach in terms of less energy usage, less actuator wear and better product quality in relation to quasilinear single-model performance.

5 CONCLUSIONS

The multi-model approach is a good alternantive of controller to systems that operate in a large operation range. The indices has shown that this approach presents better performance in relation of quasilinear single model.

REFERENCES

Almeida, E., Rodrigues, M.A., Odloak, D., 2000. Robust Predictive Control of a Gasoline Debutanizer Column. *Brazilian Journal of Chemical Engineering*, vol. 17, pp. 11, São Paulo.

Arslan, E., Çamurdan, M. C., Palazoglu, A. and Arkun, Y., 2004. Multi-Model Control of Nonlinear Systems Using Closed-Loop Gap Metric. *Proceedings of the 2004 American Control Conference,* Vol. 3, pp. 2374-2378.

Azimadeh, F., Palizban, H.A. and Romagnoli, J. A., 1998. On Line Optimal Control of a Batch Fermentation Process Using Multiple Model Approach. *Proceedings of the 37th IEEE Conference on Decision & Control*, pp. 455-460.

Fontes, A., Maitelli, A.L., Cavalcanti, A. L. O. and Angelo, E., 2006. Application of Multivariable Predictive Control in a Debutanizer Distillation Column. *Proceedings of SICOP 2006 – Workshop on Solving Industrial Control and Optimization Problems,* pp. 1-5.

Foss, B.A., Johansen, T.A. and Sorensen, A.V., 1995. Nonlinear Predictive Control Using Local Models – Applied to a Batch Fermentation Process. *Control Eng. Practice,* pp. 389-396.

Goodhart, S. G., Burnham, K. J., James, D.J.G., 1994. Bilinear Self-tuning Control of a high temperature Heat Treatment Plant. *IEEE Control Theory Appl.:* Vol. 141, no 1, pp. 779-783.

APPLICATIONS OF A MODEL BASED PREDICTIVE CONTROL TO HEAT-EXCHANGERS

Radu Bălan, Vistrian Mătieş, Victor Hodor
Dept.of Mechatronics, Technical University of Cluj-Napoca, C. Daicoviciu no. 15, Cluj-Napoca, Romania
radubalan@yahoo.com, matiesvistrian@yahoo.com, victor.hodor@termo.utcluj.ro

Sergiu Stan, Ciprian Lăpuşan, Horia Bălan
Dept. of Mechanics and Programming, Dept.of Mechatronics, Dept. of Energetics, Technical University of Cluj-Napoca
sergiustan@ieee.org, lapusanciprian@yahoo.com, horia.balan@eps.utcluj.ro

Keywords: Heat-exchanger, nonlinear control, on-line simulation, rule-based control.

Abstract: Model based predictive control (MBPC) is an optimization-based approach that has been successfully applied to a wide variety of control problems. When MBPC is employed on nonlinear processes, the application of this typical linear controller is limited to relatively small operating regions. The accuracy of the model has significant effect on the performance of the closed loop system. Hence, the capabilities of MBPC will degrade as the operating level moves away from its original design level of operation. This paper presents an MBPC algorithm which uses on-line simulation and rule-based control. The basic idea is the on-line simulation of the future behaviour of control system, by using a few control sequences and based on nonlinear analytical model equations. Finally, the simulations are used to obtain the 'optimal' control signal. These issues will be discussed and nonlinear modelling and control of a single-pass, concentric-tube, counter flow or parallel flow heat exchanger will be presented as an example.

1 INTRODUCTION

Model Based Predictive Control (MBPC) refers to a class of algorithms that utilize an explicit process model to compute the control signal by minimizing an objective function (Comacho, 1999). The performance objective typically penalizes predicted future errors and manipulated variable movement subject to various constraints. The ideas appearing in greater or lesser degree in all the predictive control family are basically:

-explicit use of a model to predict the process output in the future;

-on line optimization of a cost objective function over a future horizon;

-receding strategy, so that at each instant, the horizon is displaced towards the future, which involves the application of the first control signal of the sequence calculated at each step.

Performance of MBPC could become unacceptable due to a very inaccurate model, thus requiring a more accurate model. This task is an instance of closed-loop identification and adaptive control. Here it is important to remember that the model is only used as an instrument in creating the best combined performance of the controller and the actual system, so the model does not necessarily need to be a good open-loop model of the system. The performance measure should be able to capture as much of the closed loop behavior as possible.

Let's consider that it is possible to compute:
- the predictions of output over a finite horizon (*N*);
- the cost of an objective function,

for each possible sequence:

$$u(.) = \{u(t), u(t+1), .., u(t+N)\} \qquad (1)$$

and then to choose the first element of the optimal control sequence. For a first look, the advantages of the proposed algorithm (Balan, 2001) include the following:

-the minimum of objective function is global;

-it is not necessary to invert a matrix, so potential difficulties are avoided;

-it can be applied to nonlinear processes if a nonlinear model is available;

-the constraints (linear or nonlinear) can easily be implemented.

The drawback of this scheme is a very long computational time, because there are possibly a lot of sequences. For example, if u(t) is applied to the

process using a "p" bits numerical-analog converter (DAC), the number of sequences is 2^{p*N}. Therefore, the number of sequences must be reduced.

In the next sections, these issues will be discussed and nonlinear modelling and control of a single-pass, concentric-tube, counter flow heat exchanger will be presented as an example.

2 THE MODEL OF THE HEAT-EXCHANGER

Heat exchangers are devices that facilitate heat transfer between two or more fluids at different temperatures. Usually, MBPC uses a linear model and an on-line least square algorithm (RLS) to determine the parameters. Heat exchangers are nonlinear processes. To apply the standard MBPC algorithms it is possible to use multiple model adaptive control approach (MMAC) which uses a bank of models to capture the possible input-output behavior of processes (Dougherty, 2003). Other solutions are based on neural networks and fuzzy logic (Fischer, 1998), (Fink, 2001).

In this paper it is used an example from (Ozisik, 1985): a heat exchanger with hot fluid -engine oil at 80ºC, cold fluid - water at 20º C, by using a single-pass counter flow (or parallel flow for some experiments) concentric-tube. Other data and notations: length (L): 60m, heat transfer coefficients (k_1=1000 W/(m^2 ºC), k_2=80 W/(m^2 ºC)), the temperature profile of fluids and wall ($\theta_1(z,t)$, $\theta_2(z,t)$, $\theta_w(z,t)$), specific heat (c_1, c_2, c_w), cross-sectional area for fluids flow and wall (S_1, S_2, S_w), density of fluids and wall (ρ_1, ρ_2, ρ_w), flow speed of fluids (v_1, v_2), transfer area (S) (fig. 1).

If physical properties (density, heat capacity, heat transfer coefficients, flow speed) are assumed constant, the heat exchanger model is described using a shell energy balance as (Douglas, 1972):

-hot fluid:

$$c_1\rho_1 S_1\frac{\partial\theta_1(z,t)}{\partial t}-c_1\rho_1 v_1 S_1\frac{\partial\theta_1(z,t)}{\partial z}=\frac{k_1 S}{L}[\theta_w(z,t)-\theta_1(z,t)] \quad (2)$$

-cold fluid:

$$c_2\rho_2 S_2\frac{\partial\theta_2(z,t)}{\partial t}+c_2\rho_2 v_2 S_2\frac{\partial\theta_1(z,t)}{\partial z}=\frac{k_2 S}{L}[\theta_w(z,t)-\theta_2(z,t)] \quad (3)$$

-wall:

$$c_w\rho_w S_w\frac{\partial\theta_w(z,t)}{\partial t}=\frac{S}{L}[k_1\theta_1(z,t)+k_2\theta_2(z,t)-(k_1+k_2)\theta_w(z,t)] \quad (4)$$

Using general notation $\theta_{a(i,j)}$ with a=1 (hot fluid), a=2 (cold fluid), a=w (wall), i, j discrete elements in space respectively time, the discrete equations corresponding to partial differential equations (2),(3),(4) are:

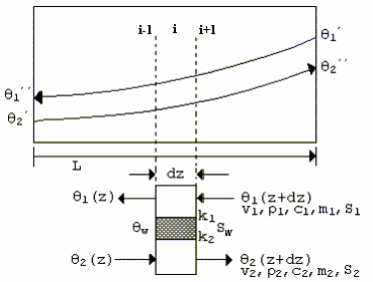


Figure 1: Temperature distributions.

$$\theta_1(i,j+1)=\theta_1(i,j)\left[1-v_1\frac{\Delta t}{\Delta z}-\frac{k_1 S\Delta t}{Lc_1\rho_1 S_1}\right]+ +v_1\frac{\Delta t}{\Delta z}\theta_1(i+1,j)+\frac{k_1 S\Delta t}{Lc_1\rho_1 S_1}\theta_w(i,j) \quad (5)$$

$$\theta_2(i,j+1)=\theta_2(i,j)\left[1+v_2\frac{\Delta t}{\Delta z}-\frac{k_2 S\Delta t}{Lc_2\rho_2 S_2}\right]- -v_2\frac{\Delta t}{\Delta z}\theta_2(i+1,j)+\frac{k_2 S\Delta t}{Lc_2\rho_2 S_2}\theta_w(i,j) \quad (6)$$

$$\theta_w(i,j+1)=\theta_w(i,j)+ +\frac{S\Delta t}{L}[k_1\theta_1(i,j)+k_2\theta_2(i,j)+(k_1+k_2)\theta_w(i,j)] \quad (7)$$

In a control application, these equations can not be used directly because v_1 and v_2 are not constant in time. Let's consider next assumptions:

-the speed of fluids is limited:

$$v_{1(\min)}<v_1<v_{1(\max)};$$
$$v_{2(\min)}<v_2<v_{2(\max)};v_{\max}=\max(v_{1(\max)},v_{2(\max)}) \quad (8)$$

- the fluids speed is only time-function:

$$v_1=v_1(t)\ ,\ dv_1/dz=0\ ,\ v_2=v_2(t)\ ,\ dv_2/dz=0 \quad (9)$$

- the length of heat exchanger is divided in n intervals: $L=n\Delta z$; (10)

- in an interval Δt, the fluids cover only a part of Δz: $n_v v_{\max}\Delta t=\Delta z\ ;\ \Delta t < L/(nn_v v_{\max})$ (11)

- two variables Δz_1, Δz_2 are using to totalize the small fluid displacements:

$$\Delta z_1(t+\Delta t)=\Delta z_1(t)+v_1\Delta t\ ;$$
$$\Delta z_2(t+\Delta t)=\Delta z_2(t)+v_2\Delta t \quad (12)$$

- in simulations, the displacements of the fluids become effective only if $\Delta z_1>\Delta z$ or/and $\Delta z_2>\Delta z$; in these cases:

$$\Delta z_1\leftarrow \Delta z_1-\Delta z \text{ or/and } \Delta z_2\leftarrow \Delta z_2-\Delta z \quad (13)$$

In other words, in simulations, the continue moves of fluids are replaced with small discrete displacements. As a result, the heat exchanger model is described by equations:

$$\theta_1(i,j+1)=\theta_1(i,j)\left[1-\frac{k_1 S\Delta t}{Lc_1\rho_1 S_1}\right]+\frac{k_1 S\Delta t}{Lc_1\rho_1 S_1}\theta_w(i,j) \quad (14)$$

$$\theta_2(i,j+1)=\theta_2(i,j)\left[1-\frac{k_2 S\Delta t}{Lc_2\rho_2 S_2}\right]+\frac{k_2 S\Delta t}{Lc_2\rho_2 S_2}\theta_w(i,j) \quad (15)$$

$$\theta_w(i,j+1)=\theta_w(i,j)+ +\frac{S\Delta t}{L}\left[k_1\theta_1(i,j)+k_2\theta_2(i,j)+(k_1+k_2)\theta_w(i,j)\right] \quad (16)$$

In a practical implementation, there are used equations (12), (13), (14), (15), (16).

It is important the number and position of temperature sensors. Here, it is considered that only the inlet and outlet temperatures (hot fluid, cold fluid and wall) and the flow rate of fluids are measured. The temperatures inside heat exchanger are estimated. The quality of heat exchange depends especially by the heat transfer coefficients. These parameters depend by temperatures, accumulation of deposits of one kind or another on heat transfer surface, shape of tube, etc. The temperature distributions inside heat exchanger (process and model) are presented in fig. 2 using notations $\theta_a(i,j)$. for process and $M\theta_a(i,j)$ for the model.

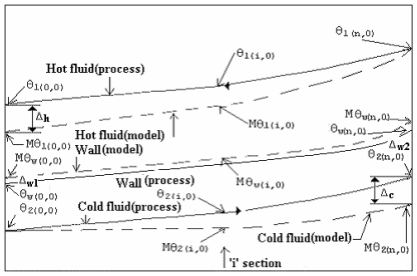


Figure 2: Process and model (counter flow) – diagrams.

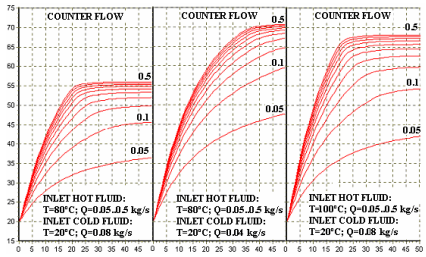


Figure 3: Step reply- counter flow.

To underline the main characteristics of the heat exchangers that are used in simulations, there are presented the step replies in some cases (counter flow - fig. 3; parallel flow – fig. 4). First, the temperatures of fluids are 20º C, than it is changed the inlet temperature of hot fluid (input of the process). There are different conditions for inlet temperatures and flow rate fluids. Flow rate of hot fluid is a parameter and permits to obtain a family of step replies.

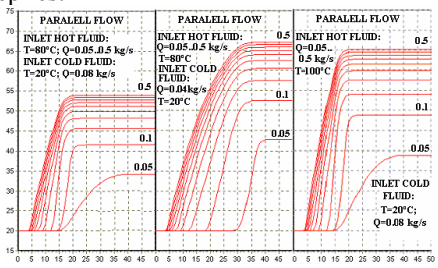


Figure 4: Step reply- parallel flow.

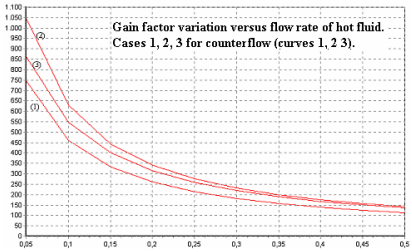


Figure 5: Counter flow- gain factor.

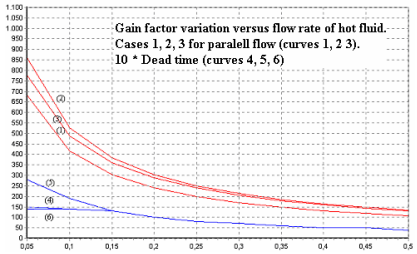


Figure 6: Parallel flow- gain, dead time.

Figures 5 and 6 present the dependence of gain factor and dead time by flow rate. These simulations underline the non-linear features of processes and, for parallel flow, a dead time, which is dependent especially by flow rate of hot fluid.

3 CONTROL ALGORITHM

A model based adaptive-predictive algorithm which uses on line simulation and rule based control, designed for linear processes, is developed in (Balan, 2001), (Balan, 2005). This algorithm can be applied with some modifies to nonlinear processes. The nonlinear equations of the process can be used directly in the control algorithm. The predictions of system output are calculated by integrating the nonlinear ordinary differential equations of the model over the prediction horizon, by using a few

control sequences (Balan, 2005). For a first stage, are used, the next four control sequences:

$$u_1(t)=\{u_{\min},u_{\min},..,u_{\min}\}$$
$$u_2(t)=\{u_{\max},u_{\min},..,u_{\min}\}$$
$$u_3(t)=\{u_{\min},u_{\max},..,u_{\max}\}$$
$$u_4(t)=\{u_{\max},u_{\max},..,u_{\max}\} \quad (17)$$

where u_{min} and u_{max} are the limits of the control signal, limits imposed by the practical constraints. These values can depend on context and can be functions of time. There are two pair sequences: ($u_1(t)$, $u_2(t)$) and ($u_3(t)$, $u_4(t)$) which are different through the preponderance of u_{min} or u_{max} in the future control signal. The pair sequences are different only through the first term.

Using these sequences results four output sequences y_1(t), y_2(t), y_3(t), y_4(t). The control signal is computed using a set of rules based on the extreme values y_{max0}, y_{max1}, y_{min0}, y_{min1} (fig. 7- d is dead time, t_1=N, y_r is setpoint) of the output predictions. In the followings, considering processes with positive sign, it can be put in evidence four usual cases:

Case 1: If $y_{max0}<y_r$ (corresponding to u_1(t) sequence) and $y_{max1}>y_r$ (corresponding to u_2(t) sequence) Then (using a linear interpolation):

$$u(t)=\frac{u_{\max}-u_{\min}}{y_{\max 1}-y_{\max 0}}y_r+\frac{u_{\min}y_{\max 1}-u_{\max}y_{\max 0}}{y_{\max 1}-y_{\max 0}} \quad (18)$$

Case 2: If $y_{min0}<y_r$ (corresponding to u_3(t) sequence) and $y_{min1}>y_r$ (corresponding to u_4(t) sequence) Then (using a linear interpolation):

$$u(t)=\frac{u_{\max}-u_{\min}}{y_{\min 1}-y_{\min 0}}y_r+\frac{u_{\min}y_{\min 1}-u_{\max}y_{\min 0}}{y_{\min 1}-y_{\min 0}} \quad (19)$$

Case 3: If: $y_{max0}>y_r$ Then $u(t_0)=u_{min}$ (20)

Case 4: If: $y_{max1}<y_r$ Then $u(t_0)=u_{max}$ (21)

In fig. 7, every output prediction curve is marked with a number which correspond to the number of control sequence from relations (17). Similar to case 3 and case 4, there are two similarly cases if $dy/dt<0$ for $t<t_0$. If the algorithm uses only these 6 rules, the variance of u(t) will be large (Balan, 2001).

So, in the second stage, depended by behaviour of the control system, are used next methods:

-an algorithm that modifies the limits of control signal:

$$u_{min}\le u_{minst}(t)\le u(t)\le u_{maxst}(t)\le u_{max}$$
$$\Delta u_{min}\le \Delta u\le \Delta u_{max} \quad (22)$$

For example:

$$u_{\text{minst}}(t)=f_1(u_{\text{minst}}(t-1),u_{\text{maxst}}(t-1),y(t),y_r(t)) \quad (23)$$
$$u_{\text{maxst}}(t)=f_2(u_{\text{minst}}(t-1),u_{\text{maxst}}(t-1),y(t),y_r(t)) \quad (24)$$

where f_1, f_2 are functions which decrease or increase (depended by behavior of the control system) the difference between u_{maxst}(t) and u_{minst}(t).

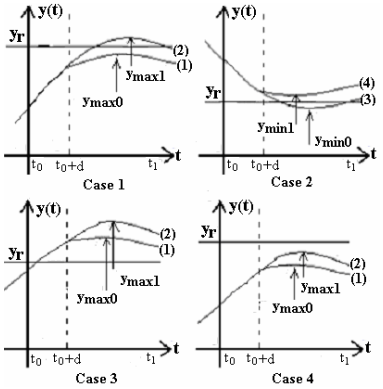


Figure 7: Examples of output predictions.

In relations (18)..(21), the values of u_{max}, u_{min} are replaced with u_{minst}(t), u_{maxst}(t). In the following, if is necessary, the next relations are used:

$$u_{\text{minst}}(t)=u_{\text{minst}}(t-1)+k_{st}(u_{st}-u_{\text{min}st}(t-1)) \quad (25)$$
$$u_{\text{maxst}}(t)=u_{\text{maxst}}(t-1)-k_{st}(u_{\text{max}st}(t-1)-u_{st}) \quad (26)$$

where k_{st} is a weight parameter and u_{st} is the estimated value of control signal in steady state. But in some circumstancing (perturbations, inaccurate model) the limits of control signal must increase. Also, it is necessary to limit the minimum value of $u_{maxst}(t)-u_{minst}(t)>d_{ust}>0$, where d_{ust} is a parameter of the control algorithm.

-using the "variable setpoint" (Balan, 2001):

$$y_{r1}(t)=y_r(t)+k_{\text{ref}}[y(t)-y_r(t)] \quad (27)$$

where k_{ref} is a weight factor

-using a filter to compute control signal (especially in steady state regime).

This paper will be tackled only the case when the main aim is to control the temperature of outlet cold fluid. To do this, it is used the flow rate of hot fluid (controller's output). There are possible other objectives for example to maximize the heat transfer between fluids. First, there was used an adaptive-predictive algorithm based on on-line simulation and a linear model (Balan, 2001). The parameters of model were identified on-line using least square algorithm. This method could be applied, with poor results, only for counter flow heat exchanger. It is

necessary to consider the non-linear features of heat exchanger and to use a model of the heat exchanger based on the finite difference method. It is supposed that initially the heat transfer coefficients are unknown and than they are identified on-line. In simulations, there are used three sets of finite difference equations: process equations, model equations, on line simulation equations.

The behaviour of heat exchanger depends by some types of parameters:

1. Construction parameters: length of tube, surface of heat transfer, diameters of tubes, etc. These parameters can be considered constants.
2. Fluids parameters: density, specific heat etc. These parameters depend by temperature and other conditions.
3. Parameters that determine the quality of heat exchange, especially the heat transfer coefficients. These parameters depend by temperatures, accumulation of deposits of one kind or another on heat transfer surface, shape of tube, etc.

At every sample period, it is possible to compute Δ_h, Δ_c, Δ_{w1}, Δ_{w2}, the temperature prediction errors of outlet hot fluid, outlet cold fluid, wall (fig. 2).

These predictions can be used to correct the temperature distributions inside the model of heat exchanger, using translations and rotations of distributions. Also, prediction errors can be used to modify the parameters of the model using an algorithm based on rules. The control scheme is presented in fig. 8.

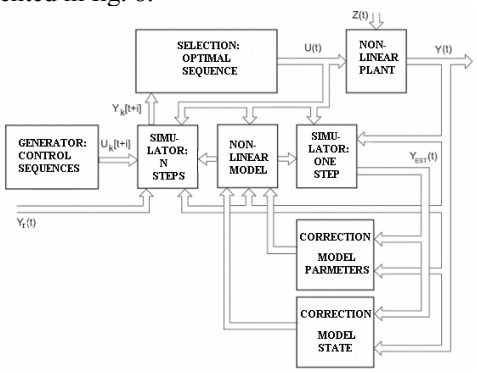


Figure 8: Control scheme.

4 APPLICATIONS WITH HEAT EXCHANGERS

The next applications show the main features of the algorithm applied to heat exchanger. The set point has a variable shape (42°C, 47°C, 52°C, 47°C, 42°C..). The limits of u(t) (hot fluid flow rate) are: $0.05 \le u(t) \le 0.5$ [kg/s]. The flow rate of cold fluid is constant (0.08 kg/s). The temperatures of cold fluid (20°) and hot fluid (80°) are constant. Some experiments with variable flow rate or/and variable temperature of cold fluid are presented in (Balan, 2001).

First, it is used an accurate model (Fig. 9, fig. 10). If the algorithm uses only 1..6 rules, the variance of $u(t)$ will be large. To reduce this variance, a solution is to use a funnel zone for control signal, based on inequality (22).

In steady-state regime, control signal is computed using average of past and new values. The algorithm do not use directly an integral component. In figure 9, steps 50..80, the algorithm tries to reduce the error as fast as possible. As a result, a damped oscillation appears. To avoid this behavior, a solution is to use a reference trajectory.

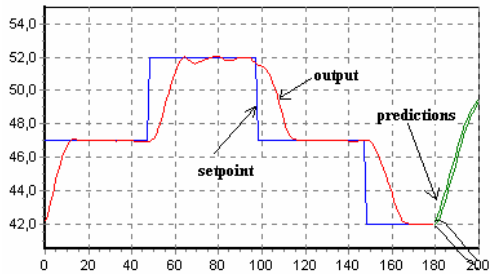


Figure 9: Setpoint, output (accurate model).

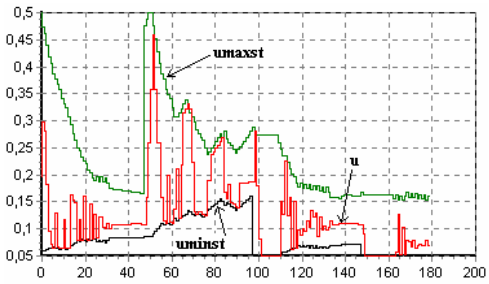


Figure 10: Controller output (accurate model).

In figure 11, 12 it is presented an adaptive case; the heat transfer coefficients depend by temperature:

$$k = k_0\left(1 + \theta / 200\right) \qquad (28)$$

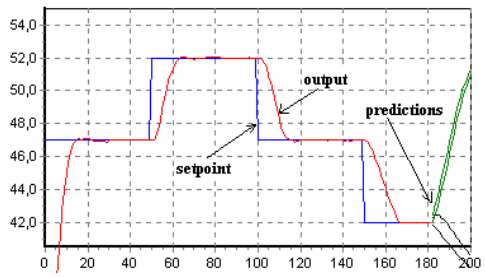


Figure 11: Setpoint, output (adaptive case).

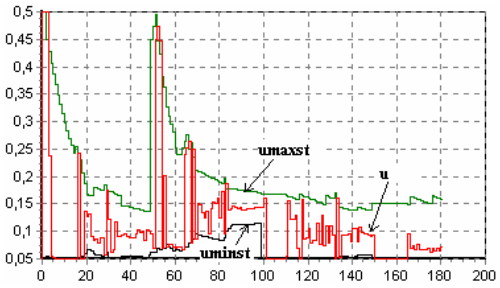


Figure 12: Controller output (adaptive case).

Initial the temperature of cold and hot fluids is $20°$. The evolution of the estimations of heat transfer coefficients is presented in figure 13. To obtain these estimations, both rotations and translations of temperature distributions and rule based correction of heat transfer coefficients are used. In figure 14 it is used the same conditions for heat transfer coefficients, but it is not used this approach.

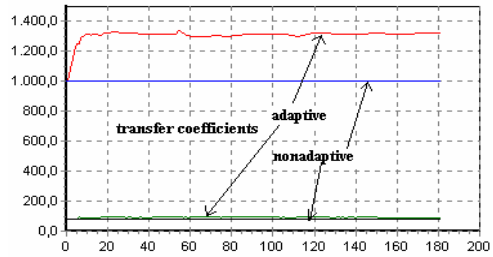


Figure 13: Parameters identification.

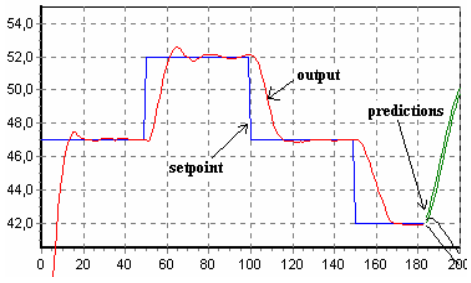


Figure 14: Setpoint, output (adaptive case).

As a result, the quality of control algorithm decreases.

5 CONCLUSION

The paper presents a simple and intuitive algorithm applied in the case of a non linear process: heat exchanger. A non-linear model of the process, based on finite difference method, is used. This approach is a numerical alternative to usual criteria equations; offer a way to ensure the accuracy of a best-fit heat exchanger selection, and point out that the fluids properties must not be mathematically emphases. Using the process model and a reduce number of the sequences control, it is simulated the future behaviour of the process and based on a set of rules it is chosen the signal control considered optimum at the actual moment. Of course there are some difficulties such as the proof of the stability, the way of choosing of the control sequences and the set of rules which will lead to a better result, choosing some parameters etc. Although, taking into account the simplicity of this algorithm the obtained results in the case of the presented examples by nonlinear systems are remarkable. A demo application that implements the proposed algorithm can be downloaded (see web link). In the future, starting from the proposed algorithm, the work will focus on: the optimal chosen of the control parameters, the study of other set of control sequences, the study of other set of control rules, adaptive case and practical implementation.

REFERENCES

Camacho E., Bordons C. (1999), "Model Predictive Control" Spriger-Verlag

Radu Balan: "Adaptive control systems applied to technological processes", Ph.D. Thesis 2001, Technical University of Cluj-Napoca Romania.

Dougherty, D., Cooper, D., "A practical multiple model adaptive strategy for a single loop", Control Engineering Practice 11 (2003) pp. 141-159

Fischer M., Nelles O., Fink A., "Adaptive Fuzzy Model Based Control" Journal a, 39(3), Pp22-28, 1998

Fink A., Topfer S., Isermann O., "Neuro and Neuro-Fuzzy Identification for Model-based Control", IFAC Workshop on Advanced Fuzzy/Neural Control, Valencia, Spain, Pages 111-116, 2001

Ozisik M. N., "Heat Transfer - A Basic Approach", McGraw-Hill Book Comp. 1985.

Douglas I.M., "Process dynamics and control", Prentice Hall Inc. 1972

Bălan, Radu, Vistrian Maties, Olimpiu Hancu, Sergiu Stan, A Predictive Control Approach for the Inverse Pendulum on a Cart Problem, IEEE-ICMA 2005 pag. 2026-2031 July 29 - August 1, 2005 Niagara Falls, Ontario, Canada.

Available online, accessed in March, 2007: http://zeus.east.utcluj.ro/mec/mmfm/download.htm

GPC BASED ON OPERATING POINT DEPENDENT PARAMETERS LINEAR MODEL FOR THERMAL PROCESS

Riad Riadi, Rousseau Tawegoum, Gérard Chasseriaux
[*] *Unité de Sciences Agronomiques appliquées à l'Horticulture SAGAH A_462, INH-INRA-UA*
Institut National d'Horticulture, 2, rue Le Nôtre 49045 Angers, France
riad.riadi@inh.fr

Ahmed Rachid
Université de Picardie Jules Verne. IUP GEII. 33 rue Saint LEU 80000 Amiens, France

Keywords: HVAC system, non linear system, generalized predictive control, operating point dependent parameters model, temperature control.

Abstract: This paper presents the application of generalized predictive control strategy (GPC) based on an OPDPLM (Operating Point Dependent Parameters Linear Model) structure to a heating and ventilation nonlinear-subsystem of a complex passive air-conditioning unit. For this purpose, several discrete-time models were identified with respect to measurable exogenous events. The parameters of the identified models change according operating conditions (sliding opening window). The objective of the studied subsystem was to guarantee a microclimate with controlled temperature set-points. The on line adaptive strategy was implemented to compute the controller parameters in order to adapt to the operating conditions variations. Efficiency of the resulting algorithm is illustrated by a real experiment.

1 INTRODUCTION

The consumption of energy by Heating, Ventilating, and Air Conditioning (HVAC) equipments in industrial and commercial buildings constitutes 50% of the world energy consumption (Arguello-Serrano and Vélez-Reyes 1999). Growing crop in greenhouse is one of important branch of agriculture industry and, it is labour intensive and technically challenging business. Optimized control helps to increase production despite saving precious sources (Young and Lees 1994). Standard air-conditioning units which are used in environment control for the growth chambers are usually composed of heating elements, a cooling system with compressor and evaporator techniques (Albright 2001), (Jones et *al*., 1984), (Hanan 1997). The air-conditioning unit studied is passive and does not use the more typical compression system or absorption-refrigeration cycle (Tawegoum et *al*., 2006a, Riadi et *al*., 2006). The specificity of the system is to produce a variable microclimate with variable temperatures and variable relative humidity set-point values.

A complete physical model of this plant developed in (Riadi et *al*., 2006), showed that the global system is complex and composed of three HVAC nonlinear subsystems. Therefore the implementation of centralized control strategy is cumbersome and decreases reliability. For these reasons, a typical local-loop control configuration for each subsystem of this air conditioning unit will be more efficient. For such control loops, self tuning controller parameters is usually considered, and the present study is focused only on one single-input-output (SISO) non linear subsystem, with multiple operating modes. Many adaptive strategies using recursive estimator are generally applied on thermal process (Arguello-Serrano and Vélez-Reyes 1999), (Landau and Dugard 1986), (Ljung 1999), especially, efficient, when parameters values are slowly varying. In our case, an idea about the model structure is possible and the parametric disturbance factor is measurable.

A generalized predictive control strategy, based on online controller parameters adaptation, was used to ensure stability and desired performances. Sub-systems operating points were modeled by a linear structures. These models should be fairly close in their structure but with different parameters values. The different models were identified for the main

operating points and the outdoor disturbance variations were taken into account and were parameterized using polynomial function interpolation in order to provide a single structure, called Operating Point Dependent Parameters Linear Model (OPDPLM) (Lakhdari et *al*., 1994), (Landau et *al*. 1987).

This paper proceeds as follows. The problem statement is presented in section II. In section III, the process control problem in presence of multiple operating modes is formulated and the proposed GPC (Generalized Predictive Control) based on OPDPLM is presented.The last section shows the real-time implementation and the experimental results are discussed.

2 SYSTEM DESCRIPTION AND PROBLEM STATEMENT

Figure 1 outlines the overall design of the proposed air-conditioning unit. The main features of the unit are: a humid climate, operation without a freezing unit, equilibrium of head losses and minimum energy consumption. Moisture removal is not required.

The unit is composed of two flows: a non-saturated flow (or non-saturated duct) and a saturated flow (saturated duct). As shown in Figure 1, in the saturated air flow, fresh air is saturated on with water after being heated by a coil resistor. Saturation operates at constant enthalpy (Chraibi et *al*., 1997). The saturated duct subsystem consists of a closed system, including a suction pump, a water tank and cross-corrugated cellulosic pads of the type using in cooling. The suction pump carries water from the tank to the top of the pads. Once the saturation steady state is reached, the pads contain a constant mass of water with a given water output and a given temperature.

In the unsaturated duct subsystem, fresh air is only heated by another resistor coil to a desired temperature

The proportional mixing of the two air flows is carried out by a sliding window driven by a DC motor.

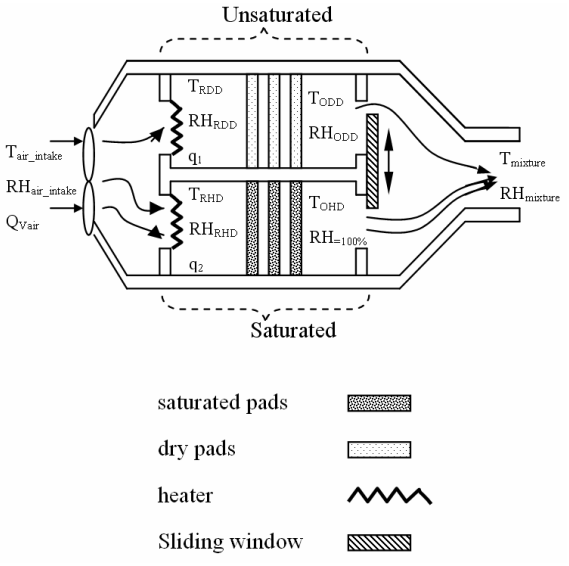


Figure 1: Complete air-conditioning system.

In this paper, we are interested in the unsaturated duct subsystem. The purpose control strategy is to regulate the temperature $T_{ODD}(t)$ (°C) of the outgoing air at constant temperature reference T_r (°C), in spite of air flows variations through the unsaturated duct and in spite of air intake temperature behaviour T_{air_intake} (°C). The air flow varied by changing the sliding window percentage, x (%). This latter is close to air flow by Eq(1).

$$q_1 = \alpha(x) Q_{Vair} \tag{1}$$

The heat balance in the unsaturated duct is given by the following equation:

$$\frac{dT_{ODD}}{dt} = \frac{\alpha(x) Q_{V\,air}}{V_{DD}} \left[T_{air\_intake} - T_{ODD} \right] + \frac{k_{RDD}}{\rho_{air} C_{air} V_{DD}} U_{DD} \tag{2}$$

with U_{DD} the applied voltage (V), proportional to the resistor heating in the dry duct, k_{RDD} the proportional coefficient between the voltage and the heating-power (J/sV), V_{DD} the volume of the dry duct (m^3).

3 GENERALIZED PREDICTIVE CONTROL OF NONLINEAR CLASS SYSTEM

3.1 The Operating Point Dependent Parameters-linear Model

The existence of a system with parameters depending on the operating point (the particular case of affine systems) means that one or more parameters of a linear differential equation vary according to an auxiliary variable ζ, which represents the operating point (Landau et *al*., 1987). An application to the temperature identification of a helium bath cryostat is presented in (Lakhdari et *al*., 1994), and it is also used in (Tawegoum et *al*., 2006b) for climate identification nearly steady weather conditions. This variable can be calculated via the input or output process, or via another measurable variable related to the operating point. The Operating Dependent Parameter Linear Model (OPDPLM) has the following properties:

- It allows the description of the non-linear phenomena with regard to the operating point ζ.
- It makes it possible to extend the linear formalism to systems that are not linear.

The system input-output form given by "(3), (4), and (5)" is as follows:

$$A(\zeta(t),d,q^{-1}).\Delta Y(t) = q^{-d}.B(\zeta(t),d,q^{-1}).\Delta U(t) \tag{3}$$

$A(\zeta(t),d,q^{-1})$ is polynomial in q^{-1}, depending on the delay d, nonlinear with respect to $\zeta(t)$, and defined by:

$$A(\zeta(t),d,q^{-1}) = 1 + \sum_{i=1}^{na} a_i(\zeta(t-d-i)).q^{-i} \tag{4}$$

The polynomial $B(\zeta(t),q^{-1})$ is non-linear in $\zeta(t)$, polynomial in q^{-1}, and is defined by

$$B(\zeta(t),q^{-1}) = \sum_{i=0}^{nb} b_i(\zeta(t-i)).q^{-i} \tag{5}$$

na, nb are respectively the polynomial degrees of

$A(\zeta(t),d,q^{-1})$ and of $B(\zeta(t),q^{-1})$, issued from the identification process.

The parameters $a_i(\zeta(t))$ and $b_i(\zeta(t))$ can be modeled by polynomial functions of order η_1 and η_2 as follows:

$$a_i(\zeta(t)) = \sum_{j=0}^{\eta_1} a_{ij}\zeta^j(t) \tag{6}$$

$$b_i(\zeta(t)) = \sum_{j=0}^{\eta_1} b_{ij}\zeta^j(t) \tag{7}$$

In our case, the percentage of the window opening represents the operating point ($\zeta = x$).

3.2 GPC Design based on OPDPLM

The basic idea of the GPC (Clarke et al., 1987 a), (Clarke et *al*., 1987 b), (Camacho and Bordons 1998) is to calculate a sequence of future control signals is such way that it minimizes a multistage cost function defined over a control horizon. The index to be optimized is normally the expectation of a function measuring the distance between the predicted system output and some predicted references sequence over the control horizon plus a function measuring the control effort on the same horizon.

Consider the plant described by CARIMA (Controlled Auto-Regressive Integrated Moving Average) model in OPDPLM case:

$$A\left(\zeta,q^{-1}\right)y(t) = B\left(\zeta,d,q^{-1}\right)u(t-1) + C\left(\zeta,q^{-1}\right)\frac{\varepsilon(t)}{\Delta\left(q^{-1}\right)} \tag{8}$$

The optimal j-step predictor defined between N_1 and N_2 is given by:

$$\hat{y}(t+j) = F_j\left(\zeta,q^{-1}\right)y(t) + H_j\left(\zeta,q^{-1}\right)\Delta u(t-1) + G_j\left(\zeta,q^{-1}\right)\Delta u(t+j-1) \tag{9}$$

Where polynomials F_j, G_j, H_j are solutions of the following Diophantine equations:

$$\Delta\left(q^{-1}\right)A\left(\zeta,q^{-1}\right)J_j\left(\zeta,q^{-1}\right) + q^{-j}F_j\left(\zeta,q^{-1}\right) = 1 \tag{10}$$

$$G\left(\zeta,q^{-1}\right) + q^{-j}H_j\left(\zeta,q^{-1}\right) = B\left(\zeta,q^{-1}\right)J_j\left(\zeta,q^{-1}\right) \tag{11}$$

The cost function is given by

$$J(N_1, N_2, N_u, \lambda) = E\left\{ \sum_{j=N_1}^{N_2} [(y(t+j) - w(t+j))]^2 + \sum_{j=N_1}^{N_u} \lambda[\Delta u(t+j-1)]^2 \right\} \quad (12)$$

with N_1 the minimum prediction horizon; N_2 the maximum prediction horizon, N_u the control costing horizon and λ a control-weighting.

Minimizing the cost function yields the control law

$$\Delta U_{opt} = \left(G^T(\zeta, d) \cdot G(\zeta, d) + \lambda I\right)^{-1} G^T(\zeta, d) \cdot (W - Y_1) \quad (13)$$

The previous equation can be written as:

$$\Delta U_{opt} = M(\zeta, d) \cdot (W - Y_1) \quad (14)$$

Where

$$M = \left(G^T(\zeta, d) \cdot G(\zeta, d) + \lambda I\right)^{-1} G^T(\zeta, d) \quad (15)$$

With G a $(N_2 - N_1 + 1) \times N_u$ matrix.

Its elements are the coefficients of step response depending on the operating point.

$W = [w(t+N_1), w(t+N_1+1), \cdots, w(t+N_2)]^T$ is the reference signal within the prediction horizon;

$Y_1 = [y_1(t+N_1), y_1(t+N_1+1), \cdots, y_1(t+N_2)]^T$ is the prediction based on the past measurements.

Notice that ΔU is not a scalar but a vector which can be written as:

$$\Delta U_{opt} = \left[\Delta u(t)_{opt} \;\; \Delta u(t+1)_{opt} \;\; \cdots \Delta u_{opt}(t+N_u)\right]^T \quad (17)$$

In real time control, only the first value of Eq (14) is finally applied to the system, according to the receding horizon strategy.

$$\Delta u_{opt} = m_1(\zeta, d) \cdot (W - Y_1) \quad (16)$$

With $m_1(\zeta, d)$ is the first line of the matrix M

The RST polynomial controller structure given by (Dumur et *al.,* 1997) can be extended for OPDPLM formalism (figure. 2) as:

$$S(\zeta, q^{-1})\Delta(q^{-1})u(t) = -R(\zeta, q^{-1})y(t) + T(\zeta, q)w(t) \quad (18)$$

With,

$$S(\zeta, q^{-1}) = 1 + m_1(\zeta)H(\zeta, q^{-1})q^{-1} \quad (19)$$

$$R(\zeta, q^{-1}) = m_1(\zeta)F(\zeta, q^{-1}) \quad (20)$$

$$T(\zeta, q) = m_1(\zeta)\left[q^{N1} \ldots . q^{N2}\right] \quad (21)$$

Where

$$F = \left[F_{N1}(\zeta, q^{-1}) \cdots F_{N2}(\zeta, q^{-1})\right]^T \quad (22)$$

$$H = \left[H_{N1}(\zeta, q^{-1}) \cdots H_{N2}(\zeta, q^{-1})\right]^T \quad (23)$$

The adapting phase of regulator parameters can be performed according stability and robustness, by considering the updating the controller parameters as: $N_1 = 1$, and $\lambda_{opt}(\zeta) = trace\left(G^T(\zeta) \cdot G(\zeta)\right)$.

The closed loop stability using the equivalent RST controller structure was studied in (Dumur et *al,* 1997), (M'saad and Chebassier.)

4 EXPERIMENTAL RESULTS

A set of electronic units was used to apply heating voltage on the resistors or to control the DC motor and thus, the window opening rate. Measurements were carried out using Pt100 sensors for temperature, and encoder sensors for position window. A sampling interval of Te=30 sec was chosen to satisfy the predominant time constant, and data acquisition time varied from two to four hours, depending on the operating point values $\zeta = x \in]0\%, 100\%[$ for a large interval variation .

4.1 Discrete Model Identification for Different Operating Modes

The air-flow measurements for the main window positions indicate a nonlinear relationship between the air-flow percentage and the window opening percentage (Tawegoum et *al*., 2006c). Therefore, in the identification process, the parameters of the model describing the output temperature behavior of the conditioning unit were assessed using the ARX model for each window position (*i.e.* for each operating point). A linear difference equation of the type of structure case is given in (Landau et *al*., 1987):

$$Y_{Mj}(k)+\sum_{i=1}^{nj}a_{ji}(x)Y_{Mj}(k-i)=\sum_{s=1}^{m}\sum_{i=1}^{nj}b_{jsi}(x)U_s(k-i-r_{js}) \quad (24)$$

The choice of ARX structures was based on their advantageous application in digital models, i.e. use of simpler and effective estimation algorithms and because of their easy and flexible usage in computer software (Borne et *al.*, 1990).

The ARX model obtained for the temperature model, in the non-saturated flow, is given in (Riadi et *al.*, 2006):

$$(1+a1(x)q^{-1}+a2(x)q^{-2}).T_{ODD}(t)=b1(x)q^{-1}.U_{DD}+b2(x)q^{-1}T_{air\_int\,ake}+e(t) \quad (25)$$

where $a_i(x)$ and $b_i(x)$ are four-degree polynomials, depending on x, the window opening percentage:

$$\begin{aligned}
a1(x) &= 13.6201\,x^4 - 34.3282\,x^3 + 30.6083\,x^2 - 11.0663\,x\\
a2(x) &= -3.4418\,x^4 + 8.8203\,x^3 - 8.2317\,x^2 + 3.0954\,x\\
b1(x) &= -1.6383\,x^4 + 3.6847\,x^3 - 2.7780\,x^2 + 0.7962\,x\\
b2(x) &= -0.4537\,x^4 + 1.0765\,x^3 - 0.8865\,x^2 + 0.3385\,x
\end{aligned} \quad (26)$$

4.2 Results of Strategy Control

The control parameters were chosen as follows: the minimum prediction horizon $N_1 = 1$, the maximum prediction horizon $N_2 = 14$, the control costing horizon $N_u = 7$, and the control weighting $\lambda(\zeta) = 0.93\ \ trace\left(G^T(\zeta)\cdot G(\zeta)\right)$.

Figure 3 illustrates the temperature response over a ten hours period, when applying the GPC strategy based on the OPDPLM elaborated in (13), subject to external temperature disturbances and to parameters disturbances by varying the window opening.

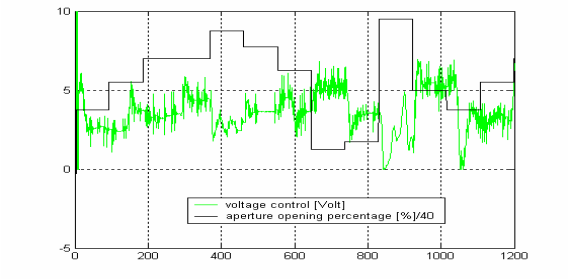


Figure 2: The window opening rate and voltage control.

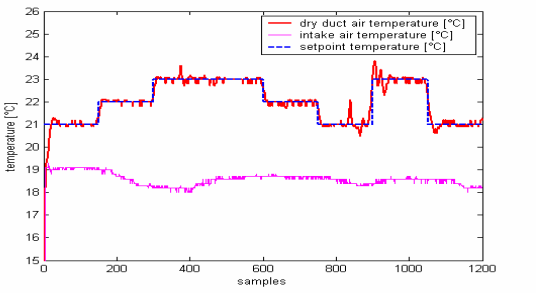


Figure 3: Regulation of temperature of non saturated using GPC based on OPDPLM.

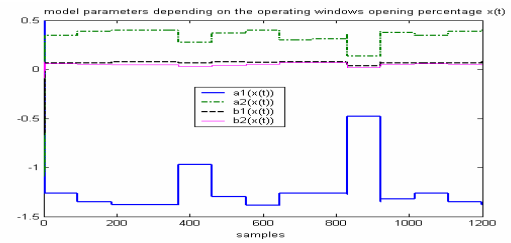


Figure 4: Parameter evolution of the model system.

The objectives of stability and energy minimization are reached (disturbances rejection and robustness stability).

As it can be seen the temperatures reach their set-points in a very short time, exhibiting a small overshoot. It can also be observed that an interaction exists between the variation of the aperture position and the output temperature.

Figure.3 shows that, consecutive to the switching parameters (window moving), oscillatory behavior appears on the output temperature response, due to the control input discontinuties. The GPC algorithm shows robustness in spite of these disturbances.

5 CONCLUSION

This paper has presented an application of the generalized predictive control using the OPDPLM structure of nonlinear thermal process. Stability is maintained with an adequate choosing of controller parameters values. The performances are maintained in spite of parameters system variation and controller disturbance rejection is capable to reduce the effect of thermal loads, with a simple updating of the regulator parameters depending on operating points.

The control strategies will be performed with an introduction of an overshoot constraint on the output temperature and with robust techniques of the GPC

algorithm.

Further investigations on the decentralized architecture make it possible to extend this local control strategy to other part of the complex air conditioning unit.

REFERENCES

Albright, L.D, Gates, R.S, Aravantis, K.G, and Drysdale, A.E. "Environment Control for Plants on Earth and Space," IEEE Control Systems Magazine, pp. 28-47, October 2001.

Arguello-Serrano, B, Vélez-Reyes, M. "*Non linear control of heting, ventilating, and air conditioning system with termal load estimation*," *IEEE Tranactions on Control Systems technology*, vol. 7, No. 1, January. 1999, pp. 56–63.

Borne, P., Dauphin Tanguy, G., Richard., J.P., Rotella, F., Zambrettakis, I. "*Modélisation et identification des processus*". Techniques de l'ingénieur, Technip 1990.

Camacho, E.F and Bordons, C "Model predictive Control'' Springer edition, 1998.

Chassériaux, G., Tawegoum , R., and Lelièvre, M. "*Thermal simulation of an air conditioning unit based on a heating system and humid corrugated pad*." 20th International Congress of Refrigeration" – Sidney (Australia)– sept. Paper code 720, 1990.

Chraibi, A., Makhlouf, S. and Jaffrin, A. "*Refroidissement évaporatif de l'air des serres* ", Journal de Physique, n°III. Juillet 1995. Dumur,D., Boucher,P., Murphy, K.M., Déqué, F. '' On Predictive Controller Design for Confort Control in Single Residential Housing''. ECC'97, juillet 1997.

Clarke, D.W., Mohtadi, C., and Tuffs, P.S., ''Generalized Predictive Control. Part I. The Basic Algorithm.'' Automatica, 23(2):137-148, 1987.

Clarke, D.W., Mohtadi, C., and Tuffs, P.S., ''Generalized Predictive Control. Part II. Extensions and interpretations'' Automatica, 23(2):149-160, 1987.

Dan Van Mien, H., and Normad-Cyrot, D. "*Non linear state affine identification methods, application to electrical power plant*", Automatica, Vol 20, pp. 175-188, 1984.

Dumur.D,P. Boucher, K.M. Murphy, F. Déqué "On predictive controller design for comfort control in single residential housing", ECC'97-juillet 1997.

Hanan, J.J. "*Greenhouse: advanced technology for protected horticulture*.", Chapter 4, pp. 236-260, 1997.

Hansen, J.M., and , Hogh Schmidt, K., "*A computer controlled chamber system design for greenhouse microclimatic modelling and control*. " Proc. Int. Sym.on Plant Production in closed Ecosystems. Acta Horticulturae, n°440, ISHS, pp. 105-110., 1996.

Jones, P., Jones., J.W., Allen, L.H. and Mishve J.W. "*Dynamic computer control of closed environmental plant growth chambers*, " Design and Verification. Transaction of ASAE. (American Society of Agricultural Engineers), pp. 879-888, 1984.

Lakhdari, Z., Lécluse, Y. and Provost, J., "*Dynamic control of a system with operating point dependent parameters, application to cryogenic*." In Proc of the international AMSE conference, Sytems Analysis, Control & Design, SYS'94, Lyon (France), July, vol. 2, pp. 243 -253. ISBN : 2-909214 -57-5, 1994.

Landau, I.D., Dugard, L. "*Commande adaptative aspect pratiques et théoriques*," , J. Masson, Ed. Paris , 1986, pp. 1–81.

Landau, I. D., Normand-Cyrot, B., and Montano, A. "*Adaptive control of a class of nonlinear discrete time systems: application to a heat exchanger*," in Proc. 28th Conference on Decision and Control, Los Angeles, Ca December 1987, pp. 1990-1995.

Ljung, L. "*System identification*", Theory for the user, Prentice Hall, 1999.

M'ssad, M., Chebassier, J. "*Commande adaptative des systèmes*," *Techniques de l'ingénieur*, vol. 7426, pp. 1–25.

Nybrant, T.G. "*Modelling and adaptive control of concurrent flow driers,* " Computers and Electronics in Agriculture, 1989, 3, 243-253.

Ramond,G., Dumur, D., Libaux, A., Boucher, P.,'' *Direct adaptive predictive control of an hydro-electric plant*'', Proceedings 10th, Conference on Control Apllications, pp. 606-611,Mexico,Septembre 2001.

Riadi, R., Tawegoum., R., Rachid, A, Chassériaux, G. ''*Modeling and Identification of a Passive Air-Conditioning Unit using the Operating Dependent Parameters-Structure*''. Presented CESA-2006: Computational Engineering in Systems Application, Beijing, Chine, 4-6 Octobre 2006.

Tawegoum, R., Teixeira, R. and Chassériax., G. "*Simulation of humidity control and temperature tracking in a growth chamber using a passive air conditioning unit*," Contol Engineering Practice Journal, 2006a, 14/8, 853-861.

Tawegoum R., Lecointre B., "A linear parametric model of an air conditioning unit with operating point dependent parameters under nearly steady weather conditions," 5 th Vienna Symposium on Mathematical Modelling, Vienna-Austria, February 2006b, Mech, 3.1-3.8.

Tawegoum,, R., Bournet, P.E, Arnould, J., Riadi R., and Chassériax., G. "*Numerical investigation of an air conditioning unit to manage inside greenhouse air temperature and relative humidity*," International Symposium on Greenhouse Cooling, Almeria-Spain, April 2006c.

Young, P.C. and Lees, M.J. "*Simplicity out of complexity in glasshouse climate modeling*, " in Proc. 2nd IFAC/ISHS Workshop on Mathematical an Control Application in agriculture and Horticulture, 12-15 september 1994, Silsoe, United kingdom, Acta Horticuturae N°406, pp.15-28.

SIMULATION AND FORMAL VERIFICATION OF REAL TIME SYSTEMS: A CASE STUDY

Eurico Seabra, José Machado, Jaime Ferreira da Silva
Mechanical Engineering Department, Enginnering School, University of Minho, 4800-058 Guimarães, Portugal
{eseabra, jmachado, jaimefs}@dem.uminho.pt

Filomena O. Soares
Industrial Electronics Department, Enginnering School, University of Minho, 4800-058 Guimarães, Portugal
fsoares@dei.uminho.pt

Celina P. Leão
Production System Department, Enginnering School, University of Minho, 4800-057 Braga, Portugal
cpl@dps.uminho.pt

Keywords: Real Time Systems, Plant Models, Object-Oriented Language, Formal Verification.

Abstract: This paper presents and discusses a case study that applies techniques of simulation together with techniques of formal verification. A new approach in the plant modelling for formal verification of timed systems is presented. The modelling of the plant was performed by using the object-oriented language Modelica with the library for hierarchical state machines StateGraph and the simulation results were used as input for the formal verification tasks, using the model checker UPPAAL. It is presented, in a more detailed way, the part of this work that is related to the plant simulation.

1 INTRODUCTION

Modern engineered systems have reached a high degree of complexity that requires systematic design methodologies and model-based approaches to ensure the correct and competitive performance. In particular, the use of digital controllers has been proven that small errors in their design may lead to catastrophic failures.

Recent years have witnessed a significant growth of interest in the modelling and simulation of physical systems. A key factor in this growth was the development of efficient equation-based simulation languages. Such languages have been designed to allow automatic generation of efficient simulation code from declarative specifications. The Modelica language (Fritzson *et al*. 1998, Elmqvist *et al*. 1999, Fritzson *et al*. 2002) and its associated support technologies have achieved considerable success through the development of specific libraries. However, a significant part of the software development effort is spent on detecting deviations from specifications and subsequently localizing the sources of such errors. The high-level of abstraction of equation-based models presents new challenges to modelling and simulation tools due to the large gap between the declarative specification and the executable machine code. This abstraction gap leads to difficulties in finding and correcting model inconsistencies and errors, which are not uncommon in the process of developing complex physical system models.

Among the several techniques of industrial controllers analysis available, Simulation (Baresi *et al.* 2000, Baresi *et al*. 2002) and Formal Verification (Moon, 1994, Roussel and Denis, 2002), can be distinguished due to their utility. In the research works on industrial controller's analysis, these two techniques are rarely used simultaneously. If the Simulation is faster to execute, it presents the limitation of considering only some system behaviour evolution scenarios. Formal Verification presents the advantage of testing all the possible system behaviour evolution scenarios but, sometimes, it takes a large amount of time for the attainment of formal verification results. In this

paper it is shown, as it is possible, and desirable, to conciliate these two techniques in the analysis of industrial controllers. With the simultaneous use of these two techniques, the developed industrial controllers are more robust and not subject to errors.

Using this approach, the command of those systems can be simulated and tested when the physical part of the machine still does not exist. This way of simulation allows to reduce the production times of the automation systems because the manufacture do not need the physical part of the machine for later perform tests and simulation of the command of the system. This paper is focused in the simulation of timed systems.

To accomplish our goals, in this work, the paper is organized as follows. In Section 1, it is presented the challenge proposed to achieve in this work. Section 2 presents a general presentation of the case study involving a system with two tanks, a heating device, a mixer device, level control sensors and valves to control the liquids flow. Further, it is presented the methodology to obtain the controller program deduced from an IEC 60848 SFC specification of the system desired behaviour. Section 3 is exclusively devoted to the plant modelling, being presented the adopted approach. Section 4 presents and discusses the obtained results on simulation performed with the Modelica Language. Finally, in Section 5, the main conclusions and future work are presented.

2 SYSTEM DESCRIPTION

The case study is an adapted version of the benchmark example presented by (Kowalewski *et al*. 2001) and (Huuck *et al*. 2001) that corresponds to an evaporator system.

The system (Fig. 1) consists of two tanks, where tank1 is heated and mixed, one condenser, two level analogue sensors (one for each tank) and four on-off valves.

In the normal operation, the system works as follows. Tank1 is filled with two solutions by opening valves V1 and V2. When the level becomes high, the valves are closed and liquids are mixed by a mixer device for dilution. After two time units, the heating device is switch on to increase the temperature of the solution. After 20 time units, the required concentration has been reached and the heater is switched off. Meanwhile, during the heating phase, part of the liquid has been evaporated and cooled by the condenser. The remaining part is drained in tank2 by opening the valve V3. When the first tank is empty, the mixer is stopped and the solution in tank2 stays for post-processing step, to stay liquid, for 32 time units. At that point, the valve V4 is opened to empty tank2.

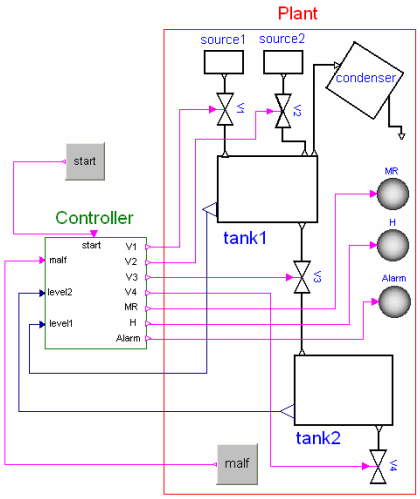


Figure 1: Scheme of the evaporator system.

Throughout normal operation mode, the system may malfunction. During evaporation, the condenser may fail: the steam can not be cooled and the pressure inside the condenser rises. Therefore, the heater must be switched off to avoid the condenser explosion. By doing so, the temperature of tank1 decreases and the solution may become solid and can not be drained in tank2. Hence, valve V3 must be opened early enough for preventing tank2 overflow, but after opening first valve V4.

In the case of a condenser malfunction, it is also necessary to ensure that some response times of the control program, taking into account the timing characteristics of the physical devices:

- whenever a condenser malfunction starts, the condenser can explode if steam is produced during 22 time units;
- if the heating device is switched off, the steam production stops after 12 time units;
- if no steam is produced in tank 1, the solution may solidify after 19 time units;
- emptying tank 2 takes between 0 and 26 time units;
- filling tank 1 takes 6 time units, at most.

2.1 Controller

In order to guarantee the desired behaviour of the evaporator system described above, the controller was developed according to IEC 60848 SFC specification, which is presented in Figure 2.

The PLC program which controls the process in closed-loop has input and output variables as described in Table1.

Table 1: Input/Output variables of the controller.

Input	Output
start – process start level1 – % fill tank1 level2 – % fill tank2 malf – condenser malfunction	V1 – open valve1 V2 – open valve2 V3 – open valve3 V4 – open valve4 H – switch Heater on MR – switch Mixer on Alarm – start alarm

The tank level is given in % of the fill tank. In this research work, the Boolean variables T1F (tank1 full) and T2F (tank2 full) were considered true when the level1 and level2 was greater than 0.98, respectively. On the other hand, the Boolean expression T1E (tank1 empty) and T2E (tank2 empty) were assumed true when the level1 and level2 was less than 0.01, respectively.

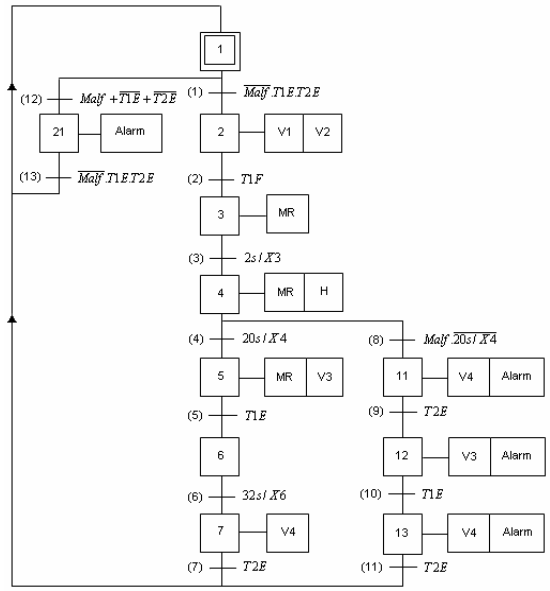


Figure 2: SFC of the system controller.

3 PLANT MODEL

The plant modelling was carried out in two steps. First, the plant was modelled using the Dymola program and the object-oriented language Modelica (Elmqvist and Mattson, 1997) with the library for hierarchical state machines StateGraph (Otter, 2005). Subsequently, the obtained models were used as a base to develop the UPPAAL (David *et al*. 2003) models that are used on formal verification tasks.

It should be highlighted that the most important data obtained by the Modelica simulation considered on the formal verification tasks is the set of simulation functioning delays. These delays are used to define the time units used on the UPPAAL modules of the plant model (Machado *et al*. 2007).

As the main aim of this paper deals with the plant simulation by using Modelica Language, it is only presented the modelling of tank1 by UPPAAL.

3.1 Tank1

The tank1 model is first simulated by using the Dymola software with the Modelica program code presented in the Figure 3. The obtained times from the simulation were used on formal verification with UPPAAL.

```
model Tank1
  Modelica.Blocks.Interfaces.RealOutput levelSensor;
  Modelica.StateGraph.Examples.Utilities.inflow inflow1;
  Modelica.StateGraph.Examples.Utilities.outflow outflow1;
  Real level "Tank level in % of max height";
  parameter Real A=1 "ground area of tank in m^2";
  parameter Real a=0.2 "area of drain hole in m^2";
  parameter Real hmax=1 "max height of tank in m";
  constant Real g=Modelica.Constants.g_n;
  Modelica.StateGraph.Examples.Utilities.inflow inflow2;
equation
  der(level) = (inflow1.Fi + inflow2.Fi - outflow1.Fo)/(hmax*A);
  if outflow1.open then
    outflow1.Fo = sqrt(2*g*hmax*level)*a;
  else
    outflow1.Fo = 0;
  end if;
  levelSensor = level;

end Tank;

connector Modelica.Blocks.Interfaces.RealOutput =
                output RealSignal "'output Real' as connector";

connector Modelica.Blocks.Interfaces.RealSignal
  "Real port (both input/output possible)"
  replaceable type SignalType = Real;

  extends SignalType;

end RealSignal;

connector Modelica.StateGraph.Examples.Utilities.inflow

    import Units = Modelica.SIunits;

  Units.VolumeFlowRate Fi "inflow";
end inflow;

connector Modelica.StateGraph.Examples.Utilities.outflow

    import Units = Modelica.SIunits;

  Units.VolumeFlowRate Fo "outflow";
  Boolean open "valve open";
end outflow;
```

Figure 3: Modelica program code for the model of tank1.

Figure 4 shows the corresponding model of the tank1 developed by UPPAAL for formal verification purposes.

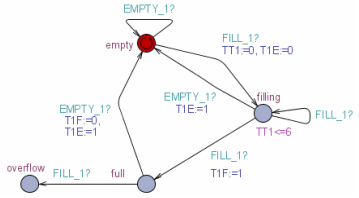


Figure 4: UPPAAL model of the tank1.

3.2 Tank2

The Modelica program code for modelling the tank2, presented in Figure 5, is similar to the code obtained for the tank1 model. The main difference between these two codes is due to the tanks have different numbers of fill sources. The tank 1 has two fill sources while the tank 2 has one only.

```
model Tank 2
  Modelica.Blocks.Interfaces.RealOutput levelSensor;
  Modelica.StateGraph.Examples.Utilities.inflow inflow1;
  Modelica.StateGraph.Examples.Utilities.outflow outflow1;
  Real level "Tank level in % of max height";
  parameter Real A=1 "ground area of tank in m²";
  parameter Real a=0.2 "area of drain hole in m²";
  parameter Real hmax=1 "max height of tank in m";
  constant Real g=Modelica.Constants.g_n;
equation
  der(level) = (inflow1.Fi - outflow1.Fo)/(hmax*A);
  if outflow1.open then
    outflow1.Fo = sqrt(2*g*hmax*level)*a;
  else
    outflow1.Fo = 0;
  end if;
  levelSensor = level;
end Tank;

connector Modelica.Blocks.Interfaces.RealOutput =
                output RealSignal "'output Real' as connector";

connector Modelica.Blocks.Interfaces.RealSignal
  "Real port (both input/output possible)"
  replaceable type SignalType = Real;

  extends SignalType;

end RealSignal;

connector Modelica.StateGraph.Examples.Utilities.inflow

    import Units = Modelica.SIunits;

  Units.VolumeFlowRate Fi "inflow";
end inflow;

connector Modelica.StateGraph.Examples.Utilities.outflow

    import Units = Modelica.SIunits;

  Units.VolumeFlowRate Fo "outflow";
  Boolean open "valve open";
end outflow;
```

Figure 5: Modelica program code for the model of tank2.

4 SIMULATION RESULTS

In order to perform the simulation, it is 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, 2006) with 500 output intervals was used.

In order to study the system behaviour different values for physical variables of the plant were used. Table 2 shows the variables considered in the simulation of the plant model.

Table 2: Variables of the plant.

Plant	Variable
source1, source 2	Q1, Q2 - flow rate [m^3/s]
tank1, tank2	G1, G2 – ground area [m^2] Ht1, Ht2 –height [m] A1, A2 – drain hole area [m^2]

The first two simulation performed were devoted to verify if the SFC of the controller system (Fig.2) modelled with Modelica language with the library for hierarchical state machines StateGraph simulated correctly the evaporator system, respectively, in their normal and malfunction operation. The values for the plant variables considered in these simulations were Q1=1, Q2=0.5, G1=G2=1, Ht1=Ht2=1, A1=0.2 and A2=0.05.

Figures 6 and 7 show results of the simulation without the occurrence of the condenser malfunction during the production cycle, which corresponds to the normal operation, respectively, for the level tanks and for the controller outputs.

Observing Figures 6 and 7 it can be concluded that the system is properly simulated by the developed program, since during the time specified by the SFC the tanks remain filled and empty, as well as, the switch logical state of the controller outputs.

On the other hand, Figures 8 and 9 show results of the simulation with the occurrence of the condenser malfunction during the production cycle, which corresponds to the malfunction operation. The malfunction occurred in a random way 15s after the start of the plant functioning.

Analysing Figures 8 and 9 it can be also concluded that the malfunction operation is properly simulated by the proposed program. Because it can be verified, taking into account the Figure 8, that at the malfunction occurrence (time 15s) the solution present in the tank1 is immediately drained for the tank2 and later emptied. In the same way, analysing the Figure 9, it can be verified that at the time 15s occur simultaneous the switch off the mixer and the heater and the alarm switch on, which corresponds to the SFC specification of the controller.

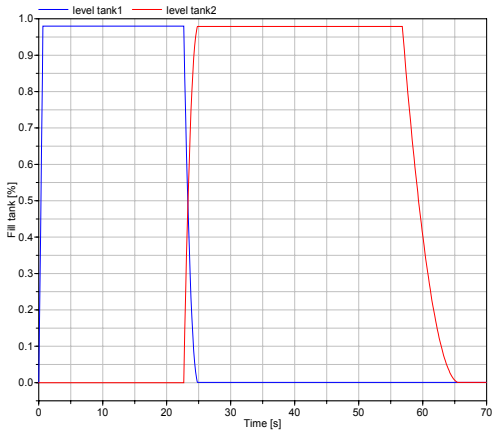


Figure 6: Level tanks in function of time in normal operation of the evaporator system.

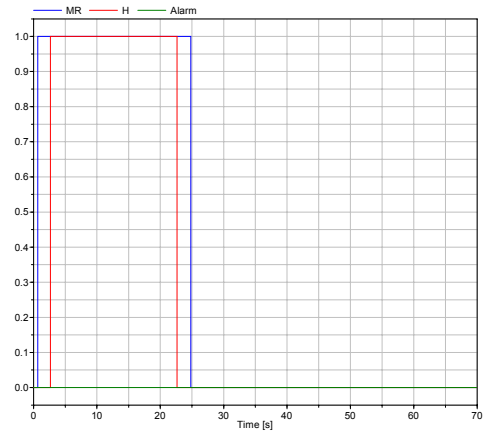


Figure 7: Switch state of the mixer, heater and alarm in normal operation of the evaporator system.

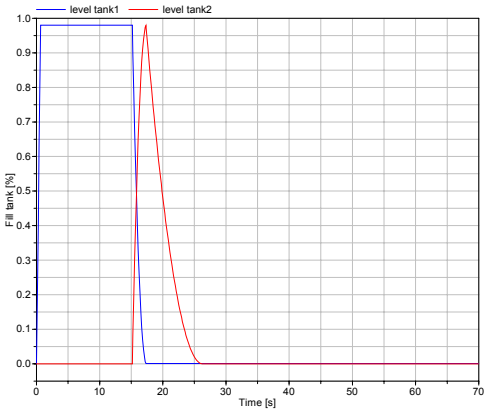


Figure 8: Level tanks in function of time with occurrence of condenser malfunction (time = 15s).

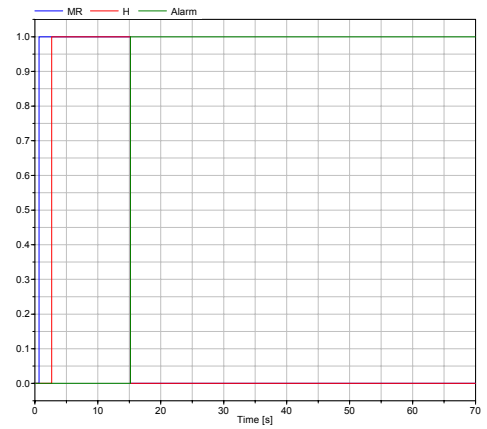


Figure 9: Switch state of the mixer, heater and alarm with occurrence of condenser malfunction (time =15s).

It becomes still necessary, in addition to the verification that the modelling of the system obey to the SFC of the system controller, to guarantee that in the case of condenser malfunction don't occur solidification or explosion of the solution in the tanks. Thus it is necessary taking into account the timing characteristics of the physical devices.

For example, the simulation presented in Figures 8 and 9, which values for the plant variables considered were Q1=1, Q2=0.5, G1=G2=1, Ht1=Ht2=1, A1=0.2 and A2=0.05, the obtained times for fill and empty the tank1 were, respectively, 0.6533s (limit 6s) and 2,1255s (limit 19s) and for the tank 2 respectively for fill and empty were 2,2655s (similar to the time of empty tank1) and 8,4361s (limit 26s). This simulation allowed to show that with these values of plant variables the system doesn't have serious functioning anomalies that can put in risk humans lives and material goods.

In order to obtain the relation between the plant variables and the time of the critical operations, some simulations were performed using several values of plant variables. Figure 10 and 11 show results of these simulations, respectively related to the empty tanks time (equal for the tank1 and tank2) and fill tank1 time, which correspond at the times of the critical operations of the evaporator system.

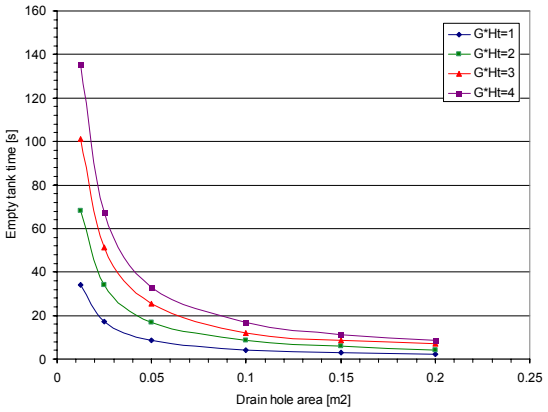


Figure 10: Empty tank time in function of plant variables.

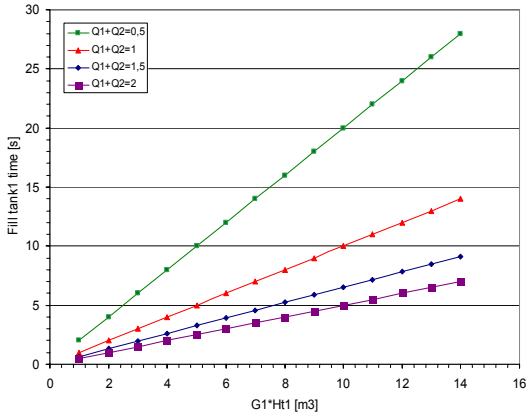


Figure 11: Fill tank1 time in function of plant variables.

5 CONCLUSIONS

The simulation used to evaluate the plant behaviour has been developed and proposed in this paper.

The results obtained suggested that this approach is adequate to obtain the relation between the plant variables involved in the evaporator system. The present research proved to be successful using the Modelica programming Language to obtain plant models and to get functioning delays in which a property can, or not, be proved using techniques of formal verification. Moreover, the simulation techniques allow us to test different delays of the plant functioning and to see if a property, for different considered delays, is still true or if different delays imply that a property is true and after is false.

For the analysis of a system controller program it is desirable the use of simulation before using formal verification. With the simulation it is possible to eliminate a set of program errors of some possible system behaviours in reduced intervals of time. This would not happen, in most of the cases, if these errors were detected only through the use of formal verification techniques. Conciliating these two techniques the time necessary for the attainment of results through the use of the formal verification technique can be substantially reduced. With this approach a manufacturer of industrial automated systems does not need the physical part of the machine for later perform tests and simulation of the system controller. In consequence, they allow, together, to reduce the times of production of the automated systems.

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

Baresi L., Mauri M., Monti A., Pezzè M., 2000. PLCTOOLS: Design, Formal Validation, and Code Generation for Programmable Controllers. *Special Session at IEEE Conference on Systems, Man, and Cybernetics*. Nashville USA.

Baresi L., Mauri M., Pezzè M., 2002. *PLCTools: Graph Transformation Meets PLC Design*. Electronic Notes in Theoretical Computer Science 72 No. 2.

Basu S., Pollack R., Roy M., 2006. *Algorithms in Real Algebraic Geometry - Algorithms and Computation in Mathematics*. Springer Editions, vol. 10, 2ªedition.

David A., Behrmann G., Larsen K. G., Yi W., 2003. *A Tool Architecture for the Next Generation of UPPAAL*. Technical Report n. 2003-011, Department of Information Technology, Uppsala University, Feb. 20 pages.

Elmqvist E., Mattson S., 1997. An Introduction to the Physical Modelling Language Modelica. *Proceedings of the 9th European Simulation Symposium, ESS'97*. Passau, Germany.

Elmqvist, Hilding, Mattsson S., Otter M., 1999. Modelica - a language for physical system modeling, visualization and interaction. *Proceedings of the IEEE Symposium on Computer-Aided Control System Design*. August, Hawaii.

Fritzson, Peter, Vadim E., 1998. Modelica, a general object-oriented language for continuous and discrete-event system modeling and simulation, *12th European Conference on Object-Oriented Programming (ECOOP'98)*. Brussels, Belgium.

Fritzson, Peter, Bunus P., 2002. Modelica, a general object-oriented language for continuous and discrete-event system modelling and simulation. *Proceedings of the 35th Annual Simulation Symposium*. April, San Diego, CA.

Huuck R., Lukoschus B., Lakhnech. Y., 2001. *Verifying Untimed and Timed Aspects of the Experimental Batch Plant*. European Journal of Control, vol. 7, nº 4, pp. 400-415.

Kowalewski S., Stursberg O., Bauer. N., 2001. *An Experimental Batch Plant as a Test Case for the Verication of Hybrid Systems*. European Journal of Control.

Machado J., Seabra E., Soares F., Campos J., 2007. A new Plant Modelling Approach for Formal Verification Purposes. *Submitted at 11th IFAC/IFORS/IMACS/ IFIP Symposium on Large Scale Systems: Theory and Applications*. Gdansk, Poland.

Moon I. 1994. *Modeling programmable logic controllers for logic verification*. IEEE Control Systems, 14**,** 2, pp. 53-59.

Otter M., Årzén K., Dressler I., 2005 StateGraph - *A Modelica Library for Hierarchical State Machines*. Modelica 2005 Proceedings.

Roussel M., Denis B., 2002. *Safety properties verification of ladder diagram programs*. Journal Européen des Systèmes Automatisés, vol. 36, pp. 905-917.

IMPLEMENTATION OF RECURRENT MULTI-MODELS FOR SYSTEM IDENTIFICATION

Lamine Thiaw, Kurosh Madani
Laboratoire Image, Signal et Systmes Intelligents (LISSI / EA 3956) IUT de Snart, Universit Paris XII
Av. Pierre Point, F-77127 Lieusaint, France
lthiaw@ucad.sn, madani@univ-paris12.fr

Rachid Malti
Laboratoire Automatique Productique et Signal University Bordeaux 1 351, Cours de la Libration, 33405 Talence Cedex, France
rachid.malti@laps.u-bordeaux1.fr

Gustave Sow
LER, Ecole Suprieure Polytechnique de Dakar, Universit Cheikh Anta Diop, BP 5085, Dakar Fan, Senegal
gsow@ucad.sn

Keywords: System identification, non-linear systems, multi-model, recurrent models.

Abstract: Multi-modeling is a recent tool proposed for modeling complex nonlinear systems by the use of a combination of relatively simple set of local models. Due to their simplicity, linear local models are mainly used in such structures. In this work, multi-models having polynomial local models are described and applied in system identification. Estimation of model's parameters is carried out using least squares algorithms which reduce considerably computation time as compared to iterative algorithms. The proposed methodology is applied to recurrent models implementation. NARMAX and NOE multi-models are implemented and compared to their corresponding neural network implementations. Obtained results show that the proposed recurrent multi-model architectures have many advantages over neural network models.

1 INTRODUCTION

Identification of nonlinear systems is an important task for many real world applications such as process behavior analysis, control, prediction, etc. In the last years, several classes of models have been developed, among which Artificial Neural Networks (ANN) and multi-models (also known as operating regime approach), for non linear system identification.

ANN are widely used for dynamical nonlinear system modeling (Cheng et al., 1997; Konur and Okatan, 2004; Vartak et al., 2005). Such implementations like Time Delay Neural Network (TDNN) (Corradini and Cohen, 2002; Konur and Okatan, 2004), Jordan Network (Jordan, 1986), Elman Network (Elman, 1999) are very suitable for time series applications but they suffer of some limitations which restrict their use (Huang et al., 2005; Tomasz and Jacek, 1997). Several papers have been dedicated to the enhancement of neural networks for recurrent models identification (Bielikova, 2005; Huang et al., 2006). In (Huang et al., 2006) for example, a Multi-Context Recurrent Neural Network (MCRN) is studied and its performances are compared with those of the Elman Network and Elman Tower Network. Even though the proposed MCRN allows to achieve good performances, the main drawback remains its complexity due to the number of parameters induced by the context layer (Huang et al., 2006) which has weighting connections with both hidden and output layers.

The main difficulty encountered in recurrent neural networks is parameters estimation complexity. The parameters estimation is mostly performed using the gradient descent method (Backpropagation Through Time algorithm, Real Time Recurrent Learning algorithms, etc.) which cannot guarantee convergence to global minimum. On the other hand, the related algorithm's performance is very sensitive to the learning rate parameter which determines the convergence rate and the stability of the algorithm.

Multi-models have recently been proposed in numerous papers (Boukhris et al., 2000; Vernieuwe et al., 2004; Li et al., 2004) for modeling and control of nonlinear systems. For such systems, it is gen-

erally difficult to find a single analytical relationship describing system's behavior in its whole operating range. The system's complexity can be considerably reduced if system's operating range is divided into different regions where local behavior could be described with relatively simple models. The system's behavior is approximated by the weighted contribution of a set of local models. The difficulty encountered in this approach is the splitting of the system's operating range into convenient regions. For that purpose, various techniques have been studied among which grid partitioning, decision tree partitioning, fuzzy clustering based partitioning (see (Vernieuwe et al., 2004; Murray-Smith and Johansen, 1997)). Fuzzy clustering based partitioning enables to gather those data that may have some "similarities", facilitating system's local behavior handling. The main difficulty is the number of clusters needed to determine the multi-model's architecture. A method is presented here to bypass this difficulty.

Parameter estimation of recurrent multi-models is much simpler as compared to recurrent neural networks. We present in this work a multi-model implementation of recurrent models with polynomial local models. The proposed structure is applied to NARMAX and NOE models. The main advantage of such structure is that it allows to adjust the complexity of local models to the detriment of global one and vice-versa. Parameters are estimated using least squares algorithms, avoiding time consuming calculations and local minima.

The paper is structured as follows: in section 2 an overview of models identification principle is presented. Section 3 describes the general principle of multi-models using polynomial local models. The implementation of recurrent multi-model is presented in section 4. Results and discussions are presented in section 5.

2 OVERVIEW OF NON LINEAR MODELS

"Black box" models are very suited for complex systems representation (Sjoberg et al., 1995). Identification of such models consists of determining the mathematical relationship linking system's outputs (or its states) to its inputs from experimental data. In general, model describing system's behavior[1] can be expressed as:

$$y(t+h) = F_0\left(\underline{u}(t), \underline{\tilde{y}}(t)\right) + e(t+h) \qquad (1)$$

where :
$y(t+h)$ is the unknown system output at time instant $t+h$;
t is the current time instant and h is the prediction step;
$F_0(\cdot)$ is an unknown deterministic nonlinear function describing the system (the true model);
$\underline{u}(t)$ is a column vector which components are system's inputs at time t and at previous time instants;
$\underline{\tilde{y}}(t)$ is a column vector which components are obtained from system's output at time t and at previous time instants. It can be built from measured output data, estimated output data, prediction errors, or simulation errors;
$e(t+h)$ is an error term at time $t+h$.

The identification task consists of determining the function $F(\cdot)$ which is the best approximation of $F_0(\cdot)$ and estimating the system's output $\hat{y}$:

$$\hat{y}(t+h) = F\left(\underline{u}(t), \underline{\tilde{y}}(t), \theta\right) = F\left(\underline{\varphi}(t), \theta\right) \qquad (2)$$

where :
$\underline{\varphi}(t) = [\underline{u}(t)^T, \underline{\tilde{y}}(t)^T]^T$ is the regression vector obtained by the concatenation of the elements of vectors $\underline{u}(t)$ and $\underline{\tilde{y}}(t)$; and θ is a parameter vector to be estimated.
If $\underline{\tilde{y}}(t)$ in (2) depends on model's output or model's states, then the model (2) is said to be recurrent. Recurrent models have the ability to take into account system's dynamics. On the other hand, data collected from a process are usually noisy due to the sensors or the influence of external factors. Recurrent models allow to obtain unbiased parameters estimation. Various model classes have been established for modeling dynamical systems in presence of various noise configurations. Model classes differ by the composition of their regression vector. Since the exact model class is frequently unknown various classes are usually tested and the best one is chosen. In this work we focus on recurrent models called Nonlinear AutoRegressive Moving Average with eXogenous inputs (NARMAX) and Nonlinear Output Error (NOE) models. These classes of models are widely used because of their ability to capture nonlinear behaviors.

NARMAX model is a very powerful tool for modeling and prediction of dynamical systems (Gao and Foss, 2005; Johansen and Er, 1993; Yang et al., 2005). It is well suited for modeling systems using noisy outputs and noisy states. It generalizes the Nonlinear AutoRegressive with eXogenous inputs (NARX) model. Its regression vector is composed of the past inputs

[1] Multi-input and single-output (MISO) systems are considered here for ease of understanding. Results can be generalized to multi-input and multi-output systems.

u_k, the past measured outputs y_s, and the past prediction errors (difference between measured and predicted outputs) e. The output of the NARMAX model is given by:

$$\begin{aligned} y(t+1) \quad = \quad & F\big(u_1(t-d_{u_1}+1),\ldots, \\ & \ldots \\ & u_k(t-d_{u_k}+1),\ldots,u_k(t-d_{u_k}-n_{u_k}+2) \\ & \ldots \\ & y_s(t-d_{y_s}+1)\ldots,y_s(t-d_{y_s}-n_{y_s}+2), \\ & e(t-d_e+1),\ldots,e(t-d_e-n_e+2)\big) \\ & +e(t+1) \end{aligned} \tag{3}$$

where:
d_{u_k}, d_{y_s} and d_e are inputs, output, and error delays respectively;
n_{u_k}, n_{y_s} and n_e are inputs, output, and error orders respectively;
The prediction step in this representation corresponds to:

$$h = \min(d_{u_k}, d_{y_s}, d_e) \tag{4}$$

A NOE model is suited for system's simulation because it does not require measured outputs (Palma and Magni, 2004). The corresponding regression vector is composed of past inputs u_k and past simulated outputs $\hat{y}_u$. The output of the NOE model is given by:

$$\begin{aligned} y(t+1) \quad = \quad & F\big(u_1(t-d_{u_1}+1),\ldots, \\ & \ldots \\ & u_k(t-d_{u_k}+1),\ldots,u_k(t-d_{u_k}-n_{u_k}+2) \\ & \ldots \\ & \hat{y}_u(t-d_{\hat{y}_u}+1),\ldots,\hat{y}_u(t-d_{\hat{y}_u}-n_{\hat{y}_u}+2)\big) \\ & +e(t+1) \end{aligned} \tag{5}$$

Identification of recurrent models such as NARMAX or NOE models is a difficult task because some of the regressors have to be computed at each time step. The parameter estimation must then be carried out recursively.

3 MULTI-MODEL'S PRINCIPLE

Multi-models were first proposed by Johansen and Foss in 1992 (Johansen and Er, 1992). A multi-model is a system representation composed by a set of local models each of which is valid in a well defined feature space corresponding to a part of global system's behavior. The local validity of a model is specified by an activation function which tends to 1 in the feature space and tends to zero outside. The whole system's behavior can then be described by the combination of all local models outputs. Figure 1 presents the basic architecture of a multi-model. The relation (2) can then be expressed as:

$$\hat{y}(t+h) = \sum_{i=1}^{M} \omega_i\big(\underline{\xi}(t)\big)\ f_i\Big(\underline{\varphi}(t), \theta_i\Big) \tag{6}$$

where:
M is the number of local models;
$\omega_i(\cdot)$ is the activation degree of local model $f_i(\cdot)$, with :

$$\omega_i\big(\underline{\xi}(t)\big) \in [0,\ 1]\ , \quad \sum_{i=1}^{M} \omega_i\big(\underline{\xi}(t)\big) = 1 \quad \forall t$$

$\underline{\xi}(t)$ is the vector of indexing variables (variables whereby system's feature space is divided into subspaces (Orjuela et al., 2006));
θ_i is a parameter vector characterizing the local model $f_i(\cdot)$;
$f_i\big(\underline{\varphi}(t), \theta_i\big) = \hat{y}_i(t+h)$ is the predicted output of the ith local model.

In (6), the prediction step h may take any discrete

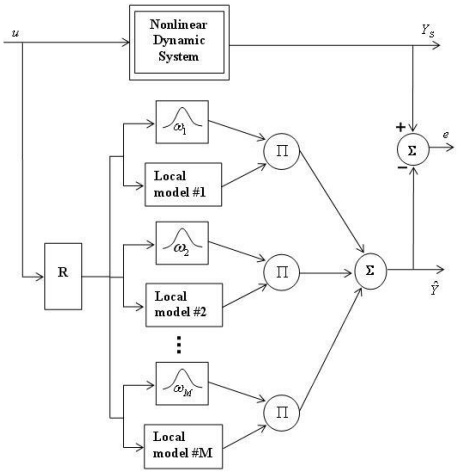


Figure 1: Basic architecture of a multi-model. Bloc R is a set of time delay operators combined with a linear or nonlinear transformation and used for the regression vector construction; Y_s is the measured system output.

value. It can also be specified by an appropriate choice of the time delays d_u, d_y and d_e of regressors in $\underline{\varphi}(t)$ (see equation (4)). So without loss of generality, we will assume that $h = 1$.

Activation degrees of local models can be defined in a deterministic way using membership functions like gaussian functions, sigmoidal functions, etc. They can also be defined fuzzily using a fuzzy clustering of the system's feature space. This latter solution seems to be more natural as it allows to gather data which may have some "similarities". The main difficulty is the determination of the number of

clusters. The proposed implementation combines architectural (number of local models or clusters) and parametrical identification. The number of clusters is successively incremented and the parameters are estimated at each step. The incrementation of the number of clusters is stopped when Akaike Information Criterion (see section §5) starts deteriorating.

The "fuzzy-c-means" algorithm (Bezdec, 1973) is implemented here because of its simplicity. This algorithm consists of maximizing the intra-cluster similarities and minimizing the inter-cluster similarities. The corresponding objective function is defined as:

$$J(c_1, c_2, \ldots, c_M) = \sum_{i=1}^{M} \sum_{t=1}^{N} \mu_{it}^m \, d_{it}^2 \qquad (7)$$

where:
$d_{it} = \|\underline{\varphi}(t) - \underline{c_i}\|$ denotes the distance between the observation $\underline{\varphi}(t)$ ($t = 1, \ldots, N$, N - number of observations) and the center $\underline{c_i}$ of the ith cluster ($i = 1, \ldots, M$, M - number of clusters or local models);
$\mu_{it} = \frac{1}{\sum_{k=1}^{M} (\frac{d_{it}}{d_{kt}})^{2/(m-1)}}$ represents membership degree of the observation $\underline{\varphi}(t)$ in the cluster i and stands for the local model's activation degree for that observation: $\mu_{it} = \omega_i\big(\underline{\xi}(t)\big)$;
$\underline{c_i} = \frac{\sum_{t=1}^{N} \mu_{it}^m}{\sum_{t=1}^{N} \mu_{it}^m \, \underline{\varphi}(t)}$ is the center of the ith cluster;
$m \geq 1$ is the "fuzzy exponent" and represents the overlapping shape between clusters (generally, $m = 2$).

Local models may be of any structural type. As suggested in (Johansen and Er, 1993), local models may be defined as the first p terms of the Taylor's series expansion of the true (unknown) model $F_0(\cdot)$ about a point located in the local model's feature space. Affine local models ($p = 1$) are mostly used because of their simplicity. This multi-model structure is very close to Takagi-Sugeno one. For complex systems, the number of linear local models may be very important because of the simplicity of their structure. We propose in this work polynomial local models with $p \geq 1$ which enable to enhance the handling of local nonlinearities, reducing then the number of models. We use a nonlinear transformation of the regression vector:

$$\underline{\varphi_p}(t) = g_p\big(\underline{\varphi}(t)\big)$$

where $g_p(\cdot)$ is a nonlinear transformation producing the new regression vector $\underline{\varphi_p}(t)$ which components are the products of elements of $\underline{\varphi}(t)$ at orders 1 to p. $\underline{\varphi_p}(t)$ can be easily obtained from the following procedure:
Let

$$\underline{\varphi}(t) = [\varphi_1 \quad \varphi_2 \quad \cdots \quad \varphi_{n_\varphi}]^T \qquad (8)$$

where n_φ is the dimension of $\underline{\varphi}(t)$.
Let us consider the following row vectors:

$$\begin{aligned}
V_{1,1} &= [\varphi_1 \quad \varphi_2 \quad \cdots \quad \varphi_{n_\varphi}] \\
V_{1,2} &= [\varphi_2 \quad \cdots \quad \varphi_{n_\varphi}] \\
&\cdots \\
V_{1,n_\varphi} &= [\varphi_{n_\varphi}] \\
V_{2,1} &= [\varphi_1 V_{1,1} \quad \varphi_2 V_{1,2} \quad \cdots \quad \varphi_{n_\varphi} V_{1,n_\varphi}] \\
V_{2,2} &= [\varphi_2 V_{1,2} \quad \cdots \quad \varphi_{n_\varphi} V_{1,n_\varphi}] \\
&\cdots \\
V_{2,n_\varphi} &= [\varphi_{n_\varphi} V_{1,n_\varphi}] \\
&\cdots \\
V_{p-1,1} &= [\varphi_1 V_{p-2,1} \quad \varphi_2 V_{p-2,2} \quad \cdots \quad \varphi_{n_\varphi} V_{p-2,n_\varphi}] \\
V_{p-1,2} &= [\varphi_2 V_{p-2,2} \quad \cdots \quad \varphi_{n_\varphi} V_{p-2,n_\varphi}] \\
&\cdots \\
V_{p-1,n_\varphi} &= [\varphi_{n_\varphi} V_{p-2,n_\varphi}] \\
V_{p,1} &= [\varphi_1 V_{p-1,1} \quad \varphi_2 V_{p-1,2} \quad \cdots \quad \varphi_{n_\varphi} V_{p-1,n_\varphi}]
\end{aligned}$$

$\underline{\varphi_p}(t)$ is then obtained from the relation:

$$\underline{\varphi_p}(t) = [V_{1,1} \quad V_{2,1} \quad \cdots \quad V_{p,1}]^T \qquad (9)$$

For example if $\underline{\varphi}(t) = [\varphi_1 \quad \varphi_2 \quad \varphi_3]^T$ and $p = 2$, then relation (9) gives:

$$\underline{\varphi_2}(t) = [\varphi_1 \;\; \varphi_2 \;\; \varphi_3 \;\; \varphi_1^2 \;\; \varphi_1\varphi_2 \;\; \varphi_1\varphi_3 \;\; \varphi_2^2 \;\; \varphi_2\varphi_3 \;\; \varphi_3^2]^T$$

The number of parameters n_{φ_p} of $\underline{\varphi_p}(t)$ may be very important if the size of $\underline{\varphi}(t)$ is important or if the order p is high.

For notation simplicity we will replace $V_{k,1}$ by V_k. Local models can then be expressed by the relation:

$$f_i\big(\underline{\varphi}(t), \theta_i\big) = \sum_{k=1}^{n_{\varphi p}} V_k \theta_{i_k} + \theta_{i_0} \qquad (10)$$

where:
θ_{i_k} ($k = 0 \cdots n_{\varphi_p}$ and $i = 1 \cdots M$) are real constants;
$\theta_i = [\theta_{i_0} \quad \theta_{i_1} \quad \cdots \quad \theta_{i_{n_{\varphi p}}}]^T$ parameters vector of the ith local model.
The main advantage of such a representation is that local models are nonlinear whereas they are linear with respect to parameters. This structure considerably simplifies parameter estimation (see §4). Equation (6) can be rewritten as:

$$\hat{y}(t+1) = \Phi(t)^T \, \theta \qquad (11)$$

where:
$\Phi(t) = \Big[\omega_1\big(\underline{\xi}(t)\big)\underline{\phi_e}(t)^T \cdots \omega_M\Big(\underline{\xi}(t)\Big)\underline{\phi_e}(t)^T\Big]^T$ is the global weighted regression vector;
$\underline{\phi_e}(t) = \big[\underline{\varphi_p}(t)^T \; 1\big]^T$ is the extended regression vector;
$\theta = \big[\theta_1^T \ldots \theta_i^T \ldots \theta_M^T\big]^T$ is a concatenation of all local models parameters vectors;

Estimating θ can be carried out by using a global learning criteria J which consists of minimizing the error between system's output and multi-model's output:

$$J = \frac{1}{2}\sum_{t=1}^{N}\left(y_s(t) - \hat{y}(t)\right)^2 = \sum_{t=1}^{N}\left[\varepsilon(t)\right]^2 \qquad (12)$$

For non-recurrent multi-models with polynomial local models, J is linear with respect to the multi-model's parameters vector. J is minimized analytically using Least-Squares method. Multi-model parameters are then computed using the expression:

$$\hat{\theta} = (\Phi_g^T \Phi_g)^{-1}(\Phi_g^T Y_s) \qquad (13)$$

where:
$\hat{\theta}$ is the estimation of θ;
$\Phi_g = \left[\Phi(t)\right]_{t=1}^{t=N}$ is global weighted regression matrix of all observations;
$Y_s = \left[y_s(t)\right]_{t=1}^{t=N}$ is the vector of output values of all observations;

For recurrent multi-models, parameters are estimated by a parametrical adaptation algorithm using at each time step the values of $\Phi(t)$, $y_s(t)$ and $\omega_i[\underline{\xi}(t)]$ as presented in the next section.

4 MULTI-MODEL'S IMPLEMENTATION OF RECURRENT MODELS

Parameter estimation in recurrent neural network models is carried out iteratively using gradient based algorithm. Convergence towards global minimum is not guaranteed and convergence rate might be high. As it will be stated here, for the proposed recurrent multi-model (RMM), parameters are estimated using recursive least squares. Hence, the criterion J in relation (12) is computed up to time step k according to:

$$J(k) = \frac{1}{2}\sum_{t=1}^{k}\left[\varepsilon(t)\right]^2 = \frac{1}{2}\sum_{t=1}^{k}\left(y_s(t) - \Phi^T(t-1)\,\theta_k\right)^2 \qquad (14)$$

with θ_k the value of θ evaluated up to time instant k. The minimization of this criterion leads to:

$$\theta_k = \Big[\sum_{t=1}^{k}\Phi(t-1)\,\Phi^T(t-1)\Big]^{-1}\sum_{t=1}^{k}y_s(t)\Phi(t-1) \qquad (15)$$

Relation (15) can be written in a recursive form. Assuming

$$A_k = \Big[\sum_{t=1}^{k}\Phi(t-1)\,\Phi^T(t-1)\Big]^{-1} \qquad (16)$$

then

$$\theta_k = A_k\sum_{t=1}^{k}y_s(t)\Phi(t-1) \qquad (17)$$

$$\theta_{k+1} = A_{k+1}\sum_{t=1}^{k+1}y_s(t)\Phi(t-1) \qquad (18)$$

The sum in the right hand side of (18) can be transformed after some manipulations to:

$$\sum_{t=1}^{k+1}y_s(t)\Phi(t-1) = A_{k+1}^{-1}\theta_k + \Phi(k)\,\widetilde{\varepsilon}(k+1) \qquad (19)$$

where:
$\widetilde{\varepsilon}(k+1) = y_s(k+1) - \Phi^T(k)\,\theta_k$ is the *a priori* prediction error (the error at time instant $k+1$ evaluated with parameters computed up to time instant k). Putting (19) in (18) leeds to a recursive expression of θ:

$$\theta_{k+1} = \theta_k + A_{k+1}\,\Phi(k)\,\widetilde{\varepsilon}(k+1) \qquad (20)$$

A_{k+1} can also be computed recursively. From (16) one can write :

$$[A_{k+1}]^{-1} = [A_k]^{-1} + \Phi(k)\,\Phi^T(k) \qquad (21)$$

Applying matrix inversion lemma to relation (21), A_{k+1} is computed recursively:

$$A_{k+1} = A_k - \frac{A_k\,\Phi(k)\,\Phi^T(k)\,A_k}{1+\Phi^T(k)\,A_k\,\Phi(k)} \qquad (22)$$

So, the parameters vector θ is updated recursively at each time step using relations (22) and (20). This learning algorithm is used for the identification of NARMAX and NOE structures based on the RMM architectures (see figures 2 and 3).

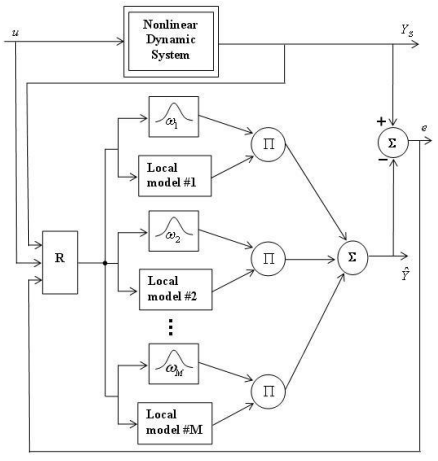


Figure 2: Recurrent multi-model implementation of a NARMAX model.

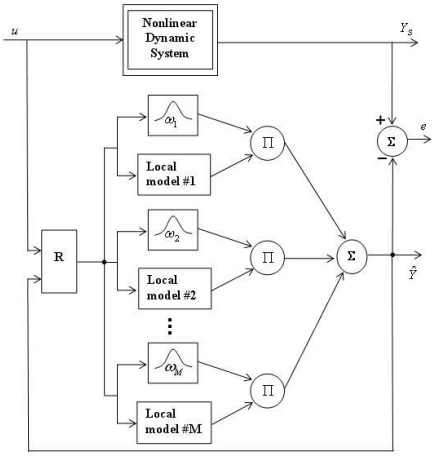


Figure 3: Recurrent multi-model implementation of a NOE model.

5 RESULTS AND DISCUSSION

To validate the proposed RMM architecture, two non linear systems are used. The first one is a simulated system which data are generated from a NARX model (Gasso, 2000). The second one is Box-Jenkins gas furnace benchmark (Box and Jenkins, 1970). For comparison purposes, we have implemented recurrent Multi-Layer Perceptron (MLP) with one hidden layer for NARMAX and NOE models, both trained with the Backpropagation Through Time (BPTT) algorithm (Werbos, 1990). To enhance the speed of learning with the BBTT algorithm, the learning rate is adapted so that it takes high values when the learning error decreases fastly and take small values when it decreases slowly.

Performances of recurrent multi-models with given order p of polynomial local models (RMM_p) are evaluated. Akaike Information Criterion (AIC) is used for model's parsimony estimation (least error with minimum parameters):

$$AIC = N \ln J + 2 n_\theta \tag{23}$$

where n_θ denotes the number of model's parameters. Root Mean Square Error criterion (RMSE) is also used for performance evaluation in learning ($RMSE_L$) and validation ($RMSE_V$) phases. The architecture of models (Arch) specifies the number of local models in multi-models case or the number of hidden neurons in MLP case. Computation time (CT) during which models parameters are determined is used for algorithms convergence speed evaluation.

5.1 Example 1: Narx Dynamic Process

The following system is simulated in a noisy context and then identified using recurrent multi-model and recurrent MLP.

$$y_s(t) = \frac{y_s(t-1)\left[0.5u_1(t-1)-0.3u_2(t-1)\right]}{1+y_s^2(t-1)} + 0.3u_1^2(t-1) - 0.5u_2^2(t-1) \tag{24}$$

Exogenous input signals $u_1(\cdot)$ and $u_2(\cdot)$ are chosen to be pulses of random magnitude (in interval $[0,1]$) and different widths; the output signal is then corrupted by a white noise e issued from a normal distribution. The signal to noise ratio equals 14 dB. The obtained noisy output y_{s_n} (see figure 4) is expressed by:

$$y_{s_n}(t) = y_s(t) + e(t) \tag{25}$$

NOE model class is the most suitable one to identify this kind of system (Dreyfus, 2002). Both NOE RMM and NOE MLP models are implemented. The following regression vector is used:

$$\underline{\varphi}(t) = [u(t-1)\ \hat{y}(t-1)]^T$$

where $\hat{y}$ is the estimated model output. The vector of indexing variables is:

$$\underline{\xi}(t) = \left[u_1(t)\ u_2(t)\right]^T$$

Table 1 shows obtained results for NOE RMMs and a NOE MLP structures. System and NOE RMM_1 outputs are plot on validation data in figure 5. The obtained results show that multi-model structures have performances equivalent to MLP structures. However, their computation time is much lower.

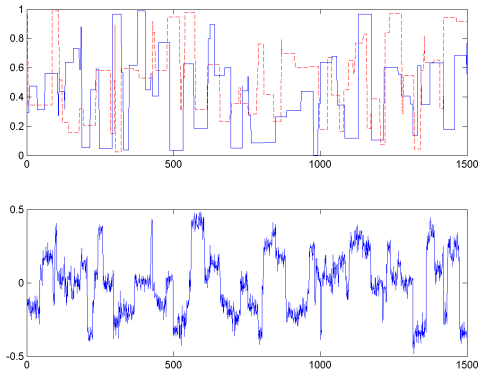


Figure 4: Inputs and output indentification data.

Table 1: NOE model for the nonlinear dynamic process: results for NOE RMM and NOE MLP structures.

Model	Arch.	AIC	CT(s)	$RMSE_L$	$RMSE_V$
RMM_1	7	-9595	8	0.040	0.010
RMM_2	3	-9617	6	0.040	0.007
MLP	3	-9587	199	0.040	0.006

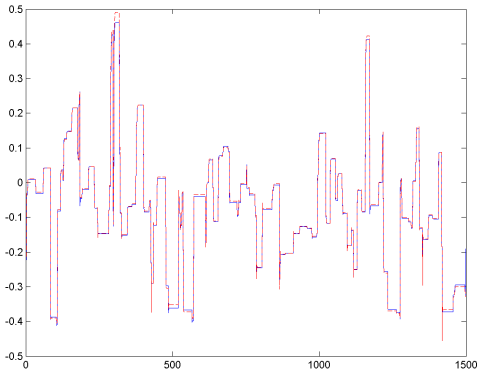


Figure 5: NOE RMM_1 output (dotted line) and noise free system output on validation data.

5.2 Example 2: Box-Jenkins Gas Furnace Benchmark

In this benchmark, data set are obtained from a combustion process of methane-air mixture. The process input is the methane gas flow into the furnace and the output is CO_2 concentration in the outlet gas (Box and Jenkins, 1970). System inputs and outputs are presented in figure 6. We have implemented and compared a NARMAX MLP and a NARMAX RMM structures based on the described methodologies. The following regression vector is used:

$$\begin{aligned}\underline{\varphi}(t) \quad = \quad & [u(t-1)\ u(t-2)\ u(t-3) \\ & y_s(t-1)\ y_s(t-2)\ y_s(t-3)\ e(t-1)]^T\end{aligned}$$

The vector of indexing variables is:

$$\underline{\xi}(t) = \left[u(t)\ y_s(t)\right]^T$$

The results are presented in table 2. The NARMAX RMM has the best parsimony and gives best performances on validation data, with a very low computation time compared to the NARMAX MLP. It can be seen that high polynomial orders reduces the number of local models. Figure 7 shows process and NARMAX RMM_1 outputs on validation data.

Table 2: NARMAX model for Box-Jenkins gas furnace data: results for Multi-model and MLP structures.

Model	Arch.	AIC	CT(s)	$RMSE_L$	$RMSE_V$
RMM_1	6	-692	3	0.12	0.55
RMM_2	2	-637	2	0.13	0.63
MLP	3	-622	35	0.17	0.58

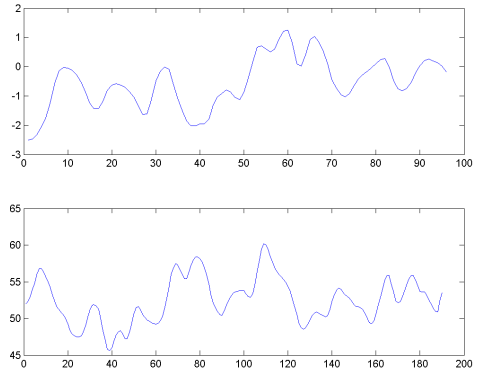


Figure 6: Process input and output on identification data.

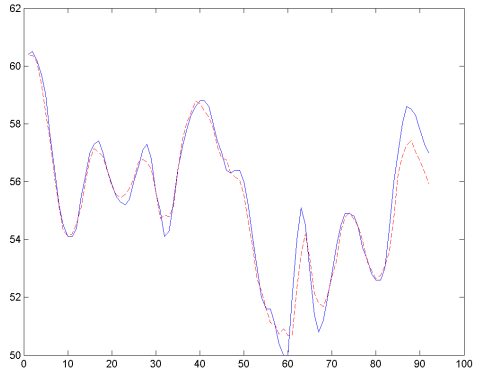


Figure 7: Process and NARMAX RMM_1 (dotted line) outputs on validation data.

6 CONCLUSION

In this work, a new recurrent multi-model structure with polynomial local models is proposed. The advantage of using polynomial local models is a better handling of local nonlinearities and reducing henceforth the number of local models. The proposed structure is used to implement NARMAX and NOE models.

Identification task is carried out very simply and obtained results show that the proposed recurrent multi-model has many advantages over recurrent

MLP model, among which the reduction of computation time. This is due to the way the parameters are estimated: least squares formula in the former model and iterative algorithm in the latter.

The perspective of this study is the implementation of the proposed structures for model predictive control in industrial processes.

REFERENCES

Bezdec, J. (1973). *Fuzzy mathematics in pattern classification*. PhD thesis, Applied Math. Center, Cornell University Ithaca.

Bielikova, M. (2005). Recurrent neural network training with the extended kalman filter. *IIT. SRC*, pages 57–64.

Boukhris, A., Mourot, G., and Ragot, J. (2000). Nonlinear dynamic system identification: a multiple-model approach. *Int. J. of control*, 72(7/8):591–604.

Box, G. and Jenkins, G. (1970). Time series analysis, forecasting and control. *San Francisco, Holden Day*, pages 532–533.

Cheng, Y., Karjala, T., and Himmelblau, D. (1997). Closed loop nonlinear process identification using internal recurrent nets. *Neural Networks*, 10(3):573–586.

Corradini, A. and Cohen, P. (2002). Multimodal speech-gesture interface for hands-free painting on virtual paper using partial recurrent neural networks for gesture recognition. *in Proc. of the Int'l Joint Conf. on Neural Networks (IJCNN'02)*, 3:2293–2298.

Dreyfus, G. (2002). *Rseaux de Neurones - Mthodologie et applications*. Eyrolles.

Elman, J. (1999). Finding structure in time. *Cognitive Science*, 14(2):179–211.

Gao, Y. and Foss, A. (2005). Narmax time series model prediction: feed forward and recurrent fuzzy neural network approaches. *Fuzzy Sets and Sytems*, 150:331–350.

Gasso, K. (2000). *Identification de systmes dynamiques non linaires: approche multi - modle*. thse de doctorat de l'INPL.

Huang, B., Rashid, T., and Kechadi, M.-T. (2005). A recurrent neural network recognizer for online recognition of handwritten symbols. *ICEIS*, 2:27–34.

Huang, B., Rashid, T., and Kechadi, M.-T. (2006). Multi-context recurrent neural network for time series applications. *International Journal of Computational Intelligence*, 3(1):45–54.

Johansen, T. and Er, M. (1992). Nonlinear local model representation for adaptive systems. In *Proc. of the IEEE Conf. on Intelligent Control and Instrumentation*, volume 2, pages 677–682, Singapore.

Johansen, T. and Er, M. (1993). Constructing narmax using armax. *Int. Journal of Control*, 58(5):1125–1153.

Jordan, M. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. In *Proceedings of IASTED International Conference of the Cognitive Science Society. (Reprinted in IEEE Tutorials Series, New York: IEEE Publishing Services, 1990)*, pages 531–546, Englewood Cliffs, NJ: Erlbaum.

Konur, U. and Okatan, A. (2004). Time series prediction using recurrent neural network architectures and time delay neural networks. *ENFORMATIKA*, pages 1305–1313.

Li, N., Li, S. Y., and Xi, Y. G. (2004). Multi-model predictive control based on the takagi-sugeno fuzzy models: a case study. *Information Sciences*, 165:247–263.

Murray-Smith, R. and Johansen, T. (1997). *Multiple Model Approaches to Modeling and Control*. Taylor and Francis Publishers.

Orjuela, R., Maquin, D., and Ragot, J. (2006). Identification des systmes non linaires par une approche multimodle tats dcoupls. *Journes Identification et Modlisation Exprimentale JIME'2006 - 16 et 17 novembre - Poitiers*.

Palma, F. D. and Magni, L. (2004). A multimodel structure for model predictive control. *Annual Reviews in Control*, 28:47–52.

Sjoberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P., Hjalmarsson, H., and Juditsky, A. (1995). Nonlinear black-box modeling in system identification: a unified overview. *Automatica 31*, 31(12):1691–1724.

Tomasz, J. and Jacek, M. (1997). Neural networks tool for stellar light prediction. In *Proc. of the IEEE Aerospace Conference*, volume 3, pages 415–422, Snowmass, Colorado, USA.

Vartak, A., Georgiopoulos, M., and Anagnostopoulos, G. (2005). On-line gauss-newton-based learning for fully recurrent neural networks. *Nonlinear Analysis*, 63:867–876.

Vernieuwe, H., Georgieva, O., Baets, B., Pauwels, V., Verhoest, N., and Troch, F. (2004). Comparison of data-driven takagi-sugeno models of rainfall-discharge dynamics. *Journal of Hydrology*, XX:1–14.

Werbos, P. (1990). Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560.

Yang, W. Z., L.H., Y., and L., C. (2005). Narmax model representation and its application to damage detection for multi-layer composites. *Composite Structures*, 68:109–117.

APPLICATION OF SPATIAL H_∞ CONTROL TECHNIQUE FOR ACTIVE VIBRATION CONTROL OF A SMART BEAM

Ömer Faruk Kircali
STM Savunma Teknolojileri Mühendislik ve Ticaret A.Ş., Ankara, Turkey
fkircali@stm.com.tr

Yavuz Yaman, Volkan Nalbantoğlu, Melin Şahin, Fatih Mutlu Karadal
Department of Aerospace Engineering, Middle East Technical University, Ankara, Turkey
yyaman@ae.metu.edu.tr, volkan@ae.metu.edu.tr, msahin@ae.metu.edu.tr, karadal@ae.metu.edu.tr

Keywords: Assumed-Modes, Model Correction, Smart Beam, Spatial H_∞ Controller Design.

Abstract: This study presents the design and implementation of a spatial H_∞ controller for the active vibration control of a cantilevered smart beam. The smart beam consists of a passive aluminum beam (507x51x2mm) and eight symmetrically surface bonded SensorTech BM500 type PZT (Lead-Zirconate-Titanate) patches (25x20x0.5mm). PZT patches are used as actuators and a laser displacement sensor is used as sensor. The smart beam was analytically modelled by using the assumed-modes method. The model only included the first two flexural vibrational modes and the model correction technique was applied to compensate the possible error due to the higher order modes. The system model was also experimentally identified and both theoretical and experimental models were used together in order to determine the modal damping ratios of the smart beam. A spatial controller was designed for the suppression of the vibrations of the smart beam due to its first two flexural modes. The designed controller was then implemented to experimentally suppress the vibrations. This study also compared the effectiveness of a pointwise controller with the newly developed spatial one.

1 INTRODUCTION

The vibration is an important phenomenon for the lightweight flexible aerospace structures. Those structures may be damaged under any undesired vibrational load. Hence, they require a proper control mechanism to attenuate the vibration levels in order to preserve the structural consistency. The usage of smart materials, as actuators and/or sensors, has become promising research and application area that gives the opportunity to accomplish the reduction of vibration of flexible structures and proves to be an effective active control mechanism.

The smart structure is a structure that can sense external disturbance and respond to that with active control in real time to maintain mission requirements (Çalışkan, 2002). Active vibration control of a smart structure requires an accurate system model of the structure. Smart structures can be modeled by using analytical methods or system identification techniques using the experimental data (Meirovitch, 1986 and Nalbantoğlu, 1998). The system model of a smart structure generally involves a large number of vibrational modes. However, the performance goals are mostly related to the first few vibrational modes since their effect on structural failure is much more prominent. Hence, a reduction of the order of the model is required (Hughes, 1981 and Moheimani, 1997). On the other hand, ignoring the higher modes can affect the system behaviour since directly removing the higher modes from the system model perturbs the zeros of the system. Therefore, in order to minimize the model reduction error, a correction term, including some of the removed modes, should be added to the model (Clark, 1997).

Today, robust stabilizing controllers designed in respect of H_∞ control technique are widely used on active vibration control of smart structures. Yaman et al. (2001 and 2003) showed the effect of H_∞ controller on suppressing the vibrations of a smart beam due its first two flexural modes. Similar work is done for active vibration control of a smart plate,

and the effective usage of piezoelectric actuators on vibration suppression with H_∞ controller was successfully presented (Yaman, 2002).

Whichever controller design technique is applied, the suppression should be preferred to be achieved over the entire structure rather than at specific points, since the flexible structures are usually those of distributed parameter systems. Moheimani and Fu (1998) and Moheimani et al. (1997) introduced spatial H_2 and H_∞ norm concepts in order to meet the need of spatial vibration control, and simulation-based results of spatial vibration control of a cantilevered beam were presented. Moheimani et al. (1999) studied spatial feedforward and feedback controller design, and presented illustrative results. They also showed that spatial H_∞ controllers could be obtained from standard H_∞ controller design techniques. Halim (2002) studied the implementation of spatial H_∞ controller on active vibration control and presented quite successful results. However his works were limited to a beam with simply supported boundary conditions.

This paper aims to present design and implementation of a spatial H_∞ controller on active vibration control of a cantilevered smart beam.

2 THE SMART BEAM MODEL

The cantilevered smart beam model and its structural properties are given in Figure 1 and Table 1, respectively. The smart beam consists of a passive aluminum beam (507mmx51mmx2mm) with symmetrically surface bonded eight SensorTech BM500 type PZT (Lead-Zirconate-Titanate) patches (25mmx20mmx0.5mm). The beginning and end locations of the PZT patches along the length of the beam are denoted as r_1 and r_2, respectively. The patches are assumed to be optimally placed by considering maximum strain characteristics (Çalışkan, 2002). The parameters L, w, t, ρ, E, A, I, d_{31} denote length, width, thickness, density, Young's modulus, cross-sectional area, second moment of area and piezoelectric charge constant; and the subscripts b and p indicate the beam and PZT patches, respectively. Note that, despite the actual length of the beam is 507mm, the effective length utilized in the study (i.e. the effective span of the beam) reduces to 494mm since it is clamped with a fixture.

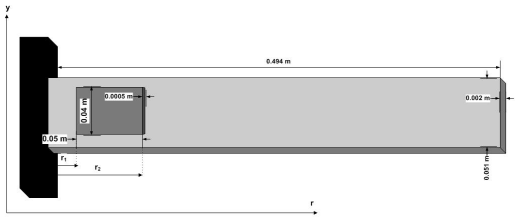


Figure 1: The smart beam model used in the study.

Table 1: The properties of the smart beam.

Aluminum Passive Beam	PZT
$L_b = 0.494m$	$L_p = 0.05m$
$w_b = 0.051m$	$w_p = 0.04m$
$t_b = 0.002m$	$t_p = 0.0005m$
$\rho_b = 2710kg/m^3$	$\rho_p = 7650kg/m^3$
$E_b = 69GPa$	$E_p = 64.52GPa$
$A_b = 1.02x10^{-4}m^2$	$A_p = 0.2x10^{-4}m^2$
$I_b = 3.4x10^{-11}m^4$	$I_p = 6.33x10^{-11}m^4$
-	$d_{31} = -175x10^{-12}m/V$

The assumed-modes model of the smart beam includes large number of resonant modes (Kırcalı, 2005). However, the control design criterion of this study is to suppress only the first two flexural modes of the smart beam. Hence, that higher order model is directly truncated to a lower order one, including only the first two flexural modes. The direct model truncation may cause the zeros of the system to perturb, which consequently affect the closed-loop performance and stability of the system considered (Clark, 1997). For this reason, a general correction term k_i^{opt} is added to the truncated model and the resultant model (Kırcalı, 2005 and 2006) can be expressed as:

$$\bar{G}_C(s,r) = \sum_{i=1}^{2} \frac{\bar{P}_i \phi_i(r)}{s^2 + 2\xi_i \omega_i s + \omega_i^2} + \sum_{i=3}^{50} \phi_i(r) k_i^{opt} \qquad (1)$$

where general correction constant is [18]:

$$k_i^{opt} = \frac{1}{4\omega_c\omega_i}\frac{1}{\sqrt{1-\xi_i^2}}\ln\left\{\frac{\omega_c^2+2\omega_c\omega_i\sqrt{1-\xi_i^2}+\omega_i^2}{\omega_c^2-2\omega_c\omega_i\sqrt{1-\xi_i^2}+\omega_i^2}\right\}\bar{P}_i \quad (2)$$

and

$$\bar{P}_i = \frac{C_p\left[\phi_i'(r_2)-\phi_i'(r_1)\right]}{\rho_b A_b L_b^3 + 2\rho_p A_p L_p^3} \quad (3)$$

The nominal system model of the smart beam is denoted by $\bar{G}_C(s,r)$. The geometric constant $C_p = E_p d_{31} w_p (t_p + t_b)$ is due to bending moment of PZT patches exerted on the beam. The parameter *r* defines the spatial variation along the longitudinal axis and *t* is the time. The cut-off frequency of the correction term is denoted by ω_c and the details of all the parameters and the detailed derivation of the equation (1) can be found in reference (Kırcalı, 2006).

Theoretical assumed-modes modeling does not provide any information about the damping of the system. Experimental system identification, on the other hand, when used in collaboration with the analytical model, helps one to obtain more accurate spatial characteristics of the structure. The modal damping ratios and more accurate resonance frequencies were determined by spatial system identification (Kırcalı, 2006) and the results are given in Table-2:

Table 2: The resonance frequencies and modal damping ratios of the smart beam.

ω_1 (Hz)	ω_2 (Hz)	ξ_1	ξ_2
6.742	41.308	0.027	0.008

3 SPATIAL H_∞ CONTROL OF THE SMART BEAM

3.1 Controller Design

Consider the closed loop system of the smart beam shown in Figure 2. The aim of the controller, *K*, is to reduce the effect of disturbance signal over the entire beam by the help of the PZT actuators.

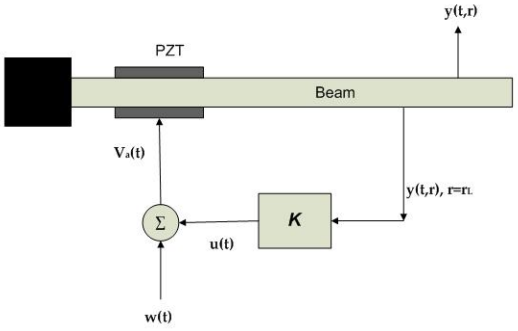


Figure 2: The closed loop system of the smart beam.

The state space representation of the system above can be shown to be (Kırcalı, 2006):

$$\begin{aligned}\dot{x}(t) &= Ax(t) + B_1 w(t) + B_2 u(t)\\ y(t,r) &= C_1(r)x(t) + D_1(r)w(t) + D_2(r)u(t)\\ y(t,r_L) &= C_2 x(t) + D_3 w(t) + D_4 u(t)\end{aligned} \quad (4)$$

where *x* is the state vector, *w* is the disturbance input, *u* is the control input, $y(t,r)$ is the performance output, $\tilde{y}(t,r_L)$ is the measured output at location $r_L = 0.99L_b$. The performance output represents the displacement of the smart beam along its entire body, and the measured output represents the displacement of the smart beam at a specific location *A* is the state matrix, B_1 and B_2 are the input matrices from disturbance and control actuators respectively, Π is the output matrix of error signals, C_2 is the output matrix of sensor signals, Θ_1, Θ_2, D_3 and D_4 are the correction terms from disturbance actuator to error signal, control actuator to error signal, disturbance actuator to feedback sensor and control actuator to feedback sensor respectively. The disturbance $w(t)$ is accepted to enter to the system through the actuator channels, hence, $B_1 = B_2$, $D_1(r) = D_2(r)$ and $D_3 = D_4$.

The state space form of the controller can be represented as:

$$\begin{aligned}\dot{x}_k(t) &= A_k x_k(t) + B_k y(t,r_L)\\ u(t) &= C_k x_k(t) + D_k y(t,r_L)\end{aligned} \quad (5)$$

such that the closed loop system satisfies:

$$\inf_{K\in U} \sup_{w\in L_2[0,\infty)} J_\infty < \gamma^2 \quad (6)$$

where U is the set of all stabilizing controllers and γ is a constant.

The spatial cost function to be minimized as the design criterion is:

$$J_\infty = \frac{\int_0^\infty \int_R y(t,r)^T Q(r) y(t,r) dr dt}{\int_0^\infty w(t)^T w(t) dt} \quad (7)$$

where $Q(r)$ is a spatial weighting function that designates the region over which the effect of the disturbance is to be reduced and J_∞ can be considered as the ratio of the spatial energy of the system output to that of the disturbance signal. The control problem is depicted in Figure 3.

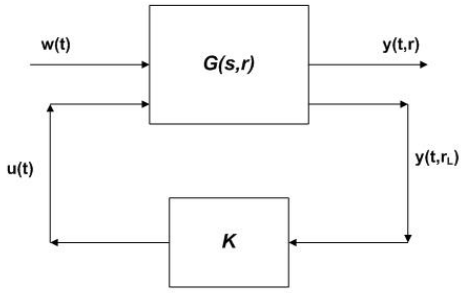


Figure 3: The spatial H_∞ control problem of the smart beam.

The spatial H_∞ control problem can be solved by the equivalent ordinary H_∞ problem (Moheimani et.al, 2003) by taking:

$$\int_0^\infty \int_R y(t,r)^T Q(r) y(t,r) dr dt = \int_0^\infty \tilde{y}(t)^T \tilde{y}(t) dt \quad (8)$$

Hence, following the necessary mathematical manipulations, the adapted state space representation will be:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 w(t) + B_2 u(t) \\ \tilde{y}(t) &= \begin{bmatrix} \Pi \\ 0 \end{bmatrix} x(t) + \begin{bmatrix} \Theta_1 \\ 0 \end{bmatrix} w(t) + \begin{bmatrix} \Theta_2 \\ \kappa \end{bmatrix} u(t) \\ y(t, r_L) &= C_2 x(t) + D_3 w(t) + D_4 u(t) \end{aligned} \quad (9)$$

The derivation of equation (9) and the below state space variables can be found in (Kırcalı, 2006) as:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\omega_1^2 & 0 & -2\xi_1\omega_1 & 0 \\ 0 & -\omega_2^2 & 0 & -2\xi_2\omega_2 \end{bmatrix} \quad (10)$$

$$B_1 = B_2 = \begin{bmatrix} 0 \\ 0 \\ \overline{P}_1 \\ \overline{P}_2 \end{bmatrix} \quad (11)$$

$$C_1 = \begin{bmatrix} \phi_1(r) \\ \phi_2(r) \\ 0 \\ 0 \end{bmatrix}^T, C_2 = \begin{bmatrix} \phi_1(r_L) \\ \phi_2(r_L) \\ 0 \\ 0 \end{bmatrix}^T \quad (12)$$

$$\begin{aligned} D_1 = D_2 &= \sum_{i=3}^{50} \phi_i(r) k_i^{opt} \\ D_3 = D_4 &= \sum_{i=3}^{50} \phi_i(r_L) k_i^{opt} \end{aligned} \quad (13)$$

$$\Pi = \begin{bmatrix} diag(L_b^{3/2})_{2x2} & 0_{2x2} \\ 0_{3x2} & 0_{3x2} \end{bmatrix} \quad (14)$$

$$\Theta_1 = \Theta_2 = \begin{bmatrix} 0_{4x1} \\ \left(\sum_{i=3}^{50} L_b^3 \left(k_i^{opt} \right)^2 \right)^{1/2} \end{bmatrix} \quad (15)$$

One should note that, the control weight, κ, is added to the system in order to limit the controller gain and avoid actuator saturation problem. In the absence of the control weight, the major problem of designing an H_∞ controller for the system given in equation (4) is that, such a design will result in a controller with an infinitely large gain (Moheimani et.al, 1999). In order to overcome this problem, an appropriate control weight, which is determined by the designer, should be added to the system. Since the smaller κ will result in higher vibration suppression but larger controller gain, it should be determined optimally such that not only the gain of the controller does not cause implementation difficulties but also the suppressions of the vibration levels are satisfactory. In this study, κ was decided to be taken as 7.87×10^{-7}. The simulation of the effect

of the controller is shown in Figure 4 as a Bode plot, and the frequency domain simulation is done by Matlab v6.5.

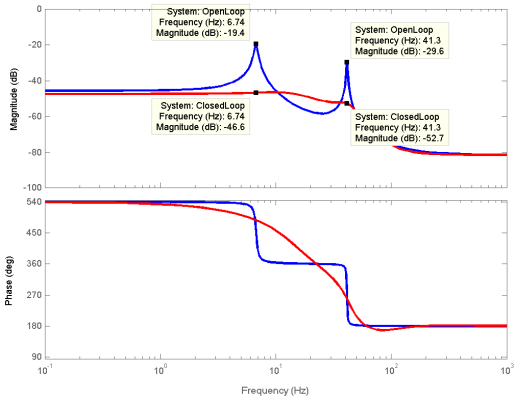


Figure 4: Bode plots of the open and closed loop frequency responses of the smart beam.

The vibration attenuation levels at the first two flexural resonance frequencies were found to be 27.2 dB and 23.1 dB, respectively. The simulated results show that the designed controller is effective on the suppression of excessive vibrational levels.

3.2 Experimental Implementation

The smart beam of this study, shown in Figure 5, consists of the PZT patches that are placed in a collocated manner to have opposite polarity and used as the actuators. A Keyence LB-1201(W) LB-300 laser displacement sensor (LDS) is used as the sensor. The closed loop experimental setup is shown in Figure 6.

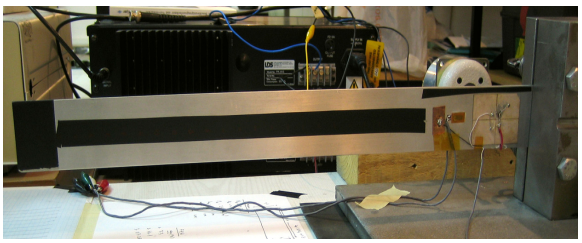


Figure 5: The smart beam used in the study.

The displacement of the smart beam at location $r_L = 0.99L_b$ was measured by using the LDS and converted to a voltage output that was sent to the SensorTech SS10 controller unit via the connector block. The controller output was converted to the analog signal and amplified 30 times by SensorTech SA10 high voltage power amplifier before applied to the piezoelectric patches. The controller unit is hosted by a Linux machine, on which a shared disk drive is present to store the input/output data and the C programming language based executable code that is used for real-time signal processing.

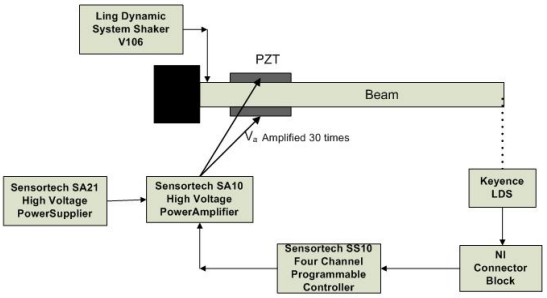


Figure 6: The closed loop experimental setup.

3.2.1 Free Vibration Suppression

For the free vibration control, the smart beam was given an initial 5 cm tip deflection and the open loop and closed loop time responses of the smart beam were measured. The results are presented in Figure 7. Figure 7 shows that the controlled time response of the smart beam settles nearly in 1.7 seconds. Hence, the designed controller proves to be very effective on suppressing the free vibration of the smart beam.

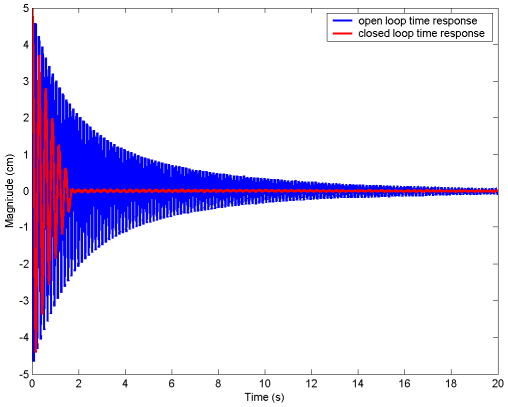
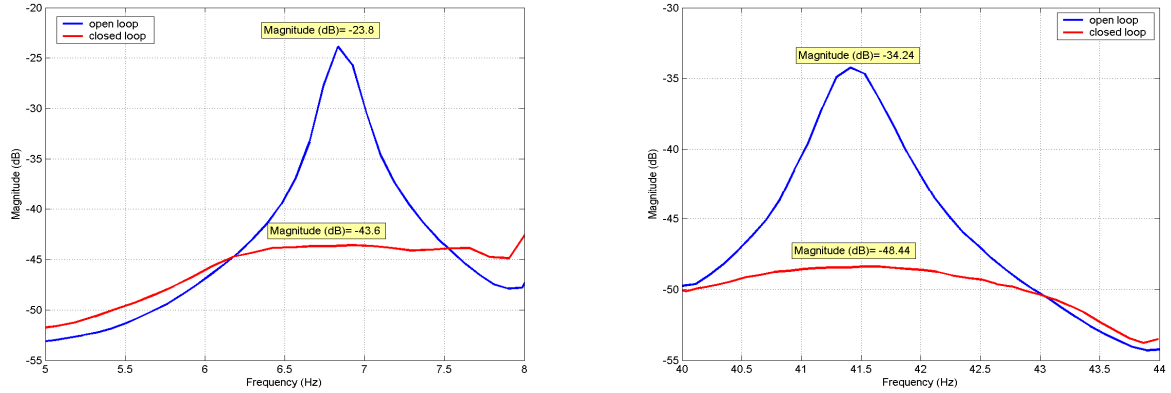
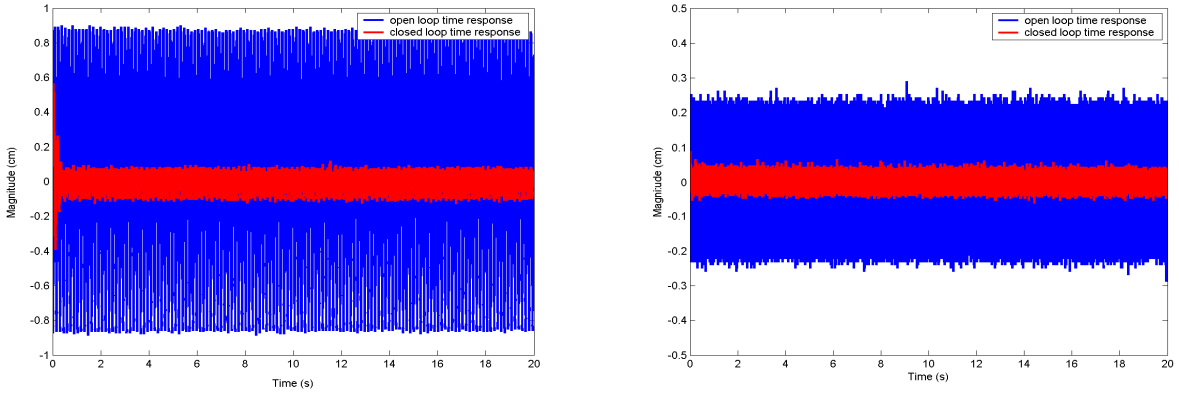


Figure 7: Free vibration suppression of the smart beam.



a) 5-8 Hz b) 40-44 Hz

Figure 8: Bode magnitude plot of the open and closed loop systems.



a) 1st resonance b) 2nd resonance

Figure 9: Open and closed loop time responses of the smart beam under constant excitation at resonance frequencies.

3.2.2 Forced Vibration Suppression

The forced vibration control of the smart beam was analyzed in two different configurations. In the first one, the smart beam was excited for 180 seconds with a shaker located very close to the root of the smart beam, on which a sinusoidal chirp signal of amplitude 4.5V was applied. The excitation bandwidth was taken first 5 to 8 Hz and later 40 to 44 Hz to include the first two flexural resonance frequencies separately. The experimental attenuation of vibration levels were determined from the Bode magnitude plots shown in Figure 8.a-b. The resultant attenuation levels were found as 19.8 dB and 14.2 dB, respectively. In the second configuration, instead of using a sinusoidal chirp signal, a constant excitation was applied for 20 seconds at the resonance frequencies again with a shaker. The ratios of the maximum time responses of the open and closed loop systems, shown in Figure 9.a-b, are considered as absolute attenuation levels. Hence, for this case, the attenuation levels at each resonance frequency were calculated approximately as 10.4 and 4.17, respectively. Consequently, the experimental results show that the controller is effective on suppression of the forced vibration levels of the smart beam.

3.3 Efficiency of the Controller

The efficiency of spatial controller in minimizing the overall vibration over the smart beam was compared by a pointwise controller that is designed to minimize the vibrations only at point $r_L = 0.99L_b$. For a more detailed description of the pointwise controller design, the interested reader may refer to the reference (Kırcalı, 2006). The implementations of the controllers showed that both controllers reduced the vibration levels of the smart beam due to

its first two flexural modes in comparable efficiency (Kırcalı, 2006). On the other hand, the simulated H_∞ norms of the smart beam as a function of *r*, shown in Figure 10, showed that the spatial H_∞ controller has a slight superiority on suppressing the vibration levels over entire beam.

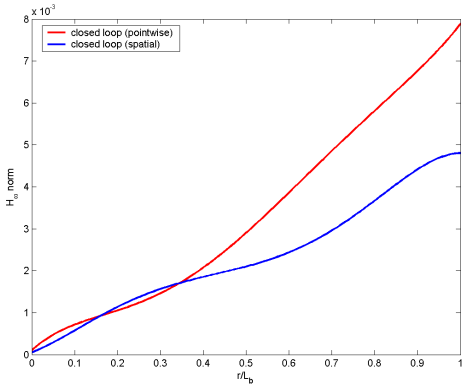


Figure 10: Simulated H_∞ norm plots of closed loop systems under the effect of controllers.

4 CONCLUSION

This study presented the active vibration control of a cantilevered smart beam. A spatial H_∞ controller was designed for suppressing the first two flexural vibrations of the smart beam. The efficiency of the controller was demonstrated both by simulation and experimental implementations. The effectiveness of the spatial controller on suppressing the vibrations of the smart beam over its entire body was also compared with a pointwise controller.

REFERENCES

Çalışkan T., 2002. Smart Materials and Their Applications in Aerospace Structures. *Ph.D. Thesis.* Middle East Technical University, Ankara, Turkey.

Meirovitch L., 1986. *Elements of Vibration Analysis.* The McGraw-Hill Company.

Nalbantoğlu V., 1998. Robust Control and System Identification for Flexible Structures. *Ph.D. Thesis*, University of Minnesota, USA.

Hughes P.C., Skelton R.E., 1981. Modal Truncation for Flexible Spacecraft, *Journal of Guidance and Control*, vol.4, no.3.

Moheimani S.O.R., Pota H.R., Petersen I.R., 1997. Spatial Balanced Model Reduction for Flexible Structures. *Proceedings of the American Control Conference,* 3098-3102. Albuquerque, New Mexico.

Clark R.L., 1997. Accounting for Out-Of-Bandwidth Modes in the Assumed Modes Approach: Implications on Colocated Output Feedback Control. *Transactions of the ASME, Journal of Dynamic Systems, Measurement, and Control*, vol.119, 390-395.

Yaman Y., Çalışkan T., Nalbantoğlu V., Prasad E., Waechter D., Yan B., 2001. Active Vibration Control of a Smart Beam, *Canada-US CanSmart Workshop on Smart Materials and Structures*. 137-147, Montreal, Canada.

Yaman Y., Ülker F. D., Nalbantoğlu V., Çalışkan T., Prasad E., Waechter D., Yan B., 2003. Application of H_∞ Active Vibration Control Strategy in Smart Structures. *3rd International Conference on Advanced Engineering Design*. Paper A5.3, Prague, Czech Republic.

Yaman Y., Çalışkan T., Nalbantoğlu V., Ülker F. D., Prasad E., Waechter D., Yan B., 2002. Active Vibration Control of Smart Plates by Using Piezoelectric Actuators, *6th Biennial Conference on Engineering Systems Design and Analysis*, Paper APM-018. Istanbul, Turkey.

Moheimani S.O.R, Fu M., 1998. Spatial H_2 Norm of Flexible Structures and its Application in Model Order Selection. *International Proceedings of 37th IEEE Conference on Decision and Control*, Tampa Florida, USA.

Moheimani S.O.R., Pota H.R., Petersen I.R., 1997. Spatial Balanced Model Reduction for Flexible Structures, *Proceedings of the American Control Conference*, pp. 3098-3102, Albuquerque, New Mexico.

Moheimani S.O.R., Petersen I.R., Pota H.R., 1999. Broadband Disturbance Attenuation over an Entire Beam, *Journal of Sound and Vibration*, 227(4): 807-832.

Halim D., Moheimani S.O.R., 2002. Experimental Implementation of Spatial H_∞ Control on a Piezoelectric Laminate Beam. *IEEE/ASME Transactions on Mechatronics*, vol.7, no: 3.

Kırcalı Ö.F., Yaman Y., Nalbantoğlu V., Şahin M., Karadal F.M., 2005. Spatial System Identification of a Smart Beam by Assumed-Modes Method and Model Correction. *Kayseri VI Aeronautics Symposium.* Nevsehir, Turkey (in Turkish).

Halim D., 2002. Vibration Analysis and Control of Smart Structures, *PhD. Thesis,*. School of Electrical Engineering and Computer Science, University of Newcastle, Australia.

Kırcalı Ö.F., 2006. Active Vibration Control of a Smart Beam: a Spatial Approach, *M.S. Thesis*, Middle East Technical University, Ankara, Turkey.

Moheimani S.O.R., Halim D., Fleming A.J., 2003. *Spatial Control of Vibration. Theory and Experiments*, World Scientific Publishing Co. Pte. Ltd.

A COMPONENT-BASED APPROACH FOR CONVEYING SYSTEMS CONTROL DESIGN

Jean-Louis Lallican * **, Pascal Berruet *, André Rossi * and Jean-Luc Philippe *
** LESTER, Unniversité de Bretagne Sud, Rue de Saint Maude – BP 92116, 56321 Lorient, France*
pascal.berruet@univ-ubs.fr , andre.rossi@univ-ubs.fr , jean-luc.philippe@univ-ubs.fr
*** SYDEL, Rue du Gaillec – Z.I. de Keryado – BP 2834, 56321 Lorient, France*
jean-louis.lallican@sydel.fr

Keywords: Conveying systems, Control engineering, Component-based approach, Generator, Model Driven Enginnering.

Abstract: This paper deals with the design of discrete control for conveying systems. A component-based approach is introduced to model controlled conveying systems. A component is a reusable element that includes several views including partial models. It is formalized referring to the notion of operations. Four views are delineated in this paper: Operating part view, Constraints view, Graphical view and Control view. Based on such a model, a methodology allowing to automatically generate the control programs is proposed to provide an easy way to obtain source code compatible with the IEC 61131-3 standard. Its purpose is to automate the development of control programs in order to reduce costs. Tools allowing to implement the methodology are also presented, along with some applications.

1 INTRODUCTION

Conveying systems are a part of manufacturing systems that transport parcels from some locations to new ones at a high flow. They are composed of different types of conveyors, elevators, consignments, sorters, and automated guided vehicles. Conveyors can be linear, curved, and circular. They can have pneumatic jacks, stops, and sensors.

Designers of such manufacturing systems are confronted to many problems. The complexity requires modular approaches, leading to split very large and complex design problems into simpler ones. It is necessary to reach the best approximations between functional solutions and material architecture at the earliest stage of design. Competition leads to decrease design and implementation times. Nevertheless a conveying system has to be robust, easy to maintain, easy to control, flexible, modular and fault tolerant. Meeting these requirements raises the need for a methodology and CAD tools.

The objective of this paper is to introduce a component-based approach for the design of discrete control to drive conveying systems. Components facilitate the models computation used to generate control programs automatically. Firstly in the context of conveying system design, the objective is to reduce the time required to create the control. Secondly in the context of the reconfiguration (Berruet et al., 2005), it is necessary to provide several versions of the control. In this case the goal is to facilitate the creation of these controls.

The proposed methodology for generating control programs is based on a MDE (Model Driven Engineering) approach, in which models are described using meta-models at each step of the process. Transformation between these models are expressed using a transformation language such as ATL (Atlas Transformation Language) (Bezivin, 2005).

The present work has been developed with Sydel society, located in Lorient (France) and specialized in integration of conveying systems.

This paper is organized as follows. A global design flow for conveying systems is defined in section 2. The component approach is presented in section 3. This paper deals only with operating part, constraints, graphical and control views of the components. Section 4 describes how components are used to generate control programs and the first experimental results are presented in section 5.

2 DESIGN PROCESS

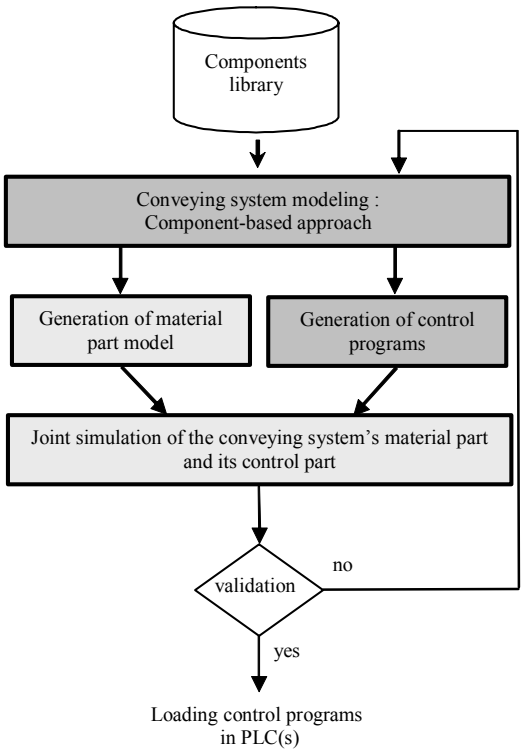


Figure 1: Global design process.

The global process is part of an usual flow based on a simulation to validate or modify the design parameters. It integrates a component-based approach making it possible to facilitate design process. Simulation concerns operating and control parts, the control program being associated with the operating part. A tool named SimSED (Lallican et al., 2005) has been developed to support the simulation. The objective of this process is to design, to validate and to implement control of conveying systems.

The procedure described in Figure 1 involves four steps: system modeling, generation of material part model, generation of control programs and simulation. The system model is built by using a components library. After validation, control programs can be loaded in PLC(s) (Programmable Logic Controller). If simulation does not correspond to the specifications, the system model is modified.

The continuation of the paper presents the two main steps emphasized in Figure 1. The first step consist of building a model using a component-based approach. It refers to operation and component notions that are delineated in section 3. Based on such a model, the aim of the control programs generation step is to produce source code that may be distributed for control.

3 COMPONENT-BASED APPROACH

This section introduces a component-based approach to model conveying systems. It provides a clear and easy way to reuse previously modeled elements or to modify the system's internal structure. The complete workshop model is obtained by successively aggregating components until having one representing the whole system. If the study is based on an existing system, the first step consists in a structural splitting up in order to get components (Coudert et al., 2002).

3.1 Definitions: Operations and Components

As components (Definition 2) refer to operations, these last are first delineated.

Definition 1: An operation is a function performed by a resource of the conveying system.

This concept is a specialization of the function concept for conveying systems. In this kind of systems, operations belong to different categories (e.g. transfer or stocking operations). Operations are defined for any resources whereas functions are defined for the complete system (i.e. the conveying system). A resource can perform several operations and operations implement the resource functionalities.

Based on the typology applied to generic functions (functions of a system item defined with no reference to the behavior of the system) and contextual functions (adaptation or composition of generic functions given by constituents, in response to the requirements of the modeled system) (Toguyeni et al., 2003), three different types of operations are defined.

A **basic operation** is a generic function performed by a basic constituent. Advance by a jack, detection by a sensor are examples of basic operations.

A **contextual operation** is a contextual function performed by a resource. Detection of a jack position by a sensor is a contextual operation because the sensor is associated with the jack. A contextual operation is issued from one or several basic operations. Two types of contextual operations

have been defined (Require Position and Detect Position).

An **effective contextual operation** is a contextual function performed on a product by a resource. Transfer from area 1 to area 2 by a jack on a conveyor is an effective contextual operation. Three types of effective contextual operations have been defined: Transfer, Stocking and Detect Area.

The typology of operations is represented by a class diagram (see figure 2).

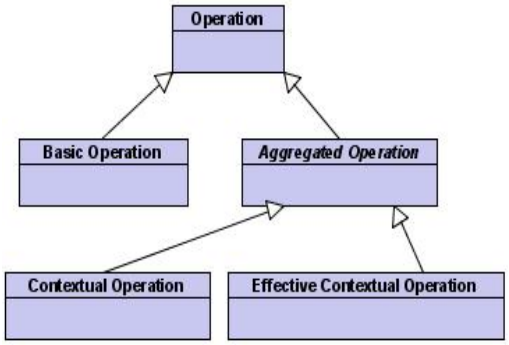


Figure 2: Typology of operations.

Definition 2: A component is a set of operations including monitoring, supervision and control point of views. Besides functions, it takes into account the system structure and its physical organization.

Components types are defined by analogy to operations types (figure 3).

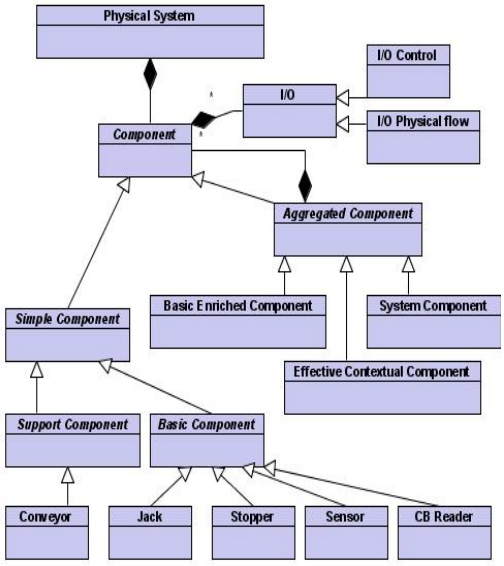


Figure 3: Typology of components.

A **basic component** is a set of basic operations supplied by the same constituent. Examples of basic components available in the library, are a stopper, a jack or a sensor. The set of basic operations can be enriched with contextual but non effective operations. This leads to **basic enriched component**. As an example of basic component, jack component gathers 2 basic operations: advance and retreat by the jack. These operations are performed by the same constituent. When sensors (end of course) are associated, four contextual operations are added. They are not performed by the same constituent. The component that contains these four operations is a basic enriched component.

The only function of a **support component** is to support. A support component can support parcels. It enables to define an area of admissible evolutions for parts (parcels, products). This area can be straight or curved for a conveyor. A belt conveyor is viewed as a support component.

An **effective contextual component** is a set of effective contextual operations put together, according with the part flow. It results in general from the association of basic components with a support component referring to parcels. For example, a jack component and a motor component associated with a conveyor component enable to define an ejector component. The ejector component has two operations: transfer from area 1 to area 2 by motorized conveyor, and transfer from area 1 to area 3 by the jack and the conveyor.

It has to be noticed that a support component is a sufficient condition for defining an effective contextual component.

A **system component** models the whole system (only one system component in the model of a conveying system exists) and refers to at least one effective contextual component.

The component description uses a black-box formalism. Inputs and Outputs relating to physical flow (connected with variables corresponding to parcels' passing) are separated from Inputs and Outputs dedicated to control (figure 3).

Basic and support components include parameters providing adaptability to different designs. They are stored in a library as validated ready-to-use models. An aggregation procedure has been developed. It consists in building a component of level L from several components of level L-1 brought together. Contextual components represent the first level of aggregation. Then it is possible to define several levels of aggregation with effective contextual components. Finally, the system component is the last level of aggregation (the whole

system). As components have the same structure at any abstraction level, the aggregated components can easily be stored and reused for future workshops design.

3.2 Components Views

A component is composed of four views (figure 4).

The **Operating part** view models the physical behavior of the modeled entity, including both discrete evolutions of the component and physical laws (linear or not), in order to represent mechanical, pneumatic and/or hydraulic phenomena. This view is conjointly simulated with the control part view to validate the behavior of the controlled system (see figure 1).

The **Graphical view** models characteristic areas of aggregated components. For a basic enriched component, characteristic areas correspond to some specific positions used for the description of contextual operations. For example, a jack associated with two sensors defines two positions: beginning and end of course. For effective contextual components, characteristic areas correspond to areas defined by effective contextual operations. For example, transfer operations refer to a source zone and a destination zone.

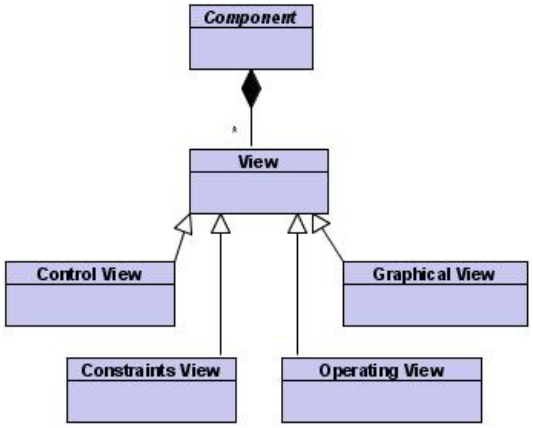


Figure 4: A components and its views.

The **Constraints view** expresses the conditions for beginning and for stopping effective contextual operations (only transfer or stocking operations). For example, a transfer operation can be activated when a parcel is detected in its source area, and can be stopped when the parcel is detected in its destination area.

The **Control view** expresses the discrete control of the modeled entity. This view is to be implemented by controllers.

Control part is described by using of sequential function charts (SFC) (IEC 61131-3, 2003). SFC has the advantage of manipulating simple concepts which are comely used by PLC program developers. Based on the component approach, the control is a hierarchical one (figure 5).

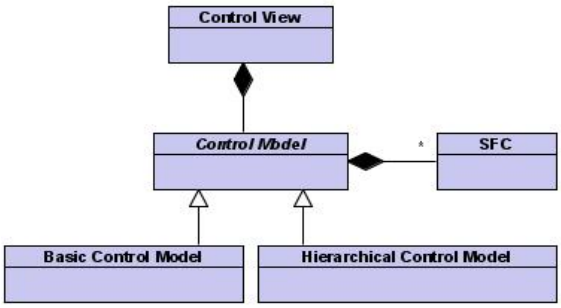


Figure 5: Control view.

When components of level *L* are selected for aggregation, the co-ordination of the different control parts, named hierarchical control part, has to be generated for the level *L*+1 component. The hierarchical control part, the goal of which is to coordinate the execution of low level SFCs is also described by means of SFCs. A high level SFC can request a low level control part to start or to stop a treatment. A lower level control part provides information to a higher level SFC.

Each operation involves a SFC.

As previously described, the control structure is a hierarchical one (figure 5). Two kinds of control part are considered: low level control part (basic control model) and hierarchical control part. Basic and support components which are stored in a library include low level control part. A hierarchical control part refers to an aggregated component (basic enriched component, effective contextual component and system component).

4 CONTROL DESIGN

4.1 Methodology

A methodology allowing to generate automatically the hierarchical control parts is presented in this section. Tools used to implement this methodology are also introduced.

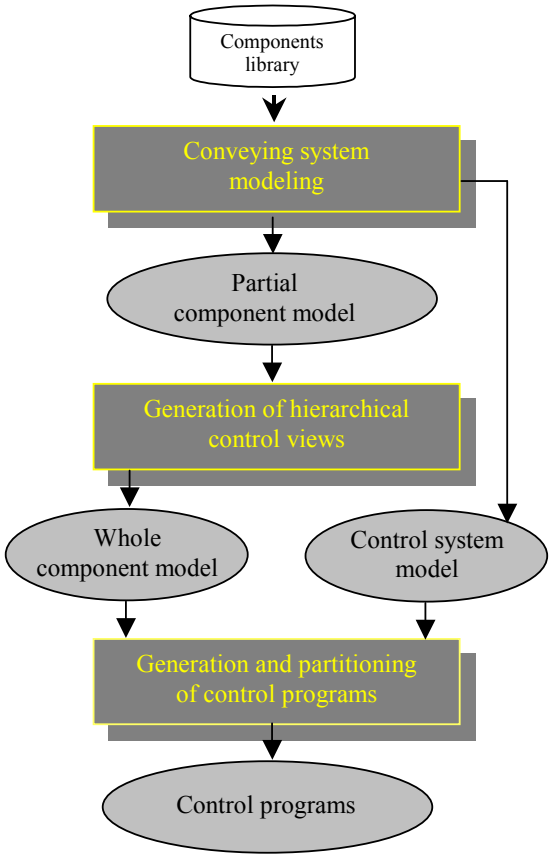


Figure 6: Control design methodology.

The control design methodology delineated in Figure 6 involves three steps : conveying system modeling, generation of hierarchical control views and generation and partitioning of control programs.

These steps involve different kinds of models to generate control programs. The first step called conveying system modeling is dedicated to the creation of the partial component model and the control system model by using a components library. These two models are detailed respectively in the sections 4.3 and 4.4. A partial component model is the reference from which the hierarchical control part generation step is performed to obtain a whole component model. This model is also detailed in the section 4.3. Both control system model and the whole component model are used in the step of generation and partitioning of control programs, to generate control programs. The control programs generated are IEC 61131-3 compliant and are expressed using XML (W3C). The XML files containing the control programs are loaded in Straton Workbench tool (Copalp, 2002) that generates back end code.

4.2 Model Engineering

Model engineering is used to implement the three steps emphasized in Figure 6. The model transformation tool used is ATL (Bézivin et al., 2003).

As seen in Figure 7, models are organized in three layers. The bottom layer L1 is the model layer. The previously mentioned models belong to that layer. The meta-models are defined in the next upper layer L2 (for example, the UML meta-model, the "component" meta-model which has been created for that purpose). They serve as definitions for the models. The class diagrams represented on the figures 2, 3, 4 and 5 compose a part of the "component" meta-model. L3 layer is called meta-metamodel and can be the MOF (MetaObject Facility) (OMG, 2002) defined by the Object Management Group (OMG). Model transformations are defined between elements of different meta-models and they are applied on models (conforms to the meta-models used to define the model transformations). Model engineering approaches provide consistency between the different models used in the design.

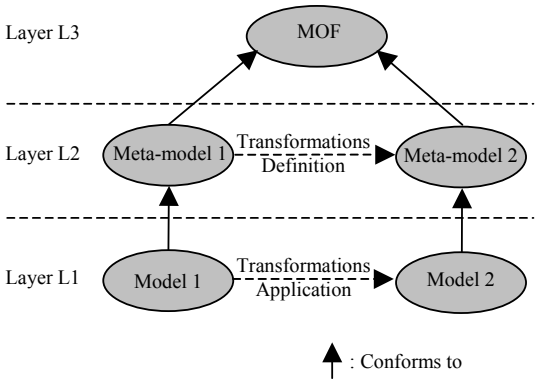


Figure 7: An overview of model transformation.

The following sub-section describes the step of generation of hierarchical control part.

4.3 Generation of Hierarchical Control Part

The step called generation of hierarchical control part is dedicated to the generation of the hierarchical control view of each aggregated component present in a partial component model. A partial component model which models a conveying system, is seen as an assembling of components. This model is known as partial, because it does not contain the control views of aggregated components. An algorithm is proposed to generate automatically the control views of aggregated components. Thus the partial component model is refined to obtain the whole component model. This algorithm divided into three successive phases (figure 8) uses a library of control templates. Each phase is dedicated to the generation of control views of one type of aggregated component.

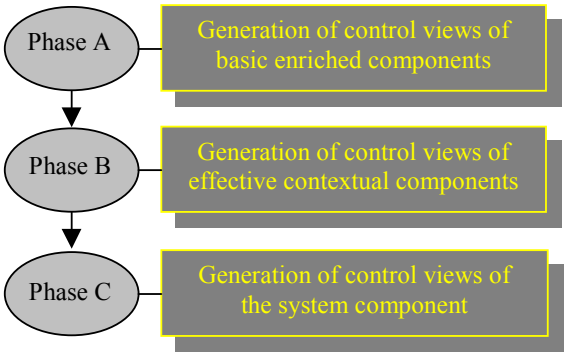


Figure 8: Phases of the generation algorithm of control views .

The two following sub-sub-sections detail the A, B, C phases.

4.3.1 Phases A and B

The first and second phases of the generation algorithm of hierarchical control parts allow to generate the control views of both basic enriched components and effective contextual components. This algorithm is based on a partial system model and a library of control templates. A control template can be compared to a SFC skeleton. In the following paragraph, the first phase of the algorithm is detailed precisely. The second phase is similar to the first one, but applied to effective contextual component.

```
For each contextual component in the
partial component model
   For each contextual operation of
   the contextual component
      Select a control Template
      Append the control template
```

Figure 9: Phase A.

The procedure described above (see figure 9) is applied to each basic enriched component present in the partial system model. For each contextual operation, a control template is chosen and supplemented. A control template is selected according to the type of the contextual operation (Require Position or Detect Position) and to the position (beginning, intermediate or end position) to which the contextual operation refers to. For example, a contextual operation of type “require position” which refers to a beginning or an ending position does not use the same control template than a contextual operation of the same type which refers to an intermediate position. The template is then appended according to information contained in the model.

Four templates have been defined for the phase A and twelve for the phase B.

4.3.2 Phase C

The third phase of the generation algorithm of hierarchical control part is different from the preceding ones. Indeed, the algorithm detailed in the previous section has been enriched with a new function called “constraint view simplifications” (see figure 10).

Indeed, constraint views express the conditions for beginning and for stopping effective contextual operations. They relate to effective contextual components and to the system component. Some constraints can be expressed on the same effective contextual operation by several components. However, to generate the control view of the system component, it is necessary to express only one control constraint by effective contextual operation. The function of constraints views simplification allows to simplify the different constraints to have only one constraint by effective contextual operation.

```
Constraints views simplification
For each contextual component in the
partial component model
      Select a control Template
      Append the control template
```

Figure 10: Phase C.

Then, for each control constraint, a control template is chosen and appended. Two templates have been defined. The first template is used when a control constraint does not define activation conditions and the second template is used when a control constraint defines conditions for beginning and for stopping an effective contextual operation.

4.4 Control Programs Generation and Partitionning

On the figure 6, the step named "generation and partitioning of control programs" makes it possible to generate the control programs, which are to be implemented by PLCs, without any transcription. This step uses a control system model which captures all aspects of a control system in terms of implementation (hardware components) and a whole component model for description of control functionalities. The approach described here follows the Model Driven Architecture (Millar and al., 2001) methodology, proposed by the OMG. The main characteristic of MDA methodology is to separate the functionalities of an application from its development using particular technlologies. The system functionalities are defined by the Platform Independent Model (PIM). In our approach, component model corresponds to the PIM. The projection of functionalities on the hardware architecture defines the Platform Specific Model (PSM). Thus in our approach, the PSM corresponds to control programs which can be implemented on PLCs. The hardware architecture (figure 11) which is mainly composed of PLCs, is described in the control system model.

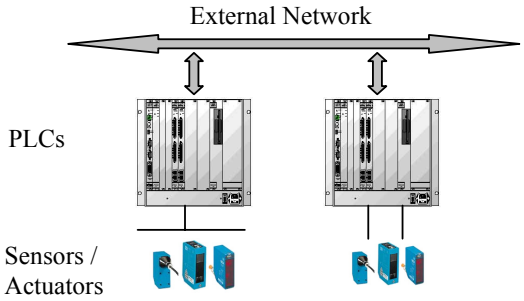


Figure 11: An example of hardware architecture.

5 EXPERIMENTATIONS

The methodology for the control design of conveying system has been successfully validated on a simple system composed of one motorized conveyor, one jack and one sensor. The behavior of the system is as follows: when a parcel is detected by the sensor, the jack ejects the parcel. The system has been modeled. From this model, the methodology presented in the section 4, has been applied to obtain a XML file (control programs). The control programs are composed of 12 SFCs and 6 I/Os. They have been validated by using SimSED tool (Lallican et al., 2006).

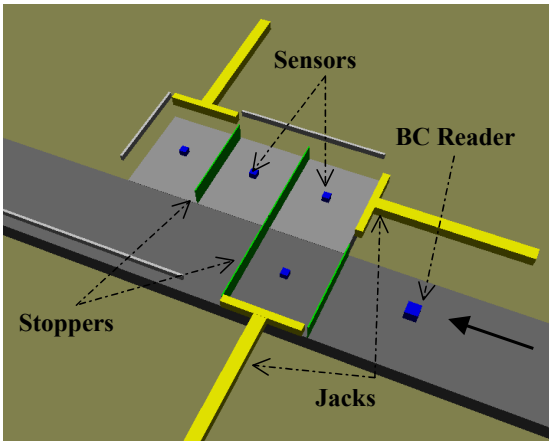


Figure 12: Example of working area.

The methodology has also been experimented on more complex application that is based on a working area of an industrial conveyor (figure 12).

It is composed of a bar code reader to identify parcels, eleven sensors to detect parcels and positions of jacks, three jacks, four stoppers and two conveyors. The working area can accept three product simultaneously. This system is controlled by

a single PLC. In the model of this system we find : 2 effective contextual components, 7 basic enriched components, 2 support components and 18 basic components. The control programs (XML file) generated are composed of 61 SFCs and 21 I/Os. Some parts of the XML file are represented on the figure 13.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<K5project version="1.1" path="D:\\StraProj\\testTrMStraton\\">
…..
 <variables>
  <varGroup name="%IX0" kind="IO">
  <var name="%IX0.0=Sensor3_I_D" type="BOOL"/>
  <var name="%IX0.1=Sensor23_I_D" type="BOOL"/>
  <var name="%IX0.2=Sensor4_I_D" type="BOOL"/>

 <programs>
  <pou name="Jack3" kind="program" period="1" phase="0" lge="SFC">
   <defines name="Jack3"/>
   <sourceSFC>
    <SFCstep kind="init" dx="1" dy="0" ref="10" name="GS10" next="GT11">
    </SFCstep>
    <SFCstep kind="init" dx="0" dy="0" ref="0" name="GS0" next="GT1">
    </SFCstep>
    <SFCstep kind=" init " dx="1" dy="2" ref="12" name="GS12" next="GT13">
     <SFCaction kind="default">
      <sourceSTIL>Jack3_O_R;</sourceSTIL>
     </SFCaction>
```

Figure 13: Example of parts of the XML file.

All the behaviors have been simulated to check the provided control. The component with its control is stored for reusing in a project of a conveyor with five working areas.

6 CONCLUSION

Components have been introduced, and the advantages they offer have been pointed out as they may be very useful to design the control of conveying systems through the views they gather. A methodology allowing to generate automatically control programs (IEC 61131-3 standard compliant) has also been described. This methodology allows the reduction of the development costs by improving, facilitating and systematising the creation of the control programs. The control programs are created at a higher level of abstraction: engineers manipulate models instead of languages of the IEC 61131-3 standard. The main drawback of this methodology is that the generated programs will be bigger than if they were built without using any methodology.

Transformation model techniques have been proved to be very powerful to implement code generation. The methodology has been illustrated through two examples.

Further works focus on the partitioning of control programs to obtain a distributed control. Thus, it will make possible applying the methodology on a industrial scale system.

REFERENCES

Berruet, P., Coudert, T., Philippe, J.L, 2003, Integration of dependability aspects in transitic systems, *Proc. IEEE-IMACS CESA 2003, Lille*.

Berruet P., Lallican J-L., Rossi A., Philippe J-L., 2005, A component based approach for the design of FMS control and supervision*, IEEE SMC , Hawaii*.

Bézivin, J., Dupé, G., Jouault, F., Pitette, G., Eddine Rougui, J., 2003, First Experiments with the ATL Transformation Language: Transforming XSLT into Xquery, *2ndOOPSLA Workshop on Generative Techniques in the context of Model Driven Architecture, Anaheim, California*.

Bézivin, J., 2005, On the Unification Power of Models, *Software and SystemModeling, Springer Verlag*.

Copalp, 2002, Straton Handbook.

Coudert, T., Berruet, P., Philippe, J.L., 2002, From Design to Integration of Transitic Systems A Component Based Approach, *Proc. IECON'02, Sevilla, Vol. 1, pp. 2487-2502.*

IEC 61131-3, 1993, International Electrotechnical Commission 61131-3, Programmable controllers - Part 3: programming languages.

Lallican, J.L., Berruet, P., Philippe, J.L, 2005, SimSED: a tool for modeling and Simulating Transitic Systems, *I3M, CMS 2005, Marseille.*

Lallican, J.L., Berruet, P., Rossi, A, Philippe, J.L, 2006, SimSED: un environnement pour modéliser et simuler des systèmes transitiques, *MOSIM, Rabat.*

Millar, J., Mukerji, J., 2001, Model Driven Architecture (MDA), *OMG, ormsc/2001-07-01, Architecture Board ORMSC1.*

OMG, 2002, OMG Meta Object Facility (MOF) Specification.

Toguyeni, A.K.A., Berruet, P., Craye, E., 2003, Models and algorithms for failure diagnosis and recovery in FMS, *Int. J. of Flexible Manufacturing Systems, Vol 15, N°1, pp. 57-85.*

W3C, Extensible Markup Language : XML, http://www.w3.org/XML/

POSTERS

STABILIZATION OF UNCERTAIN NONLINEAR SYSTEMS VIA PASSIVITY FEEDBACK EQUIVALENCE AND SLIDING MODE

Rafael Castro-Linares
CINVESTAV-IPN, Department of Electrical Engineering, Av. IPN 2508, Col. San Pedro Zacatenco, 07360 Mexico, D.F., Mexico
rcastro@cinvestav.mx

Alain Glumineau
IRCCyN, UMR 6597 CNRS, Ecole Centrale de Nantes, 1 rue de la Noe, 44321 Nantes Cedex 03, France
Alain.Glumineau@irccyn.ec-nantes.fr

Keywords: Passivity feedback equivalence, sliding mode technique, stabilization, uncertain nonlinear systems.

Abstract: In this paper, a sliding mode controller based on passivity feedback equivalence is developed in order to stabilize an uncertain nonlinear system. It is shown that if the nominal passive system obtained by feedback equivalence is asymptotically stabilized by output feedback, then the uncertain system remains stable provided the upper bounds of the uncertain terms are known. The results obtained are applied to the model of a magnetic levitation system to show the controller methodology design.

1 INTRODUCTION

In the last decade the concept of passivity has been mainly used in the stability analysis of continous-time state-space nonlinear systems (Cai and Han, 2005; Mahmoud and Zribi, 2002) and to analyze the stability properties of nonlinear interconnected systems and special cascaded structures (Byrnes et al., 1991; Ortega, 1991). Besides, an important question arises when the model of the system contains uncertain elements such as constant or varying parameters that are not known or imperfectly known. Under such imperfect knowledge of the model, the feedback that makes the uncertain system passive is no longer robust. Some works using nonlinear adaptive control have been recently devoted to this issue (Su and Xie, 1998; Duarte-Mermoud et al., 2002). On the other hand, the control of nonlinear systems with uncertainties via the sliding mode technique has been widely studied in the literature to attain robust control structures; see, for example the results presented in (Tunay and Kaynak, 1995).

The goal of the present paper is to develop a controller via passivity feedback equivalence and sliding modes that permits to stabilize an uncertain nonlinear system. Stabilization is obtained whenever the passive system associated to the nominal system is asymptotically stabilized by output feedback; a similar approach was presented in (Loria et al., 2001) where a different sliding surface is proposed. The study is completed by means of an example of height distance regulation in the model of a magnetic levitation system.

2 PASSIVITY EQUIVALENCE AND STABILIZATION USING SLIDING MODES

One considers *uncertain* MIMO nonlinear systems described by

$$\Sigma^U : \begin{cases} \dot{x} = f(x) + \Delta f(x) + (g(x) + \Delta g(x))u, \\ y = h(x) \end{cases} \quad (1)$$

where $x \in \Re^n$ is the state vector, $u \in \Re^p$ is the input vector, $y \in \Re^p$ is the output vector. f and the p columns of the matrix g are C^∞ vector fields, and the p components of the vector h are C^∞ functions. Δf and the p columns of the matrix Δg are smooth vector fields defined on $\Re^n$ which represent the model uncertainties. In addition, we suppose, without loss of generality and after a possible coordinates shift, that $f(0) = 0$ and $h(0) = 0$. The MIMO nonlinear system (1) without uncertainties, also referred as the *nominal system*, is described by

$$\Sigma : \begin{cases} \dot{x} = f(x) + g(x)u, \\ y = h(x). \end{cases} \quad (2)$$

This is, Σ is given by Σ^U with $\Delta f(x) = 0$ and $\Delta g(x) = 0$ for all x.

Let us now assume that the nominal system Σ has relative degrees $r_1 = 1, \ldots, r_p = 1$, that the matrix $L_g h(0)$ is nonsingular and that it is weakly minimal phase; this is, system (2) is locally equivalent to a passive system (Byrnes et al., 1991). Let $S(y,v) = col\{S_1(y,v), \ldots, S_p(y,v)\}$ be an p dimensional smooth function that we refer as the *switching function* where v is a new input signal. In this work, we set $S(y,v)$ as

$$S(y,v) = y - \int_0^t v(\tau)d\tau. \tag{3}$$

In the sliding mode, $S = \dot{S} = 0$, and the state trajectory of the nominal system is constrained to evolve on the *sliding surface* M_S by the so-called *equivalent control* $u = u_{eq}$. If an initial point does not belong to M_S, the *attractivity condition* $(\dot{S})^T S \leq -\lambda$ with $\lambda > 0$ must be satisfied in a neighbourhood of M_S, so that this surface becomes attractive (Utkin, 1992). The control law which permits to reach the sliding surface can be obtained from the expression $\dot{S} = -F(S)$ where $F(S)$ is, in general, a discontinuous vector function of its arguments.

Writting the uncertain system (1) in the new coordinates (y,z), with z being a set of complimentary coordinates, and substituting the feedback

$$u = u_{slid} = b(y,z)^{-1}[-F(S) - a(y,z) + v]. \tag{4}$$

where $b(y,z)$ is nonsingular for all (y,z) near $(0,0)$ and setting $F(S) = \Gamma sign(S)$ where $sign(S) := col\{sign(S_1), \ldots, sign(S_p)\}$ and $\Gamma > 0$, one has

$$\tilde{\Sigma}^U : \begin{cases} \dot{y} = v - \Gamma sign(S) + \Delta a(y,z) \\ \qquad +\Delta b(y,z) b^{-1}(y,z)(-a(y,z) - \Gamma sign(S) + v) \\ \dot{z} = f^*(z) + p(y,z)y + \{\sum_{i=1}^m q_i(y,z)y_i\}v \\ \qquad +\Delta p(y,z)y + \{\sum_{i=1}^m \Delta q_i(y,z)y_i\}v \\ \qquad +\{\sum_{i=1}^m \Delta r_i(y,z)y_i\}\Gamma sign(S) \end{cases} \tag{5}$$

where $p(y,z)$ and the $q_i(y,z)$'s are suitable matrices of appropriate dimensions and $\dot{z} = f^*(z)$ are the so called *zero dynamics* of the nominal system. $\Delta p(y,z)$, the $\Delta q_i(y,z)$'s, and the $\Delta r_i(y,z)$'s are matrices which represent the terms associated to the uncertainties in the z variables. $\Delta a(y,z)$ and $\Delta b(y,z)$ represent the uncertainties associated to the y variable.

Since it is assumed that the nominal system is weakly minimal phase, its zero dynamics are Lyapunov stable with a time-independent and C^2 Lyapunov function $W^*(z)$, and one chooses the signal v as (Byrnes et al., 1991)

$$v = [I + M(y,z)]^{-1}[-(L_{p(y,z)}W^*(z))^T + w] \tag{6}$$

where $M(y,z) = [(L_{q_1}W^*)^T \cdots (L_{q_p}W^*)^T]^T$. This choice makes the closed-loop nominal system $[\dot{y}^T \dot{z}^T] = \bar{f}(y,z) + \bar{g}(y,z)w$ passive from the input w to the output y. Assuming that this passive system is also locally *zero state detectable*[1], its equilibrium $(y,z) = (0,0)$ can be can be made asymptotically stable by the simple output feedback $w = -\phi(y)$ with $\phi(0) = 0$ and $y^T\phi(y) > 0$ for each $y \neq 0$. Let us define define $\xi = (y,z)$ and substitute the assignment (6) together with $w = -\phi(y)$ into the uncertain system (5). The resulting closed-loop system can then be written as

$$\dot{\xi} = \bar{F}(\xi) + \bar{G}(\xi) \tag{7}$$

where

$$\bar{F}(y,z) = \bar{f}(y,z) - \bar{g}(y,z)\phi(y), \quad \bar{G}(\xi) = \bar{G}_1(\xi) + \bar{G}_2(\xi)$$

and

$$\bar{G}_1(y,z) = \begin{bmatrix} \bar{G}_{11}(y,z) \\ 0 \end{bmatrix}, \bar{G}_2(y,z) = \begin{bmatrix} 0 \\ \bar{G}_{22}(y,z) \end{bmatrix} \tag{8}$$

with

$$\begin{aligned} \bar{G}_{11}(y,z) = &-\Gamma sign(S) + \Delta a(y,z) \\ &+\Delta b(y,z) b^{-1}(y,z)(-a(y,z) - \Gamma sign(S) \\ &+[I + M(y,z)]^{-1}[-(L_{p(y,z)}W^*(z))^T - \phi(y)]), \end{aligned}$$

$$\begin{aligned} \bar{G}_{22}(y,z) = &\Delta p(y,z)y + \{\textstyle\sum_{i=1}^m \Delta r_i(y,z)y_i\}\Gamma sign(S) \\ &+\{\textstyle\sum_{i=1}^m \Delta q_i(y,z)y_i\}[I \\ &+M(y,z)]^{-1}[-(L_{p(y,z)}W^*(z))^T - \phi(y)]. \end{aligned}$$

We now assume that the uncertain terms satisfy the uniform bounds

$$\| \bar{G}_1(\xi) \| \leq \delta_1, \quad \| \bar{G}_2(\xi) \| \leq \delta_2 \tag{9}$$

for all $\xi \in D$ where $D = \{\xi \in \Re^n : || \xi || < r\}$ with $r > 0$ or, equivalently,

$$\| \bar{G}(\xi) \| \leq \delta_1 + \delta_2 = \delta \tag{10}$$

for all D. Notice that $\xi = 0$ is a locally asymptotically equilibrium point of the system $\dot{\xi} = \bar{F}(\xi)$ and one can then assure, by using the Lyapunov approach, that for all bounded initial conditions $\xi(0)$, the solution $\xi(t)$ of the uncertain system (7) is locally ultimately bounded for $t \geq 0$. Moreover, one can show that the sliding surface M_S becomes attractive for any initial point $\xi(0) \in D$ if

$$\begin{aligned} \Gamma \geq [1 - \| \Delta b b^{-1} sign(S) \|]^{-1}[\| \Delta a \| \\ + \| \Delta b b^{-1}([I + M]^{-1}[-(L_p W^*)^T - \phi] - a) \| + \lambda] \end{aligned} \tag{11}$$

whenever $\| \Delta b b^{-1} sign(S) \| \neq 1$, with λ being a nonzero positive constant (see, in particular, (Khalil, 1996), Lemma 5.3, Chapter 5, p. 216).

[1] A system (2) is locally zero-state detectable if there exists a neighbourhood U of 0 such that, for all $x \in U$, $y(t) = h(x(t)) \equiv 0$ implies that $x(t) \to 0$ as $t \to \infty$. It is said to be locally zero-state observable if there exists a neighbourhood U of 0 such that, for all $x \in U$, $y(t) = h(x(t)) \equiv 0$ implies that $x(t) = 0$.

3 APPLICATION TO THE MODEL OF A MAGNETIC LEVITATION SYSTEM

In this work we consider the single-axis levitation system described in (Cho et al., 1993) (see Fig. 1). A force balance analysis leads to a state space representation of the system with state $x = (x_1, x_2) = (d - d_0, \dot{d} - \dot{d}_0)$ and control input $u = V_c - V_{c0}$ where d is the the distance of the ball from the reference line and V_c is the control voltage applied to the amplifier; d_0 and $\dot{d}_0$ are equilibrium points for a given nominal control voltage V_{c0}. The state space representation is given by

$$\dot{x} = f(x) + g(x)u, \quad y = h(x) = x_2 \tag{12}$$

with

$$f(x) = \begin{bmatrix} x_2 \\ \hat{b}(x_1)V_{c0}/m - g \end{bmatrix}, \; g(x) = \begin{bmatrix} 0 \\ \hat{b}(x_1)/m \end{bmatrix} \tag{13}$$

where m is the mass of the ball, g is the gravity and $\hat{b}(x_1) = 1/[a_1(x_1 - d_0)^2 + a_2(x_1 - d_0) + a_3]$, with a_1, a_2 and a_3 being real constant parameters. Since $L_g h(x) = \hat{b}(x_1)/m \neq 0$, the system has a relative degree $r = 1$. Thus, in the coordinates $\xi = (y, z) = (x_2, x_1)$, the levitation system (12),(13) takes the form

$$\begin{aligned} \dot{y} &= [\hat{b}(z)V_{c0}/m - g] + [\hat{b}(z)/m]u, \\ \dot{z} &= y. \end{aligned} \tag{14}$$

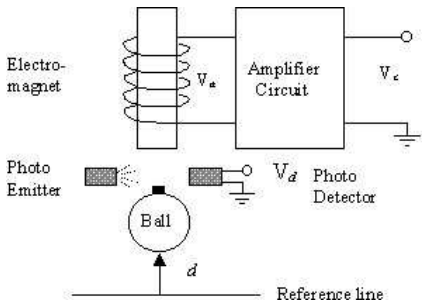


Figure 1: Schematic diagram of the magnetic levitation system.

The system's zero dynamics are then described by the first order differential equation $\dot{z} = f^*(z) = 0$ for which the quadratic positive definite function $W^*(z) = (1/2)z^2$ satisfies $L_{f^*}W^*(z) = 0$, and the system is weakly minimum phase. One then has that the feedback

$$u = \frac{m}{\hat{b}(z)}[-\frac{\hat{b}(z)}{m}V_{c0} + g - z + w] \tag{15}$$

makes the system (14) feedback equivalent to a C^2 passive system from w to y with a C^2 storage function $V = W^*(z) + (1/2)y^2$. Even more, the resultant closed-loop system is a loosless one because of the fact that $\dot{V} = yw$. One can also verify that this closed-loop system is zero-state observable, thus the additional feedback

$$w = -ky, \tag{16}$$

with $k > 0$, can make the origin $(y, z) = (0, 0)$ of the system

$$\dot{\xi} = \begin{bmatrix} \dot{y} \\ \dot{z} \end{bmatrix} = \bar{F}(\xi) = \begin{bmatrix} -k & -1 \\ 1 & 0 \end{bmatrix} \xi = \bar{A}\xi. \tag{17}$$

asymptotically stable.

In (Cho et al., 1993) it is noticed that the solenoid characteristics change with temperature, and a change of $\pm 20\%$ can appear in $\hat{b}(x_1)$ when the levitation system has been operated for a short period of time. Thus, the actual force-distance relationship, denoted by $b(d)$, may be expressed as

$$b(d) = \hat{b}(d) + \Delta\hat{b}(d) \tag{18}$$

where $\Delta\hat{b}(d)$ is an unknown modeling error which can be as high as 20% of $\hat{b}(d)$. The uncertain model associated to the nominal model (14) can then be written, also in the coordinates $\xi = (y, z)$, as

$$\begin{aligned} \dot{y} &= [\hat{b}(z)V_{c0}/m - g] + [\Delta\hat{b}(z)V_{c0}/m] \\ &\quad + ([\hat{b}(z)/m] + [\Delta\hat{b}(z)/m])u, \\ \dot{z} &= y. \end{aligned} \tag{19}$$

This is, the uncertainties are given by $\Delta a(y, z) = \Delta\hat{b}(z)V_{c0}/m$ and $\Delta b(y, z) = \Delta\hat{b}(z)/m$.

The switching function $S(y, v)$ is given by (3) with $v = -z + w$. Such a choice leads to the control law

$$u = u_{slid} = \frac{m}{\hat{b}(z)}[-\Gamma sign(S) - \frac{\hat{b}(z)}{m}V_{c0} + g - z + w], \tag{20}$$

with $\Gamma > 0$, which allows to reach the sliding surface in a finite time. By selecting the additional output feedback (16), we obtain the closed-loop system

$$\dot{\xi} = \bar{A}\xi + \bar{G}(\xi) \tag{21}$$

where

$$\bar{G}(\xi) = \bar{G}_1(\xi) = \begin{bmatrix} [-\Gamma sign(S) + \frac{\Delta\hat{b}(z)V_{c0}}{m} \\ + \frac{\Delta\hat{b}(z)}{\hat{b}(z)}[\frac{\hat{b}(z)V_{c0}}{m} + g - \Gamma sign(S) \\ - z - ky]] \\ 0 \end{bmatrix} \tag{22}$$

From the size of the modelling error $\Delta\hat{b}(z)$ one can verify, after some computations, that the uncertainty term $\bar{G}_1(\xi)$ satisfies the uniform bound $|| \bar{G}_1(\xi) || \leq \delta$ for a constant δ. It then follows that the solution $\xi(t)$ of the uncertain system (21) is ultimately bounded for $t \geq 0$.

The magnetic levitation system described by equations (12),(13) was simulated together with the passivity based sliding mode controller (3),(20). The nominal value of the ball's mass m and the constant coefficients used in the force-distance relationship $\hat{b}(z)$ were selected as in (Cho et al., 1993), this is $m = 2.206\ gr$, $a_1 = 0.0231/mg$, $a_2 = -2.4455/mg$, $a_3 = 64.58/mg$. In fact, as it is noted in (Cho et al., 1993), the validity of the $\hat{b}(x_1)$ is constrained to the range of 35 mm and 48 mm. By choosing the nominal value of the control applied to the amplifier circuit to be $V_{c0} = 4.87\ volts$, we obtained the equilibrium point $(d_0, \dot{d}_0) = (38.2\ mm, 0\ mm/sec)$. The initial conditions of the magnetic levitation system were fixed to $x_1(0) = 44.2\ mm$ and $x_2(0) = 0\ mm/sec$, while the controller parameters were selected as $\Gamma = 10$ and $k = 2$. In order to diminish the effect of chattering due to the discontinuity of the sign function, a saturation function given by

$$sat(S) = \begin{cases} 1, & if\ \ S > \varepsilon \\ S/\varepsilon, & if\ \ -\varepsilon \leq S \leq \varepsilon \\ -1, & if\ \ S < -\varepsilon \end{cases}$$

with $\varepsilon > 0$, was used instead of the *sign* function. In order to evaluate the performance of the control scheme, a variation of 20% in the value of the function $\hat{b}(z)$ was introduced at $t = 7\ sec$ in all the simulations. The time closed-loop plot corresponding to the distance d is shown in Figures 2 for $\varepsilon = 0.001$. From this plot, we can notice that the distance of the ball to the reference line is always regulated to the equilibrium point $d_0 = 38.2\ mm$ with no overshoot.

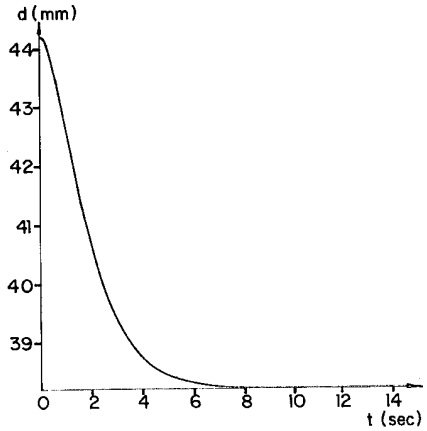


Figure 2: Closed-loop response of the distance, d; $\varepsilon = 0.001$.

4 CONCLUSIONS

In this paper, a passivity-based sliding mode controller design that allows to stabilize an uncertain nonlinear system has been presented. The proposed controller has also been applied to the model of a magnetic levitation system in order to regulate the height of a levitated ball around at one of its equilibria.

REFERENCES

Byrnes, C. I., Isidori, A., and Williams, J. C. (1991). Passivity, feedback equivalence, and the global stabilization of minimum phase nonlinear systems. In *IEEE Transactions on Automatic Control, vol.36, pp. 1228-1240*. IEEE.

Cai, X. S. and Han, Z. Z. (2005). Inverse optimal control of nonlinear systems with structural uncertainty. In *IEE Proceedings Control Theory and Applications, vol. 152, pp. 79-83*. IEE.

Cho, D., Kato, Y., and Spilman, D. (1993). Sliding mode and classical control of magnetic levitation systems. In *IEEE Control Systems, pp. 42-48*. IEEE.

Duarte-Mermoud, M. A., Castro-Linares, R., and Castillo-Facuse, A. (2002). Direct passivity of a class of mimo non-linear systems using adaptive feedback. In *International Journal of Control, vol. 75, pp. 23-33*. Taylor and Francis.

Khalil, H. (1996). *Nonlinear Systems*. MacMillan Publishing Company, New York, 2nd edition.

Loria, A., Panteley, E., and Nijmeier, H. (2001). A remark on passivity-based and discontinuous control of uncertain nonlinear systems. In *Automatica, vol.37, pp. 1481-1487*. Elsevier.

Mahmoud, M. S. and Zribi, M. (2002). Passive control synthesis for uncertain systems with multiple-state delays. In *Computers and Electrical Enginnering, vol.28, pp. 195-216*. Pergamon.

Ortega, R. (1991). Passivity properties for stabilizing of cascaded nonlinear systems. In *Automatica, vol. 27, pp. 423-424*. Elsevier.

Su, W. and Xie, L. (1998). Robust control of nonlinear feedback passive systems. In *Systems Control Letters, vol. 28, pp. 85-93*. Elsevier.

Tunay, I. and Kaynak, O. (1995). A new variable structure controller for affine nonlinear systems with nonmatching uncertainties. In *International Journal of Control, vol. 62, pp. 917-939*. Taylor and Francis.

Utkin, V. I. (1992). *Sliding Modes in Control and Optimization*. Springer, New York, 1st edition.

GENERAL FORMULATION OF SYSTEM DESIGN PROCESS

Design Process Formulation as a Controllable Dynamic System

Alexander Zemliak
Department of Physics and Mathematics, Puebla Autonomous University, Av. San Claudio s/n, Puebla, Mexico
Institute of Technical Physics, National Technical University of Ukraine, Prospect Peremogy 37, Kyiv, Ukraine
azemliak@fcfm.buap.mx

Roberto Galindo-Silva
Department of Electronics, Puebla Autonomous University, Av. San Claudio s/n, Puebla, Mexico
robertogs@hotmail.com

Keywords: Circuit design, control theory formulation, time minimization.

Abstract: The formulation of the process of analogue circuit design has been done on the basis of the control theory application. This approach produces the set of different design strategies inside the same optimization procedure. Basic equations for this design methodology were elaborated. The problem of the time-optimal design algorithm construction is defined as the problem of a functional minimization of the optimal control theory. By this context the design process is defined as a controllable dynamic system. Numerical results of some electronic circuit design demonstrate the efficiency of the proposed methodology and prove the non-optimality of the traditional design strategy.

1 INTRODUCTION

One of the main problems of a large system design is the excessive computer time that is necessary to achieve the final point of the design process. This problem has a great significance at least for the VLSI electronic circuit design. Any system design methodology includes two main parts: the block of analysis of the mathematical model of the system and optimization procedure that achieves the cost function optimal point during the design process. This is a traditional design approach for the system design and we call it as a Traditional Design Strategy (TDS). There are some powerful methods that reduce the necessary time for the circuit analysis by means of the special sparse matrix techniques (Osterby, Zlatev, 1983), (George, 1984) or by the partitioning of a circuit matrix by branches (Wu, 1976) or by nodes (Sangiovanni-Vincentelli et al, 1977).

Another formulation of the circuit optimization problem was developed in heuristic level some decades ago (Kashirsky and Trokhimenko, 1979). This idea was based on the Kirchhoff laws ignoring for all the circuit or for the circuit part. The special cost function is minimized instead of the circuit equation solving. This idea was developed in practical aspect for the microwave circuit optimization (Rizzoli et al, 1990) and for the synthesis of high-performance analogue circuits (Ochotta et al, 1996) in extremely case, when the total system model was eliminated. The last idea that excludes completely the Kirchhoff laws can be named as the Modified Traditional Design Strategy (MTDS).

More general approach was elaborated in previously work (Zemliak, 2005). This approach can be developed to define the system design problem by means of the optimal control theory.

2 PROBLEM FORMULATION

The design process for any analogue system design can be defined as the problem of the cost function $C(X)$ minimization ($X \in R^N$) with the system of constraints. It is supposed that the minimum of the cost function $C(X)$ achieves all design objects and the system of constraints is the mathematical model of the electronic circuit. It is supposed also that the circuit model can be described as the system of nonlinear equations:

$$g_j(X) = 0 \qquad (1)$$

$$j = 1,2,\ldots, M$$

The vector X is separated in two parts: $X = (X', X'')$. The vector $X' \in R^K$ is the vector of independent variables where K is the number of independent variables and the vector $X'' \in R^M$, is the vector of dependent variables, ($N = K + M$).

The optimization process for the cost function $C(X)$ minimization with constrains (1) can be defined in general case by next vector equation:

$$X^{s+1} = X^s + t_s \cdot H^s \qquad (2)$$

where s is the iterations number, t_s is the iteration parameter, $t_s \in R^1$, H is the direction of the cost function $C(X)$ decreasing. The system (1) must be solved at each step of the optimization process (2) in this case. The optimization process is realized in R^K. This is a TDS.

The specific character of the design process for the electronic systems consists in fact that it is not necessary to fulfil the conditions (1) for all steps of the optimization process. It is quite enough to fulfil these conditions for the final point only.

The problem (1)-(2) can be redefined. We suppose that all components of the vector X are independent. This is the main idea for the penalty function method application. In this case the vector function H is the function of the cost function $C(X)$ and the additional penalty function $\varphi(X)$: $H^s = f\left(C(X^s), \varphi(X^s)\right)$. The penalty function structure includes all equations of the system (1) and can be defined for example as:

$$\varphi(X^s) = \frac{1}{\varepsilon} \sum_{i=1}^{M} g_i^2(X^s) \qquad (3)$$

In this case we define the design problem as the unconstrained optimization (2) in the space R^N without any additional system but for the other type of the cost function $F(X)$. This function can be defined for example as an additive function: $F(X) = C(X) + \varphi(X)$. In this case we reach the minimum of the initial cost function $C(X)$ and comply with the system (1) in the final point of the optimization process. This is a MTDS.

It is possible to generalize the above mentioned idea. We suppose that the penalty function includes a one part of the system (1) only and the other part of this system is defined as constraints. In this case the penalty function includes first Z items only:

$$\varphi(X^s) = \frac{1}{\varepsilon} \sum_{i=1}^{Z} g_i^2(X^s) \qquad (4)$$

where $Z \in [0, M]$ and M - Z equations make up one modification of the system (1):

$$g_j(X) = 0 \qquad (5)$$

$$j = Z+1, Z+2, \ldots, M$$

This idea can be generalized more in case when the penalty function $\varphi(X)$ includes Z arbitrary equations from the system (1). The total number of different design strategies is equal to 2^M if $Z \in [0, M]$. The optimization procedure is realized in the space R^{K+Z}. The different strategies have different computer times. It is appropriate in this case to define the problem of an optimal design strategy search that has the minimal computer time.

3 CONTROL THEORY APPLY

The problem of optimal design can be defined now as the problem of the optimal control. It is possible to define a design strategy by equations (2), (4) with a variable value of the parameter Z during the all optimization process. It means that we can change the number of independent variables and the number of the terms of the penalty function in each point of the optimization procedure. It is convenient to introduce a vector of the special control functions $U = (u_1, u_2, \ldots, u_M)$ for this aim, where $u_j \in \Omega$, $\Omega = \{0;1\}$. The sense of the control function u_j is next: equation number j is presented in the system (4) and the term $g_j^2(X)$ is removed from the right part of the formula (3) when $u_j = 0$, and on the contrary, the equation number j is removed from the system (4) and is presented in the right part of the formula (3) when $u_j = 1$. The optimization procedure for the design process can be defined in discrete (Eq. (2)) or continuous form. In the last case the design process includes the next principal equations:

$$\frac{dx_i}{dt} = f_i(X, U) \qquad (6)$$

$$i = 0,1,\ldots,N$$

$$(1 - u_j) g_j(X) = 0 \qquad (7)$$

$$j = 1,2,\ldots, M$$

$$\varphi(X,U)=\frac{1}{\varepsilon}\sum_{j=1}^{M}u_j\cdot g_j^2(X) \qquad (8)$$

The functions of the right hand part of the system (5) depend on the optimization method and can be determined for example for the gradient method as:

$$f_i(X,U)=-\frac{\delta}{\delta x_i}F(X,U) \qquad (9)$$

$$i=1,2,\ldots,K$$

$$f_i(X,U)=-u_{i-K}\frac{\delta}{\delta x_i}F(X,U)+\frac{(1-u_{i-K})}{t_s}\left\{-x_i^s+\eta_i(X)\right\} \qquad (9')$$

$$i=K+1,K+2,\ldots,N$$

where $F(X,U)=C(X)+\varphi(X,U)$, x_i^s is equal to $x_i(t-dt)$, the operator $\delta/\delta x_i$ means here $\frac{\delta}{\delta x_i}\varphi(X)=\frac{\partial\varphi(X)}{\partial x_i}+\sum_{p=K+1}^{K+M}\frac{\partial\varphi(X)}{\partial x_p}\frac{\partial x_p}{\partial x_i}$, $\eta_i(X)$ is the implicit function ($x_i=\eta_i(X)$) that is determined by the system (7).

All the control functions u_j depend on the current step of the optimization process. The total number of the different design strategies which are produced inside the same optimization procedure is practically infinite. Among all of these strategies exist one or few optimal strategies that achieve the design objects for the minimum computer time. The function $f_0(X,U)$ is determined as the necessary time for one step of the system (5) integration. The additional variable x_0 is determined as the total computer time T for the system design. In this case we determine the problem of the time-optimal system design as the classical problem of the functional minimization of the control theory. In this context the aim of the design process is to result each function $f_i(X,U)$ to zero for the final time t_{fin}, and to minimize the cost function $C(X)$. The aim of the optimal control is to minimize the total computer time x_0 of the design process. It is necessary to find the optimal behaviour of the control functions u_j during the design process.

The idea of the system design problem formulation as the functional minimization problem of the control theory is not depend of the optimization method and can be embedded into any optimization procedures. In this paper the gradient method and the Davidon-Fletcher-Powell (DFP) method were used.

Now the analogue circuit design process is formulated as a dynamical controllable system. By this formulation we need to find the special conditions to minimize the transition time for this dynamical system.

4 NUMERICAL RESULTS

Some electronic circuits have been designed to demonstrate a new system design approach based on the control theory. The design process has been realized on DC mode. The cost function $C(X)$ has been determined as the sum of the squared differences between beforehand defined values and current values of the nodal voltages for some nodes. Numerical results for the transistor amplifier that is shown in Fig. 1 are discussed below.

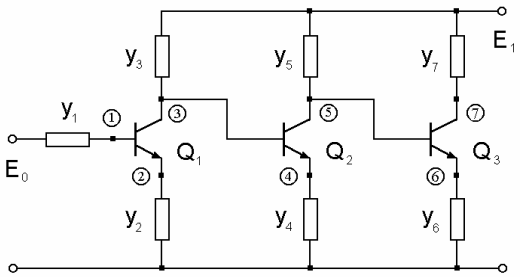


Figure 1: Circuit topology for three-cell transistor amplifier.

The Ebers-Moll static model of the transistor has been used. The analyzed circuit has seven admittance as independent variables $y_1,y_2,y_3,y_4,y_5,y_6,y_7$, ($K=7$) and seven nodal voltages as dependent variables $V_1,V_2,V_3,V_4,V_5,V_6,V_7$, (M=7).

The results of the analysis of the traditional design strategy and some other strategies that have the computer time less than the traditional strategy are given in Table 1. The first line corresponds to the TDS. The last line corresponds to the MTDS. Other nes are the intermediate strategies. The optimal strategies from this table (number 18 and 25 for two optimization procedures respectively) are not optimal in general and the data for the time-optimal

strategies are given in Table 2 by means of the control vector variation.

The time gain of the optimal design strategy with respect to the traditional strategy is equal to 285 for the gradient method and 200 for the DFP method. These data show good perspectives for proposed approach. However the potential time gain is realized only in case when we found the algorithm for the optimal control vector construction.

Table 1: Data of some strategies.

N	Control functions	Gradient	method	DFP	method
	vector	Iterations	Total design	Iterations	Total design
	U (u1,u2,u3,u4,u5,u6,u7)	number	time (sec)	number	time (sec)
1	(0000000)	6379	321.09	854	64.47
2	(0010101)	922	54.53	764	52.29
3	(0010110)	1667	80.71	650	46.13
4	(0010111)	767	35.35	426	22.68
5	(0011100)	3024	159.67	940	52.71
6	(0011101)	823	37.73	177	7.71
7	(0011110)	3068	86.87	450	14.56
8	(0011111)	553	15.75	170	6.93
9	(0110101)	465	10.01	101	2.66
10	(0110110)	1157	31.92	111	3.85
11	(0110111)	501	8.82	124	2.66
12	(0111100)	2643	72.66	314	9.24
13	(0111101)	507	9.24	170	4.62
14	(0111110)	3070	67.27	423	12.25
15	(1010101)	1345	28.07	397	16.94
16	(1010111)	615	10.01	191	4.62
17	(1011101)	699	10.71	197	4.97
18	(1011111)	366	4.97	103	1.96
19	(1110101)	789	10.43	201	4.97
20	(1110110)	3893	61.53	1158	18.06
21	(1110111)	749	7.71	148	2.11
22	(1111100)	4325	90.72	945	19.18
23	(1111101)	796	8.47	133	2.31
24	(1111110)	2149	29.26	1104	13.44
25	(1111111)	2031	5.67	180	0.77

5 CONCLUSIONS

The traditional approach for the analogue circuit design is not time-optimal. The problem of the time-optimum design algorithm can be solved adequately on the basis of the control theory application. The construction of the time-optimal design algorithm is formulated as the problem of a functional minimization of the control theory. This approach can reduce considerably the total computer time for the system design. Analysis of the different electronic systems gives the possibility to conclude that the potential computer time gain of the time-optimal strategy increases when the size and complexity of the system increase. The proposed approach gives the possibility to find the time-optimal algorithm as a solution of the typical problem of the optimal control theory. The optimal structure of the control vector can be finding by the approximate methods of control theory.

ACKNOWLEDGEMENTS

This work was supported by the Mexican National Council of Science and Technology – CONACYT, under project SEP-2004-C01-46510.

REFERENCES

Osterby, O., Zlatev, Z., 1983. *Direct Methods for Sparse Matrices*, Springer-Verlag, N.Y.

George, A., 1984. On Block Elimination for Sparse Linear Systems, *SIAM J. Numer. Anal*. vol. 11, no.3, pp. 585-603.

Wu, F.F., 1976. Solution of Large-Scale Networks by Tearing", *IEEE Trans. Circuits Syst.*, vol. CAS-23, no. 12, pp. 706-713.

Sangiovanni-Vincentelli, A., Chen, L.K., Chua, L.O., 1977. An Efficient Cluster Algorithm for Tearing Large-Scale Networks*, IEEE Trans. Circuits Syst.,* vol. CAS-24, no. 12, pp. 709-717.

Kashirsky, I.S., Trokhimenko, Y.K., 1979. *The Generalized Optimization of Electronic Circuits*, Tekhnika, Kiev.

Rizzoli, V., Costanzo, A., Cecchetti, C., 1990. Numerical Optimization of Broadband Nonlinear Microwave Circuits, *IEEE MTT-S Int. Symp*., vol. 1, pp. 335-338.

Ochotta, E.S., Rutenbar, R.A., Carley, L.R., 1996. Synthesis of High-Performance Analog Circuits in ASTRX/OBLX, *IEEE Trans. on CAD*, vol.15, no. 3, pp. 273-294.

Zemliak, A., 2005. Generalization of Analog System Design Methodology, *In Proc. 5th WSEAS Int. Conf. on Instrumentation, Measurement, Control, Circuits and Syst*., Cancun, Mexico, pp.114.119.

Table 2: Data of the optimal design strategies.

N	Method	Optimal control functions vector U (u1,u2,u3,u4,u5,u6,u7)	Iterations number	Switching points	Total design time (sec)	Computer time gain
1	Gradient method	(1111111); (1111101)	363	350	1.127	285
2	DFP method	(1111111); (1110111)	69	66	0.322	200

DESIGN AND IMPLEMENTATION OF AN FPGA-BASED SVPWM IC FOR PWM INVERTERS

Cheng-Hung Tsai
Department of Electrical Engineering, China Institute of Technology
245 Yen-Chiu Yuan Road, 3rd Sec., Taipei, 11581, Taiwan
chtsai@cc.chit.edu.tw

Hung-Ching Lu
Department of Electrical Engineering, Tatung University
40 Chungshan North Road, 3rd Sec., Taipei, 10451, Taiwan
luhung@.ttu.edu.tw

Keywords: FPGA, SVPWM IC, High-performance.

Abstract: This paper presents a new circuit design scheme of the space-vector pulse-width modulation (SVPWM) strategy, including linear and overmodulation ranges. The proposed scheme has been developed using the state-of-the-art field-programmable gate array (FPGA) technology. The SVPWM control integrated circuit (IC) can be realized by using only a single FPGA (Cyclone) from Altera, Inc. Experimental results show that this controller can present an excellent drive performance and its switching frequency, which can be set to over 100kHz, is adjustable as well as its deadtime. The output fundamental frequency can be adjusted over 2000Hz. This SVPWM IC can be included in the digital current control loop for stator current regulation. The IC also provide a simple hardware and low cost for high-performance ac drives.

1 INTRODUCTION

In engineering practice, because of the complexity of servo control algorithm, it is usually implemented with software based on DSP (Lai and Chang, 1999) (Marwali et al., 1999) (Ma et al., 2001) (Zadeh, 2001). This approach can provide a flexible control scheme, but suffers from a long period of development and exhausts many resources of the CPU. In some cases, dual DSPs have to be adopted to achieve superior performances (Tzou et al., 1996) (Tzou et al., 1996b) (Tzou et al., 1996c). Such additional hardware and software design for such a dual-DSP controller will complicate the design process enormously.

In recent years, a novel design methodology has arisen, that is FPGA-based hardware implementation technology (Carrica et al., 2003) (Man et al., 1995) (Lin et al., 2005) (Zeidman et al., 2002) (Tonelli et al., 2001). Because of the programmable characteristics of FPGA and IP cores, users can design their application-specific integrated circuit (ASIC) in lab according to their schemes, instead of participation of the semiconductor manufacturer. In addition, since FPGA can carry out parallel processing by means of hardware mode, which occupies nothing of the CPU, a very high speed level of the system can be obtained as well as an exciting precision. This novel design methodology has now been used in high performance motion control field, such as (Tonelli et al., 2001) (Tzou and Hsu, 1997) (Zhou et al., 2004). Literature (Tonelli et al., 2001) proposed a universal SVPWM controller with overmodulation and deadtime compensation, but it is not a flexible design and its switching frequency can not be set arbitrarily; besides it also needs an extra EPROM for sin and cosine values as a lookup table. In (Tzou and Hsu, 1997), the constructed IC has a strong function but it does not include overmodulation range and also needs additional EPROM. In (Zhou et al., 2004), the switching frequency can only be set to 40 kHz maximum.

Employing FPGA to realize PWM strategies provides advantages such as rapid prototyping, simple hardware, software design and very high switching frequency. Thus, the realization of the

SVPWM schemes by state-of-the-art FPGA technology is the best way to improve the bandwidth of the current or speed controller. This paper proposed a simple hardware FPGA-based control structure for ac drives depicted in Fig. 1 and develops a high performance realization scheme for the SVPWM control IC employing a single FPGA (Cyclone) from Altera, Inc.

The designed IC can serve either for ac motor drives or three-phase ac-voltage regulation systems. It can also be incorporated as part of the digital current loop for ac drives. Fig. 2 shows the circuit configuration of a DSP-controlled ac drive using the SVPWM control IC. The rest of this paper is organized as follows. Section II briefly introduces the principle of the space-vector PWM method. Section III discusses developing a strategy for the hardware design scheme. Section IV implements the circuits on a low cost FPGA and shows the experimental results. In section V some conclusions are drawn.

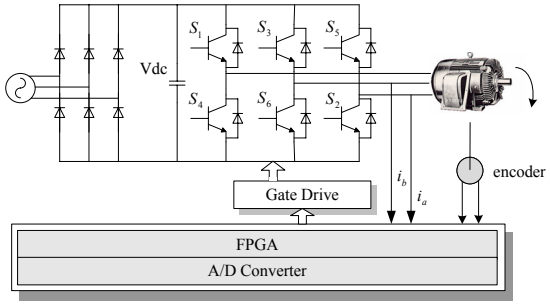


Figure 1: Circuit schematics of a voltage-source full bridge three-phase PWM inverter.

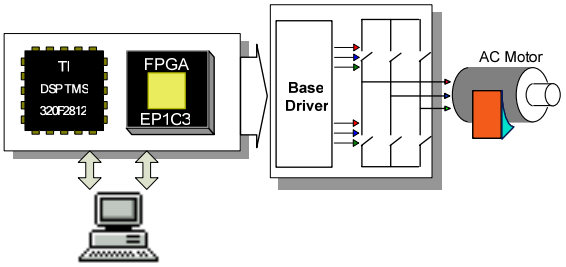


Figure 2: Circuit configuration of the DSP-controlled FPGA-based PWM current controller for the ac drive.

2 PRINCIPLE OF SVPWM

The main purpose of the PWM inverter is to generate a variable-voltage variable-frequency (VVVF) three-phase voltage to the ac motor. Conventional sinusoidal PWM employs several kinds of sampling methods with sinusoidal signals according to a carrier signal, e.g., natural sampling (Schonung and Stemmler, 1964) (Pollack, 1972), or symmetric or asymmetric regular sampling (Bowers, 1975).

The principle of the SVPWM is more clearly explained by representing a space vector (Park, 1929). The motor stator voltage vector can be expressed as a combination of the inverter output-phase voltage V_a, V_b, V_c which can be described in vector form as:

$$\vec{V}_s = V_a + \alpha^2 V_b + \alpha V_c, \ \alpha = \exp(j 2\pi/3) \quad (1)$$

where

$$V_a = V_d \sin wt$$
$$V_b = V_d \sin(wt - 120^\circ)$$
$$V_c = V_d \sin(wt + 120^\circ)$$

and V_d is the amplitude of the fundamental component. As shown in Fig. 3, there are eight voltage vector configurations of the three-phase PWM inverter. Their corresponding voltage vectors are expressed as:

$$\vec{V}_n = \frac{2}{3} V_d \exp\left[j \frac{(n-1)\pi}{3} \right], \quad n = 1,2,...,6 \quad (2)$$

$$V_7 = V_0 = 0 \quad (3)$$

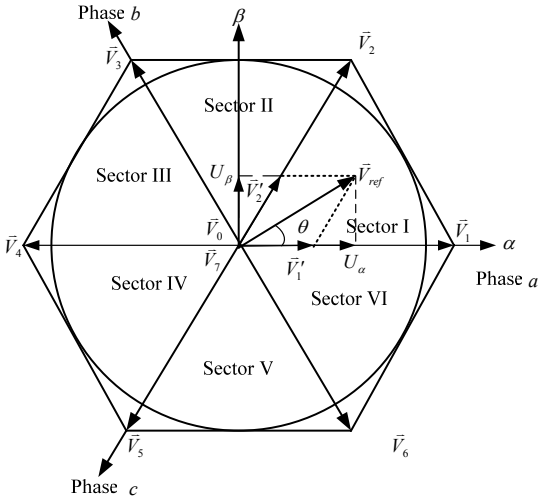


Figure 3: Voltage space vector diagram.

The stator voltage vector can be decomposed of a combination of two basic vectors, as Fig. 3 indicates. The advantage of the SVPWM strategy is the minimization of harmonic distortion in the current

by selecting the appropriate switching vectors and determination of the corresponding dwelling widths.

If the reference vector $\vec{V}_{ref}$ is located in sector I, then it is composed of voltage vector V_1, V_2, and zero voltage vectors V_7 and V_0 as illustrated in Fig. 3. The flux produced by the reference voltage vector in a PWM switching period is a combination of each individual flux produced by its corresponding voltage vector. The relationships of the voltage vector can be expressed as:

$$\int_0^{T_s} V_{ref}\,dt = \int_0^{T_0} V_0\,dt + \int_0^{T_1} V_1\,dt + \int_0^{T_2} V_2\,dt + \int_0^{T_7} V_7\,dt. \quad (4)$$

Because the voltage vectors V_1 and V_2 are basic vectors and V_0 and V_7 are zero vectors, that gives

$$V_{ref} = V_1 \frac{T_1}{T_s} + V_2 \frac{T_2}{T_s} \quad (5)$$

Where T_s is the switching period and T_1, T_2 are the dwelling time of V_1 and V_2, respectively. This voltage space vector can be expressed in rectangular coordinates as follows:

$$T_1 \cdot \sqrt{2/3} \cdot V_d \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} + T_2 \cdot \sqrt{2/3} \cdot V_d \cdot \begin{bmatrix} \cos 60^\circ \\ \sin 60^\circ \end{bmatrix} = \frac{T_s}{2} \sqrt{2/3} \cdot V_d \cdot M \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \quad (6)$$

Where $M = \left|V_{ref}\right| \Big/ \sqrt{2/3} V_d$, $0 \le \theta \le 60^\circ$, and the dwelling time of each vector can be got.

$$T_1 = \frac{T_s}{2} \cdot M \cdot \frac{\sin(60^\circ - \theta)}{\sin 60^\circ} \quad (7)$$

$$T_2 = \frac{T_s}{2} \cdot M \cdot \frac{\sin(\theta)}{\sin 60^\circ} \quad (8)$$

$$T_0 = T_{a7} = \frac{T_s}{2} - T_1 - T_2 \cdot \quad (9)$$

If the calculated value $T_1 + T_2 > T_s$, i.e., the zero vector time is minus, overmodulation occurs and the duration time should be processed again. To improve the resolution of the binary divider, the new time can be reconsidered as follows:

If $T_1 > T_2$, then

$$T_1' = \frac{T_1}{T_1 + T_2} \frac{T_s}{2}, \quad T_2' = \frac{T_s}{2} - T_1' \quad (10)$$

otherwise

$$T_2' = \frac{T_2}{T_1 + T_2} \frac{T_s}{2}, \quad T_1' = \frac{T_s}{2} - T_2' \quad (11)$$

3 DESIGN OF THE FPGA-BASED SVPWM IC

Many factors need to be considered in designing the PWM control IC, such as simplicity, flexibility, and complexity of the circuit design. In practical applications, the PWM IC is not possible to corporate with a conventional microprocessor, therefore, the control IC is stand alone. The major design goal is to relieve the microprocessor from time-consuming computational tasks such as PWM signal generating, deadtime compensation , and current control.

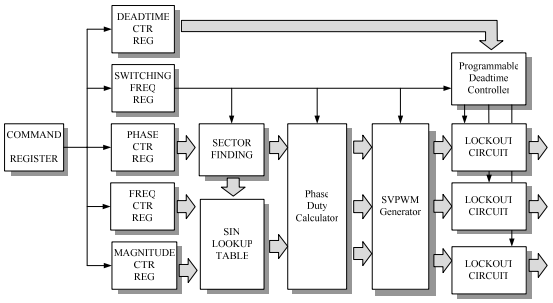


Figure 4: Functional block diagram of the programmable FPGA-based SVPWM IC.

Fig. 4 depicts the block diagram of a proposed programmable FPGA-based SVPWM control IC. This design consists of five registers for the command frequency, modulation factor, phase of the stator voltage vector, the switching frequency of the PWM and the deadtime for the power module.

To simplify the interface circuit, the commands of these five registers are downloaded from the computer directly. The control parameters can be set in the original project file. The internals of the designed IC consist of a sin-table address decoder, a duty-ratio calculator, a PWM waveform generator and a programmable deadtime register. The firing time width of an SVPWM waveform involves computing of sin function. Therefore, arithmetic computational methods and the bit length for manipulating data are important factors in designing the digital hardware for the SVPWM. Floating-point arithmetic complicates the approach to hardware

design greatly; only positive integer arithmetic can provide a feasible solution. In this paper, all positive integer arithmetic without external EPROM sin-table reference has been adopted for the digital realization of the SVPWM IC.

The simple requirements for realizing the SVPWM scheme is to first compute the time width of each voltage vector. Second, these three-phase PWM waveforms are converted to centralized PWM waveforms. Finally, the PWM gating signals are inserted with adjustable time delay to protect the power module from short circuit.

The specified voltage amplitude, frequency, switching frequency, initial phase and deadtime are set in the VHDL file or received from DSP to produce the three-phase PWM gating signals. The sin reference is recorded in the built in memory of the FPGA for the PWM duty-ratio generator.

In ideal conditions, the gating signals to the power switches of same phase leg of the PWM inverter should be complementary. However, in order to avoid short circuit in the power semiconductors, an appropriate delay time must be inserted between these two gating signals. The length of this deadtime is depending on the characteristic of adopted IGBT module. Typically the deadtime is set to $2\sim4\,us$. A programmable deadtime controller is included in the designed SVPWM IC, which greatly facilitates its practical applications.

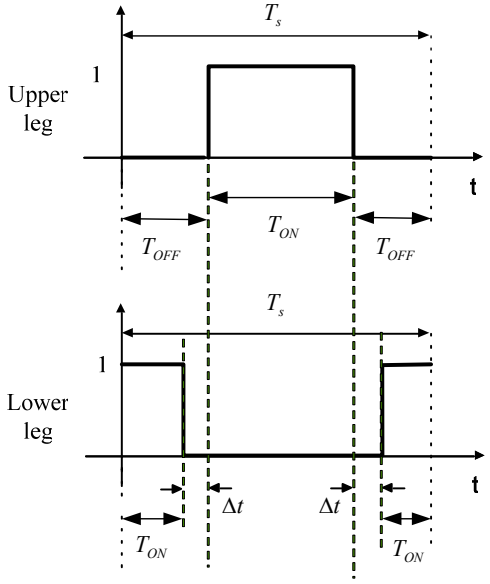


Figure 5: Timing diagram of the PWM signal with deadtime.

The PWM signal waveforms and their corresponding delay signals are described in Fig. 5. The Δt is the specified deadtime. The deadtime controller generates the gating signals to the registers, which includes a digital comparator and results in PWM signals with a specified time delay. Fig. 6 shows simulation results of the programmable FPGA design for the PWM waveform.

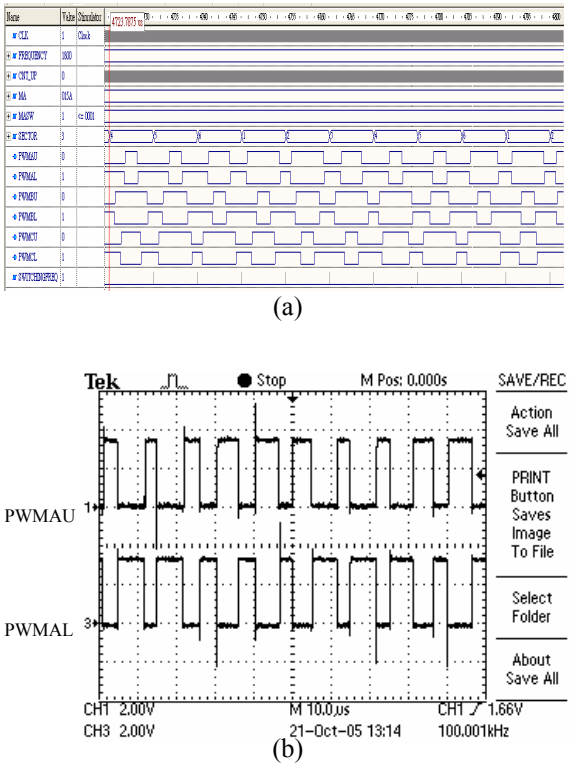


Figure 6: Simulation and experimental results of the voltage vector to three-phase PWM duty ration converter at switching frequency is 100kHz and the output frequency is 1800Hz.

4 HARDWARE IMPLEMENTATIOAND EXPERIMENTAL RESULTS

For the realization of the proposed SVPWM scheme, cost and speed considerations led to selecting an SRAM-based FPGA Cyclone EP1C3 from Altera, Inc. for realizing of the SVPWM IC. The EP1C3 has around 2910 logic gates, 60k RAM bits, and 1 phase-locked loop (PLL). The internal clock can operate at 200 MHz. Some important specifications of the EP1C3 are listed in Table I. Altera also provides EDA tools (Quartus II) for the development of ASIC's employing FPGA's. The Quartus II consists of a schematic entry editor, an interface with the schematic entry editor, logic and timing

simulation software, and design implementation software. The logic and timing simulation software is especially relevant to the design of complicated digital circuits because it is best suited to resolve circuit problems during the early design stage.

Fig. 7 illustrates the pin assignment of the designed SVPWM IC. The SVPWM modules are all described with VHDL and synthesized with Synplify software. The designed IC can operate at 200Mhz system clock by using the internal PLL, and the switching frequency as well as deadtime is adjustable. Fig. 8 is the circuit configuration of the SVPWM IC employing a single-chip DSP (TMS320F2812) from Texas Instruments. The simplicity in the interface circuit design illustrates its feasibility for practical applications. The IC can construct a current loop, and it can also be considered as an IP core which can be integrated into a system on one chip (SOC) with other IP cores.

Fig. 9 illustrates the experimental results of the phase voltage of the SVPWM gating signals from linear to overmodulation region. The output fundamental frequency can be adjusted for over 2000Hz. Such a wide frequency control range, with high-frequency switching, is only feasible by utilizing the state-of-the-art VLSI digital circuit design technique. In this designed IC, the SVPWM scheme is finished in 70 clocks. It means the whole software takes only 350 *ns* . Therefore, the PWM switching frequency can be set for over 100kHz. The deadtime for the PWM gating signals is also adjustable. The PWM waveforms with deadtime are shown in Fig. 10 by using Tektronix oscilloscope. Fig. 11 shows the experimental results of the designed SVPWM IC used in a PWM inverter ac motor drive with 1000 and 2000Hz output, respectively. Experimental results show the constructed SVPWM IC can generate a wide range of output frequencies with controlled fundamental voltage.

5 CONCLUSIONS

This paper presents the design and realization of a programmable SVPWM control IC for high performance ac servo drives. The SVPWM scheme is implemented and tested by using an FPGA technology. Simulation and experimental results are provided to verify the implemented SVPMW control IC. The designed IC is also easy to interface with DSP or other IP cores to form a closed loop control system. Besides, it doesn't need external EPROM, and the source codes can be easily replanted to different FPGA's without any changes. Given that an economic manufacturing cost can be achieved, it is believed that the PWM control IC's will become the important components in power converters and motor drives of the future.

ACKNOWLEDGEMENTS

This work was supported by the National Science Council of the republic of China under the contract of NSC 93-2213-E036-018.

REFERENCES

Y. S. Lai, and S. C. Chang, 1999, "DSP-based implementation of new random switching technique of inverter control for sensorless vector-controlled induction motor drives," *IEE Proceedings- Electric Power Applications,* Vol. 146, No. 2, pp. 163-172.

M. N. Marwali, A. Keyhani, and W. Tjanaka, 1999, "Implementation of indirect vector control on an integrated digital signal processor-based system," *IEEE Transactions on Energy Conversion,* Vol. 14, No. 2, pp. 139-146.

J. D. Ma, W. Bin, N. R. Zargari and S. C. Rizzo, 2001, "A space vector modulated CSI-based AC drive for multimotor applications," *IEEE Transactions on Power Electronics,* Vol. 16, No. 4, pp. 535-544.

S. V. Zadeh, 2001, "Variable flux control of permanent magnet synchronous motor drives for constant torque operation," *IEEE Transactions on Power Electronics*, Vol. 16, No. 4, pp.527-536.

Y. Y. Tzou, M. F. Tsai, Y. F. Lin and H. Wu, 1996, "Dual DSP based fully digital control of an AC induction motor," *Proceedings of the IEEE International Symposium on Industrial Electronics,* Vol. 2, pp. 673-678.

Y. Y. Tzou, M. F. Tsai, Y. F. Lin and H. Wu, 1996, "Dual DSP based fully digital control of an AC induction motor," *Proceedings of the IEEE International Symposium on Industrial Electronics,* Vol. 2, pp. 673-678.

Y. Y. Tzou, W. A. Lee and S. Y. Lin, 1996 , "Dual-DSP sensorless speed control of an induction motor with adaptive voltage compensation," 27th Annual IEEE Power Electronics Specialists Conference, Vol. 1, pp. 351-3.57.

D. Carrica, M. A. Funes and S. A. Gonzalez, 2003, "Novel stepper motor controller based on FPGA hardware implementation," *IEEE/ASME Transactions on Mechatronics*, Vol. 8, No. 1, pp. 120-124.

K. F. Man, Y. C. Ho, K. P. Cheuk and S. Kwong, 1995, "Hardware implementation of variable pulse frequency algorithm," Electronics Letters, Vol. 31, No. 10, pp. 839-840.

F. J. Lin, D. H. Wang and P. K. Huang, 2005, "FPGA-based fuzzy sliding-mode control for a linear induction motor drive**,"** IEE Proceedings- Electric Power Applications, Vol. 152, No. 5, pp. 1137-1148.

B. Zeidman and R. Zeidman, 2002, *Designing with FPGAs and CPLDs*, Berkeley.

M. Tonelli, P. Battaiotto and M. I. Valla, 2001, "FPGA implementation of an universal space vector modulator*," The 27th Annual conference of the IEEE Industrial Electronics Society,* pp. 1172-1177.

Y. Y. Tzou and H. J. Hsu, 1997, "FPGA realization of space-vector PWM control IC for three-phase PWM inverters," *IEEE Trans. On Power Electronics,* Vol. 12, No. 6, pp. 953-963.

Z. Zhou, T. Li, T. Takahashi and E. Ho, 2004, "Design of a universal space vector PWM controller based on FPGA," *IEEE Applied Power Electronics Conference and Exposition,* Vol. 3, pp. 1698-1702.

A. Schonung and H. Stemmler, 1964, "Static frequency changers with subharmonic control in conjunction with reversible variable speed ac drives," *Brown Boveri Rev.,* Vol. 122, No. 5, pp. 555-577.

J. J. Pollack, 1972, "Advanced pulsewidth-modulated inverter techniques," *IEEE Transaction on Industrial Applications.,* Vol. IA-8, No. 2, pp. 145-154.

S. R. Bowers, 1975, "New sinusoidal pulse width modulated inverter," *Proceedings of Industrial Electrical Engineerings.,* Vol. 122, No. 5, pp. 514-520.

R. M. Park, 1929, "Two-reaction theory of synchronous machines part I: Generalized method of analysis," *AIEE Trans.,* Vol. 48, pp. 716-730.

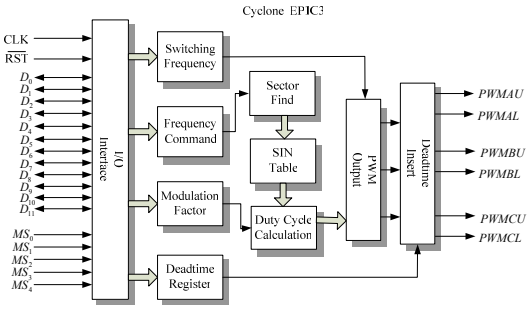


Figure 7: Pin assignment of the FPGA-based SVPWM control IC.

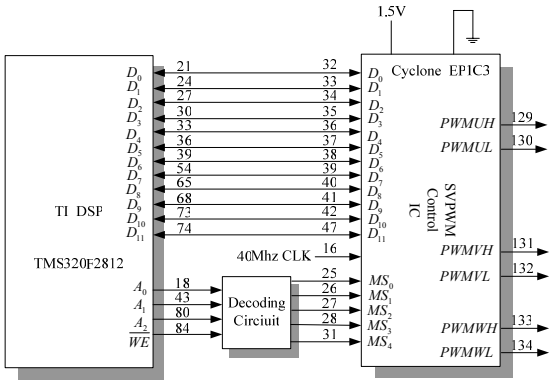
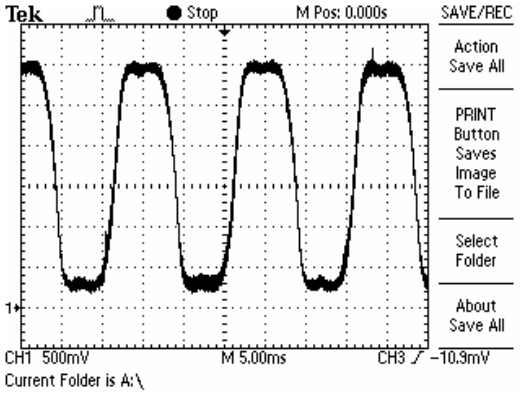
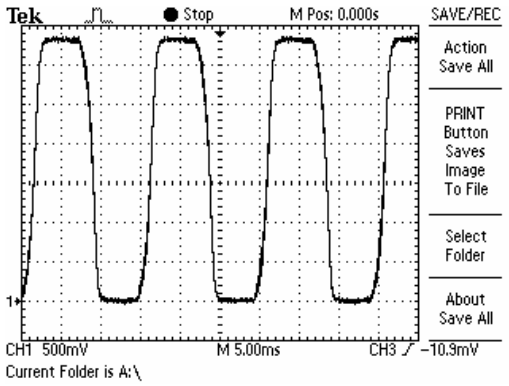


Figure 8: Circuit block diagram of the designedSVPWM IC interface with a sing-chip DSP TMS320F2812.

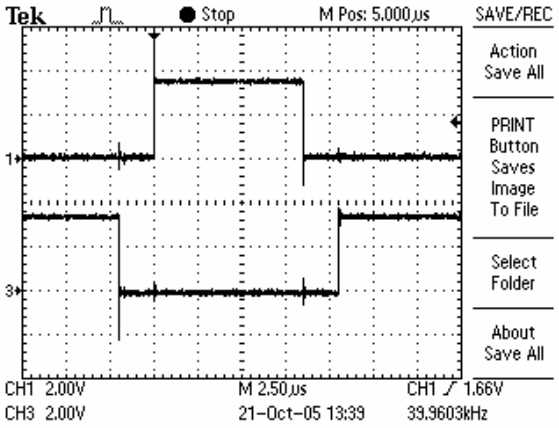


(a)

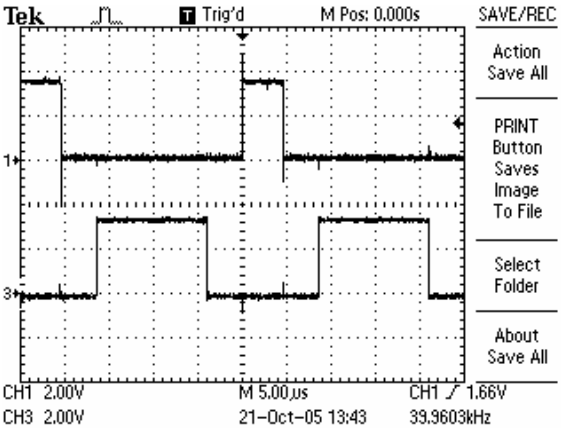


(b)

Figure 9: Experimental phase voltage wave form in the (a) linear region; (b) overmodulation region.

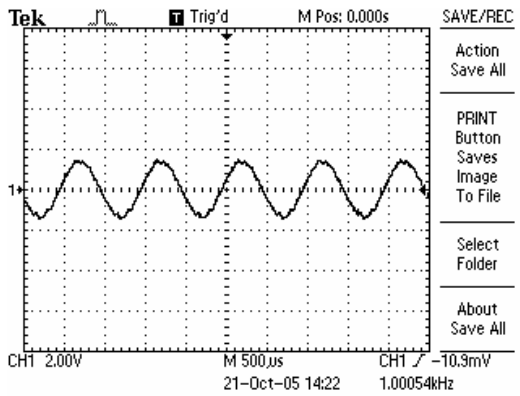


(a)

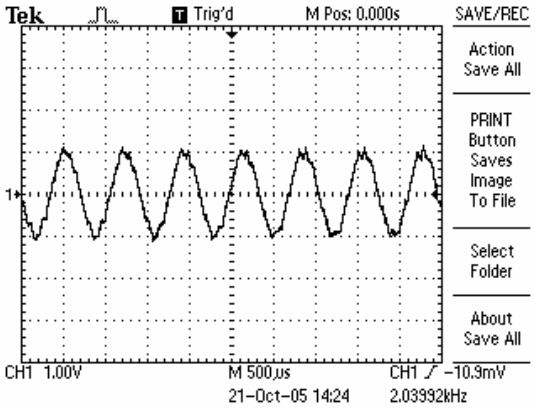


(b)

Figure 10: Experimental results of the programmable delay time with (a) 2 and (b) 4 us, respectively.

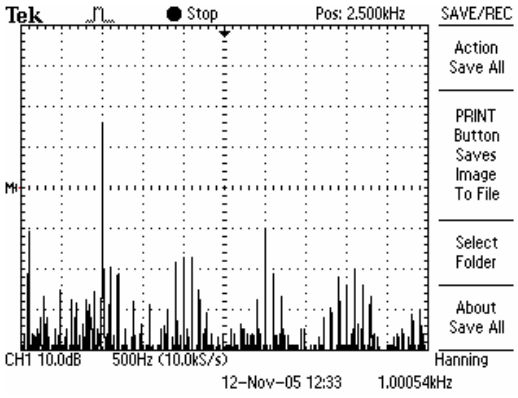


(a)

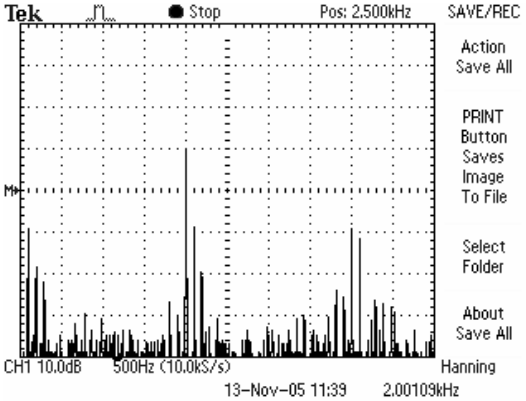


(c)

Figure 11: Experimental results of the SVPWM IC used in a PWM inverter drive with 1000- and 2000-Hz output: (a) and (c) are the phase currents.



(b)



(d)

Figure 11: Experimental results of the SVPWM IC used in a PWM inverter drive with 1000- and 2000-Hz output: (b) and (d) are the corresponding frequency spectrum.

A NEW UART CONTROLLER

Nonel Thirer
HIT, Holon Institute of Technology, Holon, Israel
Tirer_n@hit.ac.il

Radu Florescu
ORT Braude College, Carmiel, Israel
rflorec@ort.ac.il

Keywords: UART, Hamming Code, FPGA.

Abstract: The paper presents a new UART (Universal Asynchronous Receiver/Transmitter) controller which differs from traditional UARTs by providing a user defined data path width of 8 ,16, or 32 bits; by using a one bit error detection and correction algorithm (Hamming); and by permitting a large range of baud rates without the need of adding chips. By using the Hamming code, the communication throughput is increased, especially when a large data path width is defined. This new UART better responds to modern microprocessors' requirements, and was successfully implemented in an FPGA circuit.

1 INTRODUCTION

UARTs and USARTs (Universal Synchronous Asynchronous Receiver/Transmitter) devices are parallel-to-serial interfaces widely used in modern computer systems. They appear both as discrete components, as well as a part of a VLSI chip, or implanted as an intrinsic part of some microcontrollers and microprocessors.

In most UARTs and USARTs the parallel data path width is a maximum 8 bits. A detection error procedure, based on a parity bit, permits only to detect a one bit error. Also, for correction, an upper communication layer must be used (and data must be resent). In edition, the baud rate is constrained by the value of the external clock and an auxiliary programmable counter must be used to obtain various baud rates.

By improving these features, the new UART will be most suitable to modern microprocessors which have a large data path and a fast transmision rate.

2 THE NEW UART

This paper presents the possibility to improve the features of the traditional UARTs, by widening the data path; by adding a one bit error correction procedure; and by providing a larger range of baud rates.

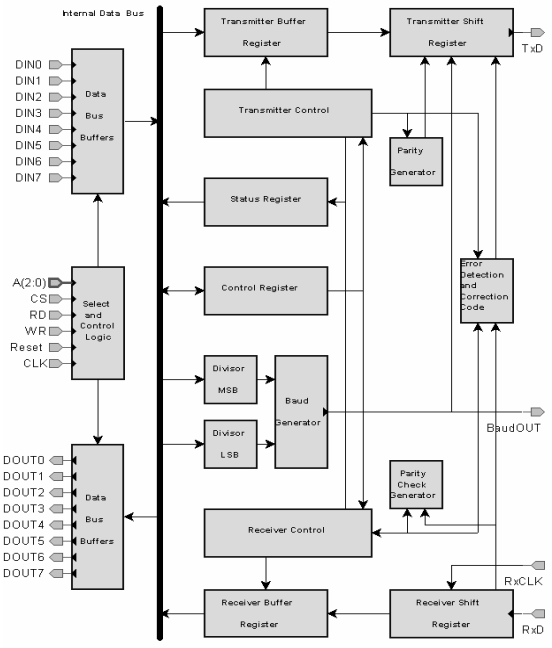


Figure 1: The UART block scheme.

The principial block scheme is presented in fig.1. (note: only 8 bits of parallel data are presented in this figure).

2.1 Data Encoding

The algorithm that encodes the data according to the Hamming code (Peterson, 1992) is simple:
- all bit positions that are powers of two (positions: 1, 2, 4, 8 and so on) are used as parity bits (P1, P2, P3, P4 and so on);
- all other bit positions (positions 3, 5, 6 and so on) are reserved for the data itself (D0, D1, D2 and so on)

Thus the data stream will be:
- P1, P2, D0, P3, D1, D2, D3, P4, D4, D5, D6, D7 for an 8 bits data;
- P1, P2, D0, P3, D1, D2, D3, P4, D4 … D10, P5, D11 … D15 for a 16 bits data;
- P1, P2, D0, P3, D1, D2, D3, P4, D4 … D10, P5, D11 … D25, P6, D26 … D31 for a 32 bits data.

Adding a start bit (‘0’) and a stop bit (‘1’), a total of 14 bits are necessary to transmit/receive a byte, 23 bits to transmit/receive a word, and 40 bits to transmit/receive a double word.

To transmit/receive the 8 bits data, a “classical” UART (working with only one stop bit) uses 3 supplementary bits (a 37.5% overhead).

In the proposed UART, the total number of supplementary bits will be 6 for an 8 bits data (a 75% overhead), 7 for a 16 bits data (44%), but only 8 for a 32 bits data (only a 25% overhead).

Thus, to transmit a byte, the classical UART must transmit only 11 bits and the proposed UART must transmit 14 bits, decreasing the communication throughput. But, to transmit a double word, the classical UART must transmit 44 bits, whereas the proposed UART transmits only 40 bits, and therefore increasing the communication speed.

This joins another feature of the new UART which ups communication speed, which is its error self-correction ability (which saves the need for resending data).

2.2 Error Detection and Correction

When a transmission error occurs, the traditional UART can only detect it. Correction must be made at a superior level of communication, and involves resending the data, a lengthy process.

It is to be noted that this is typical also to other communication protocols, such as the very popular USB communication standard, which uses a CRC method to detect the transmission error – a Not Acknowledge (NAK) response of the receiver if an error will appear, will also require to repeat the transmission.

In contradiction, the new UART’s multiple parity bits mechanism, as laid in the Hamming Code, allows it to correct, by itself, every single error. In this manner, no superior level of communication is required. Moreover, using a pipeline procedure, the receiver can correct the error in parallel with a new data receiving process, thus the communication throughput is not affected by the eventual transmission errors (Thirer, 2006).

It is also notable that Hamming code also allows detecting double errors, but can only correct a single error. Meaning, the method of error correction is not best suited for errors that come in bursts, but to situations in which errors are randomly occurring (and those are the most common in UART).

2.3 Baud Rate

To provide a large range of baud rates, a programmable 16 bits counter is included in the UART.

Thus the UART is able to work with various baud rates, from the transmission clock value TXCLK to TXCLK / 65536 without additional circuits.

3 AN FPGA IMPLEMENTATION

This UART was successfully implemented and tested in an ALTERA CYCLONE EP1C6Q240C8 FPGA chip

The UART controller includes a transmitter unit, a receiver unit and an internal counter.

The user can select the data path width (8, 16 or 32 bits), the error control method (parity bit or Hamming code) and also the baud rate (by the value of the counter acting as a clock divider).

Fig. 2 presents the state machine to generate and transmit the serial data stream by using the parity control bit or by using the Hamming control bits.

For generate the parity bits, a XOR based scheme was implemented in the transmitter unit. Evidently, the use of separate schemes for every parity bit will increase the speed, by a parallel computation, but a cheaper UART implementation can use a single parity generator scheme for a serial computation of the P1, P2, …P6 bits.

Fig. 3 presents the state machine for the receiver unit, including the parity check and the Hamming error check and fix.

Fig. 4 presents the block scheme of the receiver unit.

In the receiver unit, the parity check and fix section includes a parity generator (like the parity circuit of the transmitter unit) a six bit comparator (to check the received parity bits) and an adder

circuit to compute the position of the erroneous bit(if occurs). To fix this bit a simple inverter is used.

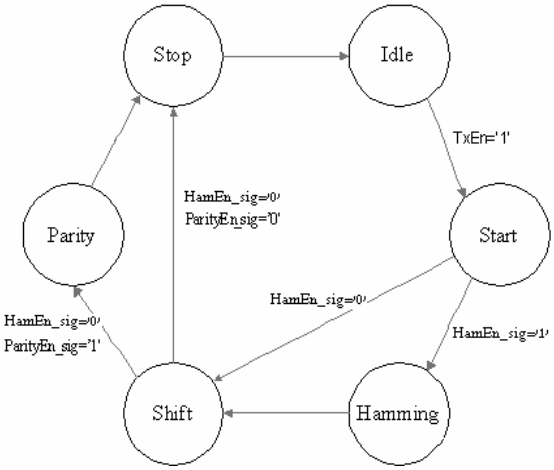


Figure 2: The Transmit Data Unit State Machine.

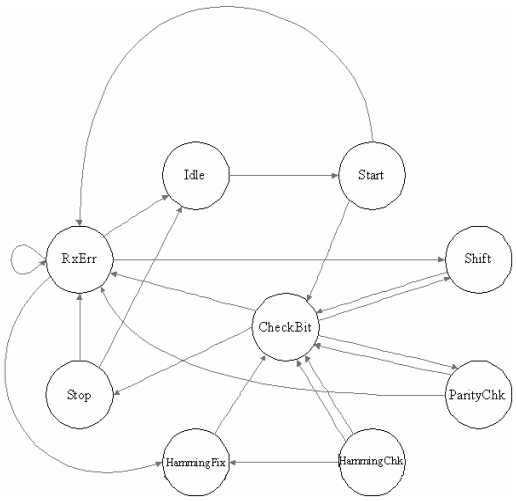


Figure 3: The Receiver Unit State Machine.

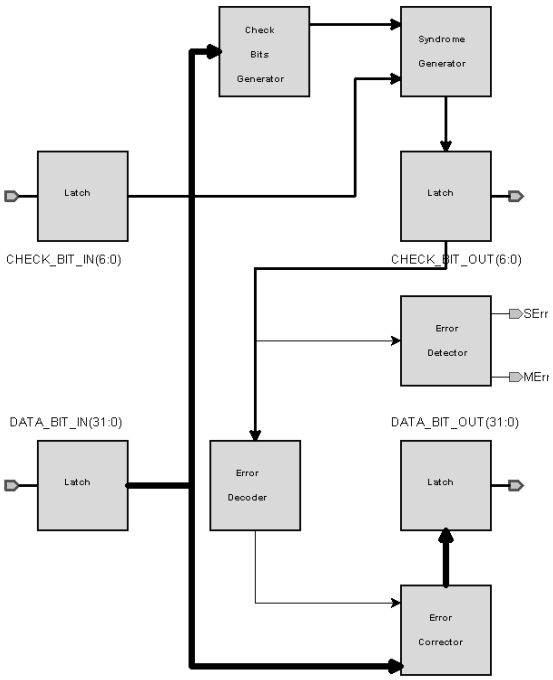


Figure 4: The Receiver Unit block scheme.

4 CONCLUSIONS

An improved UART controller is presented, compatible with the modern microprocessors and microcontrollers.

The implementation of an error correction algorithm permits to increase the transmit rate. The communication throughput is not affected by the eventual transmission errors.

For the new 64 bit microprocessors, a 64 bit data path can be easily implemented in this UART, by adding only a single Hamming parity bit.

REFERENCES

Durda, F., 1996. Serial and UART Tutorial. In *http://jamesthornton.com/freebsd/articles/serial-uart/*

Peterson, Wesley W., Weldon Jr, E.J , 1992, Error Corecting Codes, *The MIT Press*, 2nd Edition

Thirer, N., Efron, a.o. U., 2006. Improvement of FPGA Pipelines Implementation. In *PROC of SPIE Conf, "Optical Engineering and Instrumentation" (paper #6294-37)*

ADAPTIVE PREDICTIVE CONTROLLER APPLIED TO AN OPEN WATER CANAL

Luís Rato, Pedro Salgueiro
CITI-UE, Universidade de Évora, R. Romão Ramalho 59, 7000-671 Évora, Portugal
lmr@di.uevora.pt, pds@di.uevora.pt

João Miranda Lemos
INESC-ID/IST, R. Alves Redol, 9, 1000-029 Lisboa, Portugal
jlml@inesc.pt

Manuel Rijo
NuHCC, Universidade de Évora, R. Romão Ramalho 59, 7000 Évora, Portugal
rijo@uevora.pt

Keywords: Water canal, SCADA system, PLC, data acquisition, adaptive control, predictive control.

Abstract: This paper concerns to the application of adaptive control to a large scale water canal experimental plant. Water canals are complex spatially distributed systems which aim at distributing water either for irrigating, or domestic, or industrial purposes. In this paper a predictive adaptive control algorithm (MUSMAR) is applied to a large scale experimental water canal prototype. The experimental facilities with a fully instrumented canal, a PLC network and a SCADA system, are briefly described. This paper describes the developed software module and the MUSMAR control algorithm. Finaly, Some experimental results obtained in the experimental water canal, are presented.

1 INTRODUCTION

Water distributing systems are increasingly important as fresh water scarcity is becoming a critical issue in many places worldwide. In this increasing water scarcity situation an efficient management of water canals, minimising water losses, is an obviously important issue. Nevertheless, this management task brings conflicting goals - minimise water loss *vs.* Quality of Service. On one side, users demand more and more flexibility on water withdrawing from canals. On the other side, the use of the traditional pre-scheduled water turns may attain a very low level of water losses but at cost of users QoS. Thus, modern water canals with advanced control techniques may have and an important role on the management of these conflicting goals.

Two strategies of upstream automatic control are applied: local upstream control and distant upstream control. Local control is the most practical way to introduce automatic control on existing canals, since every equipment can be concentrated an one place. Nevertheless, distance upstream control as well as remote supervision are promising approaches.

Adaptive control techniques are most suited to situations where the dynamic behaviour is unknown or slowly changing. Nevertheless, the adaptive predictive algorithm MUSMAR has also shown to cope well with incomplete order modelling and minor nonlinearities. Thus MUSMAR has been tested and successfully applied in several experimental processes ranging from distributed solar power plants (Coito et al., 1997)(Rato et al., 1997) to Internet traffic control (Costa et al., 2002).

This control approach is implemented through a developed software package that communicates to the supervisory control and data acquisition (SCADA) system which is connected to a programmable logic controller(PLC) network.

2 WATER CANAL DESCRIPTION

The experimental water canal used in this work is at the NuHCC (Núcleo de Hidráulica e Controlo de Canais) of the University of Évora, Fig. 1.

The canal has a trapezoidal cross section geometry, and is constituted by four pools of roughly 40 m each, resulting in a 145 m long instrumented canal, plus a traditional water canal which completes a closed water circuit. The canal inlet water flow is defined by an electrical controlled MONOVAR valve. The flow along the four pools may be controlled by three sluice gates and there is a water off-take up-

Figure 1: Experimental water canal at the University of Évora.

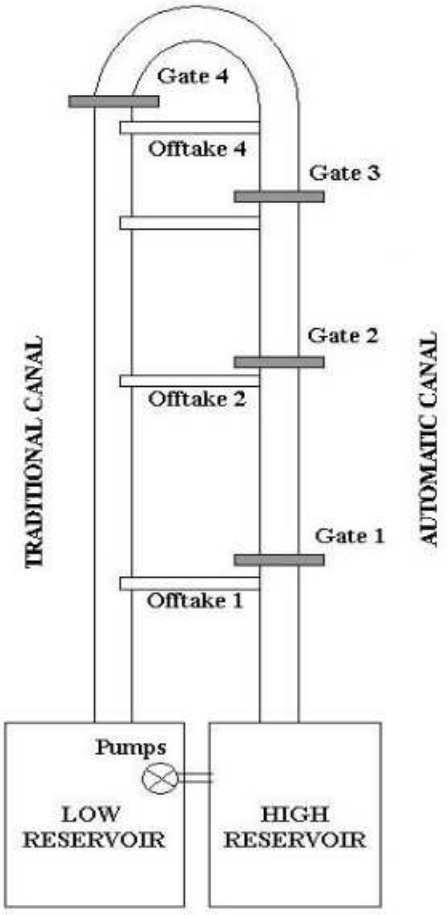


Figure 2: Schematic diagram of the experimental water canal at the University of Évora.

stream of each gate. This off-take is equipped with a flow meter and an electrical butterfly valve and discharges into the traditional return canal. Concerning sensors, there are three float and counter-weight level sensors in stilling wells for each pool - one at each end and one in the middle of the pool. Figure 2 shows a schematic diagram of the experimental plant. This facility is described in more detail in (Ratinho et al., 2002).

It should be noted that the dynamic behaviour of this type of plant is modelled by the Saint-Venant equations which are nonlinear partial differential equations which can be linearised for small variations around stationary values. Thus, this plant belongs to a class of distributed parameter plants with transport phenomena, such as highway traffic, distributed solar plants, and boiler circuits of thermal power plants that have been studied with success in the scope of advanced control algorithms, as adaptive and predictive control techniques(Silva et al., 2003; Marques and Silva, 2005).

2.1 Data Acquisition and Supervision System

The experimental plant is equipped with a network of 6 PLCs. Five local PLCs (one for each sluice gate or inlet valve) and one central master PLC. The data acquisition and analog-to-digital conversion are performed locally at each PLC. These are interconnected by a MODBUS network to the master PLC, which communicates to the SCADA computer by a serial port RS232 interface.

The SCADA system is build over the WIZCON environment, and presents a user friendly graphical interface to command the process as well as to observe the evolution of measured variables over time. The SCADA also has a DDE interface which allows the communications between the SCADA and external applications.

This system is described in more detail in (Almeida et al., 2002).

2.2 Control Algorithm Software Package

Due to the complexity of MUSMAR algorithm, it is impracticable to implement local control directly in the local PLCs. Moreover, although the SCADA may implement a predefined set of controllers, it has little support to develop general algorithms. Thus a software package was developed in C language in order to extend the control capabilities of the SCADA system and implement the MUSMAR algorithm has an external process.

Once the SCADA system has a Dynamic Data Exchange(DDE) communication interface, this was the chosen process of interaction between the external process (the controller) and the SCADA system.

The initialisation, reading and writing of the variables which define the state of the process can be performed through the following functions

```
DDEInit("WIZCON", "GATE")
DDERequest(char var[])
DDEPoke(char []var, char []valor)
```

These functions define the basic application programming interface (API).

However, while the reading needs usually just one call to DDERequest(), to define a command into the actuators, several calls of DDEPoke(...) are usually necessary, stating the control mode of one or more (cascade) loops, and defining the variable value.

Thus, we have defined a two layer API: the base API (dde-base.c) which communicates to the SCADA through DDE and defines DDERequest(...) and DDEPoke(...); and a second layer with a more user friendly API (scada-api.c).

This second layer defines a set of 55 functions to read and write in or from each of the plant sensors and actuators. The API implementation for the case of a sluice gate is presented above.

Sluice Gates Api Implementation

We consider in the following, the case of gate API to gate number 1, which is connected to PLC2. The sluice gate control mode is defined by the tag "MODE_MATIC_A2", which can take values from 1 to 4 corresponding to direct control of the actuator; local control of sluice gate position; local control of upstream level; and local control of downstream level. When MODE_MATIC_A2 is set to 1 it is possible to command the gate directly with three commands: open, close, and stop.

If the gate is closing and the desired option is to completely open the gate it is necessary to set MODE_MATIC_A2=1 to set the direct control mode; then an order to inhibit the closing command, MATIC_CLOSE_OUT_A2=0, and only then should be sent the order to open the gate MATIC_OPEN_OUT_A2=1, as shown below.

```
void open_gate_1(){
  DDEPoke("MODE_MATIC_A2", "1");
  DDEPoke("MATIC_CLOSE_OUT_A2", "0");
  DDEPoke("MATIC_OPEN_OUT_A2", "1");
}
```

Similar procedures are developed to interact with other actuators.

Api Definition

In the API definition is listed below, "N" stands for the number of the PLC that controls the device.

```
void open_gate_N()
void close_gate_N()
float level_gate_N()
void set_level_gate_N(int level)
void close_monovar()
void open_monovar()
void set_in_flow(int flow)
void set_monovar_level(float level)
float flow_monovar_in()
float level_monovar()
float flow_valve_gN()
float level_valve_gN()
void close_valve_gN()
void open_valve_gN()
void set_valve_flow_gn(float flow)
void set_valve_level_g1(float level)
int level_upstream_canal()
int level_middle_gN()
int level_upstream_gN()
int level_downstream_gN()
```

Other Package Functions

Along with the API, the developed package provides also: a simple command line; a text oriented output interface; data logging; and a timing function. These are functions that may be naturally adapted to the experiment to be performed.

Development Environment

This package was developed in a MinGw environment. This is a set of free open-source tools for the Windows operating system, and includes among others: a port of the *GCC* compiler, a *bourne shell* compatible environment (MSYS) and a *makefile* utility.

3 MUSMAR ADAPTIVE PREDICTIVE CONTROLLER

In this paper, experimental results are presented on the application of MUSMAR, a predictive adaptive control algorithm for which there is evidence of robustness against plant unmodelled dynamics(Mosca et al., 1989), and have been tested in large number of experimental plants.

At the beginning of each sampling interval, recursively perform the following steps: **1.** Sample the process output at time t, compute the tracking error. **2.** Using Recursive Least Squares, update the estimates of the parameters in a set of predictive models. **3.** Apply to the plant the control given by

$$u(t) = F's(t) + \eta(t) \quad (1)$$

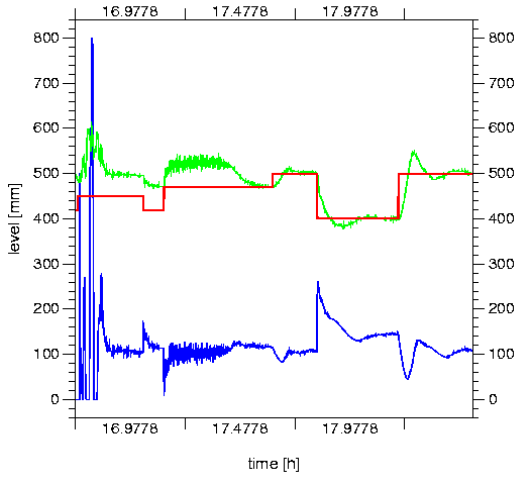


Figure 3: Experimental results. Local upstream control. Water level (mm), reference (mm), gate position (mm), time (h).

where η is a white dither noise of small amplitude, such that and F is the vector of controller gains, computed from the estimates of the corresponding predictive models by the optimization of a cost function across a predefined horizon T.

An integral effect has been also considered in parallel with MUSMAR. This algorithm has been implemented in C and linked with the software package presented above. A detailed description of MUSMAR is presented in (Mosca et al., 1989).

4 EXPERIMENTAL RESULTS

The following results were obtained at the experimental canal with MUSMAR controller in January 2007.

In the experiment the control structure is a local upstream one. The sampling time was set to 5 s, the controlled variable is the level upstream of gate 2, the manipulated variable is the position of gate 2. The inlet flow of the canal was locally controlled to 35 l/s, off-take valves were closed and gate 1, 3, and 4 were opened.

Experimental results are shown in Fig.3. After the startup the gains converge and the algorithm follows the reference, although with a significant static error. At instant $17,2$ the integral gain was set to 0.05 eliminating the static error.

5 CONCLUSIONS

An adaptive predictive control algorithm and an API software package were implemented, and tested in an experimental process plant. The results show the applicability of advanced control algorithms in the context of water canal systems. Instrumented canal plants with centralised control and supervision are essential to the application of complex control algorithms, which are impractical to implement on local PLCs. As future work the inclusion of a priori information (loading initial gains and initial covariance matrix) is a promising step as it has been observed in other applications to be an important issue in order to apply the MUSMAR algorithm in production environment.

ACKNOWLEDGEMENTS

This work has been supported under project FLOW - POSC/EEA-SRI/61188/2004.

REFERENCES

Almeida, M., Figueiredo, J., and Rijo, M. (2002). Scada configuration and control modes implementation on an experimental water supply canal. In *MED'2002, 10th Mediterranean Conference on Control and Automation, Lisbon, Portugal*.

Coito, F., Lemos, J., Silva, R. N., and Mosca, E. (1997). Adaptive control of a solar energy plant: exploiting accessible disturbances. In *Int. Journal of Adaptive Control and Signal Processing*, volume 11, pages 327–342.

Costa, B., Nunes, M. S., and Lemos, J. (2002). Adaptive predictive control of ip traffic. In *MED'2002, 10th Mediterranean Conference on Control and Automation, Lisbon, Portugal*.

Marques, M. C. and Silva, R. N. (2005). Traffic simulation for intelligent transportation systems development. In *IEEE Intelligent Transport Systems Conference, Viena, Austria*.

Mosca, E., Zappa, G., and Lemos, J. M. (1989). Robustness of multipredictive adaptive regulators: Musmar. In *Automatica*, volume 25, pages 521–529.

Ratinho, T., Figueiredo, J., and Rijo, M. (2002). Modelling, control and field tests on an experimental irrigation canal. In *MED'2002, 10th Mediterranean Conference on Control and Automation, Lisbon, Portugal*.

Rato, L., R. N. Silva, J. L., and Coito, F. (1997). Multirate musmar cascade control of a distributed collector solar field. In *ECC'97, European Control Conference, Brussels, Belgium*.

Silva, R. N., Lemos, J., and Rato, L. (2003). Variable sampling adaptive control of a distributed collector solar field. In *IEEE Trans. Control Systems Technology*, volume 11(5), pages 765–772.

TRACKING PLASMA ETCH PROCESS VARIATIONS USING PRINCIPAL COMPONENT ANALYSIS OF OES DATA

Beibei Ma, Seán McLoone and John Ringwood
Department of Electronic Engineering, National University of Ireland Maynooth, Maynooth, Ireland
beibei.ma@eeng.nuim.ie, sean.mcloone@eeng.nuim.ie, john.ringwood@eeng.nuim.ie

Keywords: Semiconductor manufacturing, plasma etching, metal etching, optical emission spectroscopy (OES), principal component analysis (PCA), batch processing.

Abstract: This paper explores the application of principal component analysis (PCA) to the monitoring of within-lot and between-lot plasma variations that occur in a plasma etching chamber used in semiconductor manufacturing, as observed through Optical Emission Spectroscopy (OES) analysis of the chamber exhaust. Using PCA, patterns that are difficult to identify in the 2048-dimension OES data are condensed into a small number of principle components (PCs). It is shown, with the aid of experimental data, that by simply tracking changes in the directions of these PCs both inter-lot and intra-lot patterns can be identified.

1 INTRODUCTION

Modern day semiconductor manufacturing is a highly competitive business in which companies are required to produce vast quantities of reliable high performance integrated circuits (ICs) at low cost. As such, close monitoring and tight control of hundreds of complex process steps are needed to maintain production standards and high product throughput.

In this context we focus on plasma etching of semiconductor wafers, an important process step in the manufacture of many ICs (Sugawara, 1998). A typical reaction ion etching (RIE) chamber is illustrated in Fig. 1. Gas is pumped into the chamber under vacuum and ionised using a high power Microwave (MF) source to create a plasma. A radio frequency (RF) electromagnetic field accelerates the resulting ionised species towards the electrode, where they interact both chemically and physically with the wafer, etching away the exposed surface. The etch rate and profile obtained are determined in a complex and nonlinear fashion by the plasma chemistry and energy as well as several process variables including gas flow rates and RF power.

Monitoring the chemistry of the plasma in the chamber can be achieved using Optical Emission Spectroscopy (OES) (Splichal *et al.*, 1987). In the plasma chamber considered in this study the OES data is collected for the exhaust plasma leaving the chamber using a 2048 wavelength OES sensor (170nm to 875nm) with a sampling interval of 0.75s. Using this setup OES data was collected for 17 lots of 24 wafers, with each waver undergoing a two step etch process lasting 45s. A sample OES data set for a single wafer is shown in Fig. 2.

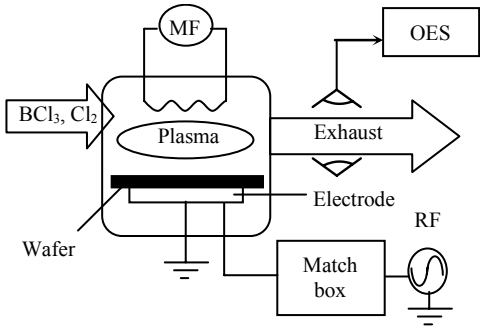


Figure 1: Diagram of a plasma etching chamber.

With the OES footprint of each wafer having dimensions of 60 × 2048, direct visualisation and monitoring of variations in the plasma chemistry across wafers and across lots is impractical. Fortunately, optical emission spectra are inherently highly redundant making it possible to achieve substantial data compression using Principle Component Analysis (PCA) techniques without loosing valuable information on plasma changes. In this paper we show that simply monitoring changes

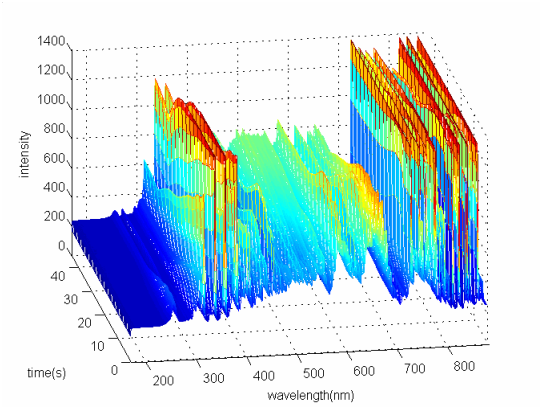


Figure 2: A plasma etch OES data set for a single wafer.

in the directions of the principle component loading vectors, computed on either a wafer-by-wafer or lot-by-lot basis, is sufficient to detect valuable information on process trends that are not immediately apparent when looking at the OES data as a whole.

2 PRINCIPAL COMPONENT ANALYSIS

Principal component analysis (PCA) is a linear multivariate data projection technique widely used for data compression and analysis (Jackson, 1991). It provides a means of generating low dimension representations of high dimension datasets while retaining the maximum amount of information.

2.1 Definition

PCA is a method of writing a matrix $\mathbf{X}$ of rank r as a sum of r matrices of rank 1, where the rank 1 matrices are expressed as outer products of two vectors, a score $\mathbf{t}_i$ and a loading $\mathbf{p}_i$ (Jackson, 1991)

$$\mathbf{X} = \sum_{i=1}^{r} \mathbf{t}_i \mathbf{p}_i^{\mathrm{T}} \qquad (1)$$

The loading vectors, $\mathbf{p}_i$, are eigenvectors of the matrix $\mathbf{X}^{\mathrm{T}}\mathbf{X}$, that is

$$(\mathbf{X}^{\mathrm{T}}\mathbf{X})\mathbf{p}_i = \lambda_i \mathbf{p}_i \qquad (2)$$

where λ_i is the eigenvalue associated with the i^{th} eigenvector $\mathbf{p}_i$. The loading vectors $\mathbf{p}_i$ describe the principal directions of variation in **X,** are orthogonal to each other:

$$\mathbf{p}_i^{\mathrm{T}} \mathbf{p}_j = 0, \quad \forall i \neq j \qquad (3)$$

and have unit length, while the eigenvalues indicate the amount of variance represented by each direction. For a given $\mathbf{X}$ and $\mathbf{p}_i$, the corresponding score vector $\mathbf{t}_i$ is given by:

$$\mathbf{t}_i = \mathbf{X}\mathbf{p}_i \qquad (4)$$

A principle component (PC) model of $\mathbf{X}$ is then obtained by selecting the components $(\mathbf{p}_i, \mathbf{t}_i)$ with the largest eigenvalues to represent it. When data redundancy is high two or three PCs are often sufficient to obtain a good model.

Essentially, PCA projects a high dimensional data space onto a lower dimensional sub-space where the axes are the PC loading vectors and the coordinates of the data the PC score vectors.

Singular Value Decomposition (SVD) can be used to calculate all r principal components in one step. Alternatively, the nonlinear iterative partial least squares (NIPALS) algorithm (Geladi, 1986) can be used to calculate them one at a time in order of significance.

2.2 Multi-way PCA (MPCA)

Since batch process data is usually arranged in a 3-way matrix (batch-variable-time), it must be unfolded into a 2-way matrix in order to apply PCA. This is known as Multi-way Principal Component Analysis (MPCA) and was first introduced by Wold *et al.* (1987). There are several ways to unfold a 3-way matrix. In this paper, we choose to unfold the data along wavelength direction (Fig. 3), because we are interested in tracking process changes over time.

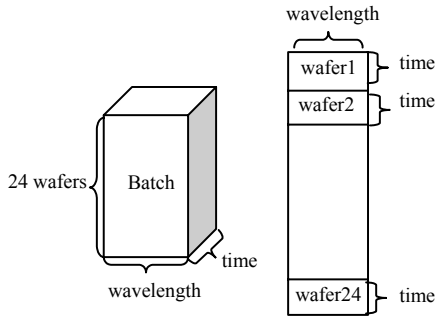


Figure 3: Unfolding of the 3-way OES data blocks. Each block corresponds to a lot of 24 wafers.

2.3 Monitoring PC-loadings

If PCA is performed on the OES data as a whole process trends can only be observed by monitoring the time evolution of the scores. However, if PCA is applied on a wafer-by-wafer or lot-by-lot basis very effective monitoring of process variation can be achieved by tracking the changes in the directions of the PC loadings. Changes can be expressed either in terms the angle between loadings or the magnitude of the vector difference between them as illustrated in Fig. 4. The angle θ (in radians) is given by

$$\theta = arccos\left(\frac{\mathbf{v}_1\mathbf{v}_2^{\mathrm{T}}}{|\mathbf{v}_1||\mathbf{v}_2|}\right) \qquad (5)$$

while the magnitude of the vector difference φ is simply defined as

$$\varphi = |\Delta\mathbf{v}| = |\mathbf{v}_1 - \mathbf{v}_2| . \qquad (6)$$

Since, by definition, loading vectors are unit length it follows that for small θ the two measures are approximately equivalent, i.e. $\varphi \approx \theta$.

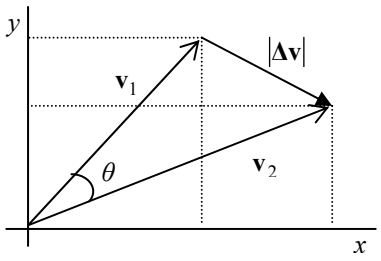


Figure 4: Measuring changes in loading vector directions.

3 OES DATA ANALYSIS

3.1 Data Pre-processing

Data pre-processing is an essential first step in PCA analysis. Variables need to be appropriately scaled and irrelevant or corrupted measurements removed if valid and interpretable results are to be obtained. In this study the following pre-processing step were performed on the OES data: (1) Data segments corresponding to non-etch periods at the start and end of each etch cycle were removed; (2) Saturated wavelengths were omitted and; (3) Wavelength intensities were scaled to have zero mean.

3.2 Lot-by-lot Analysis

Having unfolded the OES data as indicated in Fig. 3, analysis by PCA can be performed by treating each lot of 24 wafers as a single data matrix. We will refer to the resulting PCs as lot-PCs, consisting of lot-PC loadings and lot-PC scores. The variance explained by the first three lot-PCs is plotted as a function of lot number in Fig. 5. This shows that across all lots the first three principal components can explain over 99% of the plasma variation captured by the OES data. In fact the first PC captures over 85% of the data variation observed across all 2048 wavelengths.

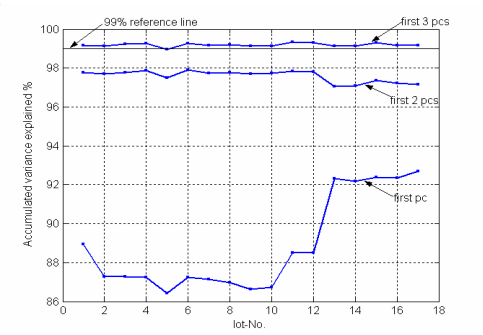


Figure 5: Accumulated variance explained by the first three lot PCs.

A closer look at Fig. 5 shows that a jump occurs in the variance explained by lot-PC_1 at lot 13. Analysis of the variation in the direction of lot-PC_1 across lots (Fig. 6) reveals that this is linked to a significant change in the orientation of lot-PC_1 from lot 13 onwards. Following investigation it was determined that the plasma change was as a result of a small drift in the flow rate of a cooling gas applied to the backside of the wafers during etching, a change that was not detected by the existing plasma chamber process monitoring schemes.

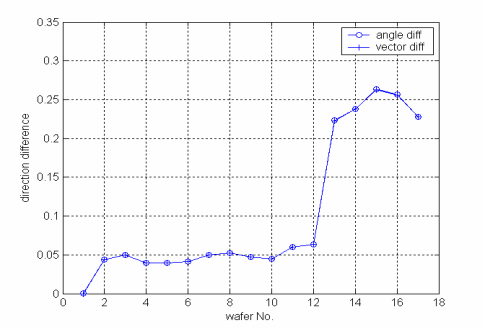


Figure 6: Variation in lot-PC_1 (loading) direction across lots (with respect to lot 1).

3.3 Wafer-by-wafer Analysis

Here, we simply perform PCA analysis on individual wafer OES data sets and compare the patterns across wafers. This allows us to explore the variation that takes place within lots.

Fig. 7 shows the variation in wafer-PC_1 direction over all the wafers with lot-PC_1 from lot 1 taken as a reference. The plasma change at lot 12 observed in the lot-PC analysis is clearly present in this data as well, as is a small offset during the first lot.

Large spikes are evident throughout Fig. 7. These occur at the first wafer in each lot. This is highlighted in Fig. 8 which shows a zoomed in view of Fig. 7 covering two lots. These sharp changes were attributed to changes in the absorption characteristics of the plasma chamber wall as a result of a cleaning cycle that is performed between lots. While a dummy etch cycle is performed following each clean cycle to counter this affect, it is clear from Fig. 8 that cleaning still has a significant impact on plasma characteristics for the first (and to a lesser extent) the second wafer etch of each lot.

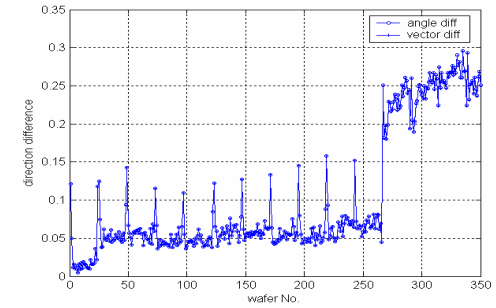


Figure 7: Variation in wafer-PC_1 direction across all wafers (using lot-PC_1 from lot 1 as a reference).

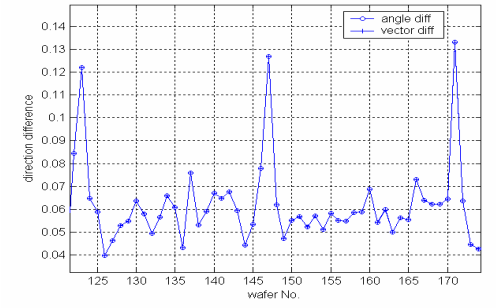


Figure 8: Variation in wafer-PC_1 direction over two lots.

3.4 Score Patterns

As an illustration of the data compression and pattern visualisation capabilities of PCA the score patterns generated by the OES data for all the wafers in lot 9 are plotted in Fig. 9. Here, the first three lot-PC loadings from lot 8 were used as a reference PC model and the PC-scores for each wafer calculated according to Eq. (4). It is easy to see that the evolution of the OES data for the first and second wafers is substantially different from the remaining wafers, as predicted by the wafer-PC loading analysis.

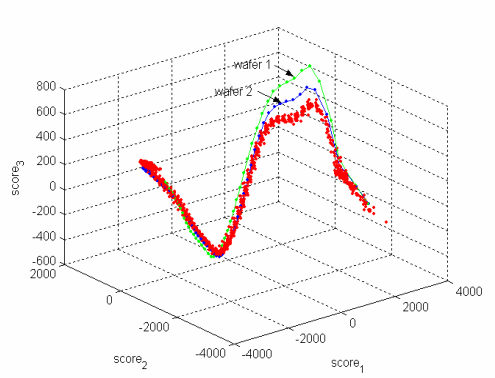


Figure 9: The first three scores of all the wafers in lot 9.

4 CONCLUSIONS

In this paper we have demonstrated how monitoring changes in PC directions can be a useful tool in revealing patterns contained in the high dimensional data sets generated from OES analysis of wafer etch plasma chambers.

ACKNOWLEDGEMENTS

The financial support provided by Enterprise Ireland is gratefully acknowledged.

REFERENCES

Sugawara, M., 1998. Plasma Etching: Fundamentals and applications, Oxford University Press, New York.

Splichal, M., Anderson, H., 1987. Application of Chemometrics to Optical Emission Spectroscopy for Plasma Monitoring. *Proc. SPIE*, **2**, pp. 189-203.

Jackson, J. E., 1991. A User's Guide to Principal Components, Wiley Interscience Press, New York.

Geladi, P., Kowalski, R. B., 1986. Partial least-squares regression: a tutorial. *Analytica Chimica. Acta.*, **185**, pp.1-17.

Wold, S., Geladi, P., Esbensen, K., Ohman, J., 1987. Multi-way principal components and PLS analysis. *Journal of Chemometrics*, **1**, pp. 41-56.

DESIGN OF AN AUTOMATED FIXED BED REACTOR USED FOR A CATALYTIC WET OXIDATION PROCESS

A. El Khoury, B. Bejjany
Laboratoire de Chimie Industrielle – Génie des Procédés (EA21), Cnam, 2 rue Conté, Paris III, France
ea21@cnam.fr

M. Debacq, A. Delacroix
Laboratoire de Chimie Industrielle – Génie des Procédés (EA21), Cnam, 2 rue Conté, Paris III, France
ea21@cnam.fr

Keywords: Wet Air Oxidation, WAO, Data acquisition, Intelligent sensor module, Regulation, Supervision, Monitoring.

Abstract: Treatment of polluted industrial wastes is one of the challenging research topics that occupy an important position in various chemical processes. Wet Air Oxidation (WAO) is one of the emerging processes suited for the treatment of special aqueous wastes. The system consists of an oxidation in the liquid phase of the organic matter by molecular oxygen at high temperature (200-325°C) and high pressure (up to 175 bar). It is an enclosed process with a limited interaction with the environment as opposed to incineration. In this paper, we will discuss the setup and the design of an automated fixed bed reactor used for wet oxidation of various types of wastes. The system is controlled by a set of intelligent sensor modules used for data acquisition. Regulation loops integrated within the sensor modules had been developed in order to control the gas flow, the reactor temperature and the liquid sampling part. The process supervision and monitoring had been achieved through the deployment of a SCADA software application. The graphical interface developed for this purpose monitors the major parts of the process.

1 INTRODUCTION

The identification of highly refractory and non-biodegradable organic pollutants in wastewater, especially coming from the chemical and petrochemical industry, has challenged the conventional wastewater treatment such as incineration or biological abatement. There is a clear need to test and set-up an emerging alternative technology that can deal with highly concentrated and/or toxic non-biodegradable organic water pollutants. However, it seems impossible in the close future to dispose of one universal method able to destroy all of the detected pollutants at an acceptable cost (Masende, 2003). Therefore, Wet Air Oxidation (WAO) is an efficient process by which organic pollutants can be transformed by oxidation under high pressures (50-250 bar) and high temperatures (200-325°C), into carbon dioxide and water (Mishra, 1995). The process can be performed under milder conditions (temperatures and pressures) by using a homogenous or heterogeneous catalyst. Catalytic Wet Air Oxidation (CWAO) is thus an attractive process for wastewater treatments of toxic pollutants such as phenol, pesticides, methyl *tert*-butyl ether (MTBE) and their intermediate oxidation compounds (Pintar, 1992).

Several studies with noble metal catalysts, mainly Ru and Pt supported on carbon, Al_2O_3, TiO_2 and CeO_2 have revealed their stability and capacity to destroy organic pollutants (Imamura, 1988). In contrast to platinum, ruthenium was found to be an active metal during the oxidation of acetic acid, which is very refractory. Comparison of Ru/CeO_2 and Ru/TiO_2 showed that titanium oxide was more stable in acetic and oxidizing medium, but the loading of Ce on the catalyst significantly changes the surface properties resulting in a better dispersion of the noble metals. Thus, Ruthenium and Cerium metals supported on alumina are considered to be stable, accurate and cost effective catalysts (Oliviero, 2000). There are only some tens of industrial plants in the world and very few documents are available for the scientific design of

such processes due to their complexity and the delicacy needed for their proper operation (Debellefontaine, 1999). Therefore, in this work we will show the essential techniques and equipments allowing us to control and monitor a pilot scale reactor designed for the wet oxidation of organic pollutants.

After presenting the system architecture and the communication interface, we will attempt to emphasize on the use of intelligent sensor modules in order to adequately control the reactor temperature and pressure as well as the gas flow and the liquid outlet.

2 SYSTEM ARCHITECTURE AND COMMUNICATION

The installation (Figure 1) consist of an L-316 tubular fixed-bed reactor (7.6 cm internal diameter and 70 cm in length), which is placed in the center of an oven implementing an electrical resistor controlled by a PID controller. The solution is introduced to the reactor by a high-pressure pump at a flow rate ranging from 1 to 10 $cm^3.min^{-1}$. The catalyst is placed between two layers of glass beds in the reactor. The oxygen is directly fed from a high-pressure bottle whereas a gas flow indicator and controller (FIC – Brooks) controls its flow rate. The effluent of the reactor passes through two condensers and a gas-liquid separator. The gas phase is released in the hood after passing through a gas flow indicator (FI – Brooks) and the liquid phase is stored in a tank whereas a level indicator (LI – Bamo) controls a regulation valve (LV - Samson) prior to liquid evacuation. A backup pressure regulator (PIC) placed at the gas outlet maintains a stable pressure inside the system.

Sensors and actuators are plugged into a set of four intelligent sensor modules (ISM112 – Gantner) interconnected through an RS485 field bus. An RS232/RS485 converter enables the supervision station to communicate with the sensor modules using the Modbus RTU protocol. The intelligent sensor module supports measuring methods with 2-, 3-, and 4-wire technique and measuring methods with 4- and 6-wire bridge connection. Consequently, the most varying measurement tasks can easily be solved by means of the different analog inputs and in combination with the force output, which provides the local power supply for the transducers. The module can simultaneously take up and process sensor signals from several heterogeneous sensors. As many sensors can be connected as there are analog and digital signal inputs and outputs available. With the ISM112 these are 6 sensors at the most, 4 analog and 2 digital sensors. The RS485 interface permits the simultaneous connection and operation of a maximum of 32 bus users per segment. Among analog and digital signal processing; the intelligent sensor module can handle a controller variable by which a sensor variable can be monitored for a definable set value. Deviations of the sensor variable's value will be corrected depending on the set function of the controller (PID-controller) and will then be assigned to the controller variable. This corrected value can be assigned to an analog output and then be used to influence the input signal by a corresponding connection. Accordingly, we were able to control and monitor most of the system parameters in order to boost and optimize the reaction conditions. A set of five thermocouples indicates the temperatures at different levels of the process, especially at the center of the reactor where the temperature had been adequately controlled and monitored. Pressure indicators monitor the system global pressure required for the reactor proper operation. Possible fluid leakage can be detected through pressure drops inside the system. Gas flow is controlled by an algorithm set by the manufacturer whereas the intelligent sensor modules directly control the reactor temperature and the regulation valve through a set of regulation parameters defined by the *Ziegler-Nichols* method.

3 TUNING A PID CONTROLLER

The first step in the design strategy is to install and tune a PID controller (Tan, 2006). The ideal continuous PID controller returns the controller output u, as given by equation (1), where K_p is the proportional gain, T_i is the integral time, T_d the derivative time, and e the error between the reference (*ref.*) and the process output (y).

$$u = K_p\left(e + \frac{1}{T_i}\int_0^t e.dt + T_d\,\frac{de}{dt}\right) \qquad (1)$$

We are concerned with small sampling periods T_s, the equation may be approximated by a discrete approximation. Replacing the derivative term by a backward difference and the integral by a sum using rectangular integration, an approximation may be given by the equation (2).

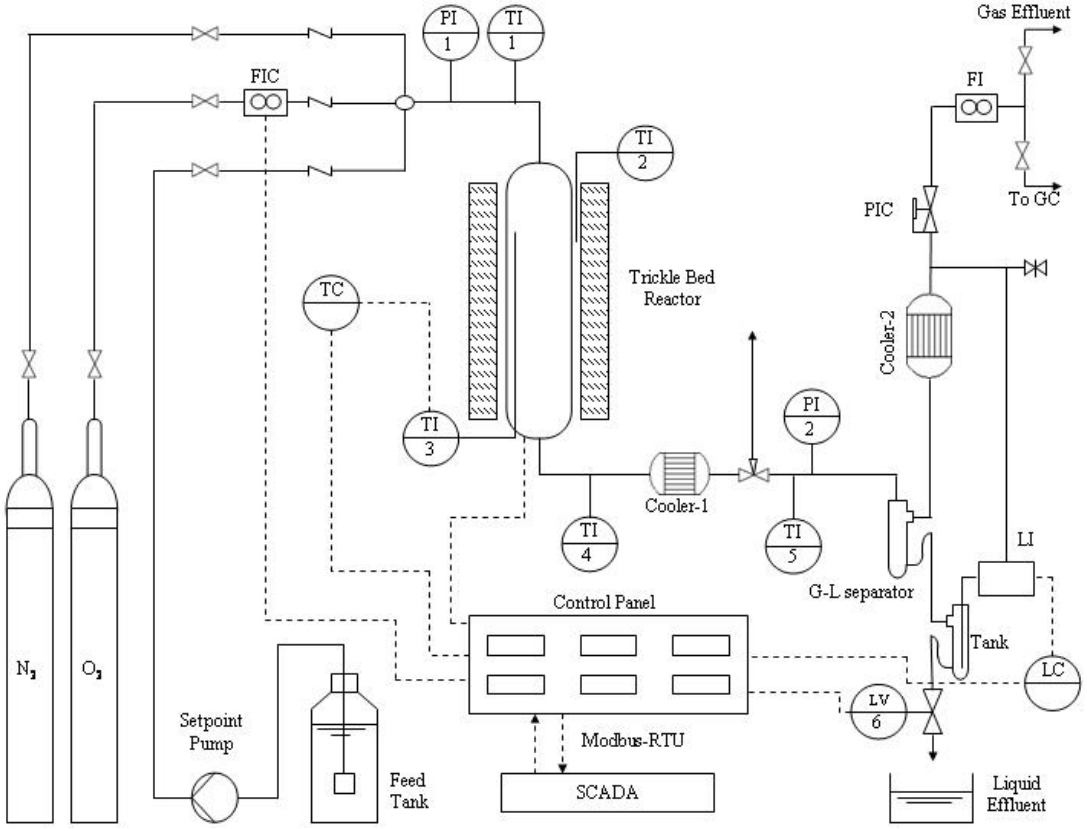


Figure 1: Wet Air Oxidation process diagram. Abbreviations: FI: Flow indicator; FIC: Flow indicator and controller; GC: Gas Chromatography; LC: Level controller; LI: Level indicator; LV: Regulation valve; PI: Pressure indicator; PIC: Backup pressure regulator; SCADA: System control and data acquisition; TI: Temperature indicator.

$$u_n = K_p\left(e_n + \frac{1}{T_i}\sum_{j=1}^{n} e_j T_s + T_d\,\frac{e_n - e_{n-1}}{T_s}\right) \qquad (2)$$

Index n refers to time instant. By tuning we shall mean the activity of adjusting the parameters K_p, T_i and T_d. Several tuning aspects may be illustrated by static considerations. For purely proportional control ($T_d = 0$ and $1/T_i = 0$), the control law (2) reduces to the following equation:

$$u_n = K_p e_n \qquad (3)$$

Considering the feedback loop in Figure 2, where the controller has the proportional gain K_p and the process has the gain K in steady state, the output x can be related to the reference (*ref.*), the load l, and the measurement noise n by the following equation:

$$x = \frac{K_p K}{1 + K_p K}(ref - n) + \frac{K}{1 + K_p K} l \qquad (4)$$

If n and l are zero, then K_p should be high in order to insure that the process output x is close to the *ref.* Furthermore, if l is nonzero, a high value will make the system less sensitive to changes in the load l. But if n is nonzero, K_p should be moderate otherwise the system will be too sensitive to noise. Obviously, the setting of K_p is a balance between: stability, noise sensitivity, and load regulation.

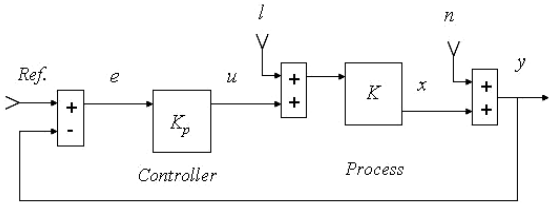


Figure 2: Closed loop system identification.

A PID controller may be tuned using the *Ziegler-Nichols frequency response method*, according to the following procedure:

(a) Increase the proportional gain until the system oscillates (Figure 3); that gain is the ultimate gain K_u.

(b) Read the time between peaks T_u at this setting.

(c) Approximate values for the controller parameters are given in a table.

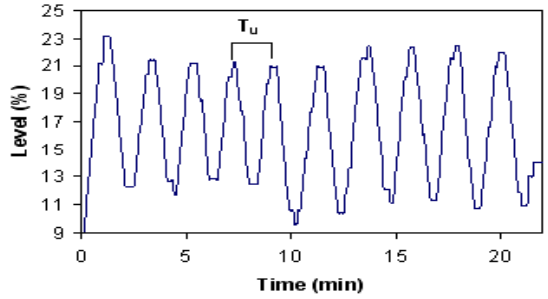


Figure 3: Ziegler-Nichols frequency response method.

The sample period may be related to the derivative gain T_d. In connection with the *Ziegler-Nichols* rules, this implies that T_s should approximately be equal to 1 – 5 percent of the ultimate period T_u. Taking full advantage of this method; we were able to adequately control the reactor temperature and the liquid evacuation unit. When the system reaches the steady state, the controller allows us to maintain a constant liquid level in the tank. Consequently, liquid flow can be continuously evacuated at the system's outlet.

4 HUMAN MACHINE INTERFACE (HMI)

An operator's graphical interface was developed using the FIX MMI Intellution SCADA software which combines high performance monitoring and control with wide range of data acquisition on the Windows NT/2000/9x operating systems. The FIX application contains three sets of multithreaded processes: the user process (HMI), the FIX engine and the industrial automation device servers. These processes interact through a client-server relationship. The user process displays the user interface and executes blocks of code that are defined for control algorithms, supervisory control, analysis and visual presentation. The event-driven engine maintains a real time database, communicates with device servers and performs a multitude of tasks including engineering unit scaling, alarm processing and historical data collection and trending. Device servers are the applications that communicate with Input/Output devices. The FIX application establishes a communication with the ISM112 intelligent sensor modules through the deployment of a Modbus RTU server fully compliant with the latest Modbus RTU protocol definitions. Therefore, the ISM112 data registers can be accessed and modified to the desired values allowing thus the operator to have full control of the process variables.

5 CONCLUSION

The aim of this work is to setup an oxidation process that meets the conditions needed for the aqueous destruction by oxygen or air of organic pollutants. The aforementioned techniques and equipments which in priority are based on regulation and automation procedures, allowed us to design an automated fixed bed reactor that fulfills the required temperatures and pressures conditions (up to 300°C and 25 bar) usually used for CWAO processes. The developed monitoring interface allows the operator to easily manage and control the process parameters. Chemical runs allowing us to validate the system efficacy during the oxidation of various types of aqueous wastes, are in process of completion.

REFERENCES

Debellefontaine, H., Foussard, J.N.,Wet air oxidation for the treatment of industrial wastes, *Waste Manage.*, 20 (2000) 15.

Imamura, S., Fukuda, I., Ishida, S., Wet oxidation catalyzed by ruthenium supported on cerium(IV) oxide, *Ind. Eng. Chem. Res.*, 27 (1988) 718.

Masende, Z.P.G., Kuster, B.F.M., Ptasinski, K.J., Janssen, F.J.J.G., Katima, J.H.Y., and Scouten, J.C., Platinium catalysed wet oxidation of phenol in a stirred slurry reactor: A practical operation window, *Appl. Catal. B*, 41 (2003) 247.

Mishra, V.S., Mahajani,V.V., Joshi, B., Wet Air Oxidation, *Ind. Eng. Chem. Res.*, 34 (1995) 2.

Oliviero, L., Barbier Jr, J., Duprez, D., Guerrero-Ruiz, A., Bachiller-Baeza, B., Rodriguez-Ramoz, I., Catalytic wet air oxidation of phenol and acrylic acid over Ru/C and Ru–CeO_2/C catalysts, *Appl. Catal. B*, 25 (2000) 267.

Pintar, A., Levec, J., Catalytic oxidation of organics in aqueous solutions : I. Kinetics of phenol oxidation, *J. Catal.*, 135 (1992) 345.

Tan, W., Liu, J., Chen, T., and Marquez, H.J., Comparaison of some well-known PID formulas, *Comput. Chem. Eng.*, 30 (2006) 1416.

DIRECTIONAL CHANGE AND WINDUP PHENOMENON

Dariusz Horla
Poznan University of Technology
Institute of Control and Information Engineering
Division of Control and Robotics
ul. Piotrowo 3a, 60-965 Poland
Dariusz.Horla@put.poznan.pl

Keywords: Windup phenomenon, Directional change, Control limits, Multivariable systems.

Abstract: The paper addresses two inherently connected problems, namely: windup phenomenon and directional change in controls problem for multivariable systems. By comparing two ways of performing anti-windup compensation and two different saturation modes a new definition of windup phenomenon for multivariable systems has been obtained, changing definitions present in the literature. It has been shown that avoiding directional change does not have necessarily to mean that windup phenomenon has been avoided too.

1 INTRODUCTION

Consideration of control limits is crucial for achieving high control performance (Peng et al., 1998). There are two ways in which one can consider possible constraints during synthesis of controllers, e.g. imposing constraints during the design procedure, what leads to difficulties with obtaining explicit forms of control laws. The other way is to assume the system is linear and, subsequently, having designed the controller for unconstrained system – impose constraints, what requires then additional changes in control system due to presence of constraints.

A situation when because of, e.g., constraints (or, in general, nonlinearities) internal controller states do not correspond to the actual signals present in the control systems is referred as windup phenomenon (Walgama and Sternby, 1993; Horla, 2004). It is obvious that due to control signal constraints not taken into account during a controller design stage, one can expect inferior performance because of infeasibility of computed control signals.

There are many methods of compensating the windup phenomenon (Peng et al., 1998; Walgama and Sternby, 1993), but a few work well enough in the case of multivariable systems. In such a case, apart from the windup phenomenon itself, one can also observe directional change in the control vector due to, say, different implementation of constraints, what could affect direction of the original, i.e. computed, control vector.

The paper aims to compare two strands in controller design subject to constraints, as mentioned before, and two ways of anti-windup compensation with respect to directional change in controls.

As a result, a new definition of windup phenomenon will be obtained with respect to directional change in controls, which in the case of multivariable systems cannot be omitted.

2 ANTI-WINDUP COMPENSATION

There are two general schemes in anti-windup compensation (AWC) connected with controller design. If the controller has been designed for the case of a linear plant, i.e. with no constraints, introducing them would require certain (most often) heuristic modifications in the control law that usually feed back the difference in between computed $\underline{v}_t$ and constrained control vector $\underline{u}_t$. This is referred in the literature as a posteriori AWC (Horla, 2006a; Horla, 2006b).

The second AWC is incorporated implicitly into the controller, i.e. when controller generates feasible control vector only (belonging to the domain $\mathscr{D}$ of all control vectors for which a certain control perfor-

mance index J_t is of finite value), what is addressed as a priori AWC.

3 A POSTERIORI AWC

One of the most popular AWCs (Peng et al., 1998) are those based on the RST equation, which in the case of multivariable systems is of the form (Horla, 2004)

$$R(q^{-1})\underline{v}_t = -S(q^{-1})\underline{y}_t + T(q^{-1})\underline{r}_t\,, \qquad (1)$$

where

$$\begin{aligned} R(q^{-1}) &= I + R_1 q^{-1} + \cdots + R_{nR} q^{-nR}\,, \\ S(q^{-1}) &= S_0 + S_1 q^{-1} + \cdots + S_{nS} q^{-nS}\,, \\ T(q^{-1}) &= T_0 + T_1 q^{-1} + \cdots + T_{nT} q^{-nT} \end{aligned}$$

are controller polynomial matrices of appropriate sizes, designed for the unconstrained case, $\underline{y}_t \in R^p$ is the output vector, $\underline{v}_t \in R^m$ is the control vector, $d > 0$ is a dead-time.

When the nonlinearities, such as control limits, are taken into consideration the computed vector $\underline{v}_t$ is different from the constrained, i.e. applied, control vector $\underline{u}_t$. In such a case one can modify the control law according to AWC schemes given below (Horla, 2004; Peng et al., 1998).

- Deadbeat AWC (DB)
  $$\underline{v}_t = (I - R(q^{-1}))\underline{u}_t - S(q^{-1})\underline{y}_t + T(q^{-1})\underline{r}_t\,. \quad (2)$$
  The controller is fed back with the constrained control vector, thus no lack of consistency occurs.
- Generalised AWC (G)
  A matrix $A_o(q^{-1})$ of observer polynomials with $nA_o \leq nR$ is added
  $$\begin{aligned} A_o(q^{-1})\underline{v}_t &= (A_o(q^{-1}) - R(q^{-1}))\underline{u}_t - \\ &\quad - S(q^{-1})\underline{y}_t + T(q^{-1})\underline{r}_t\,. \quad (3) \end{aligned}$$
- Conditioning technique AWC (CT)
  The control vector and reference signal are computed as
  $$\begin{aligned} \underline{v}_t &= (I - R(q^{-1}))\underline{u}_t - S(q^{-1})\underline{y}_t + \\ &\quad + (T(q^{-1}) - T_0)\underline{r}^r_t + T_0\underline{r}_t\,, \quad (4) \\ \underline{r}^r_t &= \underline{r}_t + T_0^{-1}(\underline{u}_t - \underline{v}_t)\,. \quad (5) \end{aligned}$$
  A special case of CT is Modified conditioning technique AWC (MCT) where instead of T_0^{-1} there is an inversion of matrix $(T_0 + \Upsilon)^{-1}$ which is responsible for the rate of modification of the reference vector subject to constraints.
  In the CT case, often outputs that were intended to me unmodified, are modified due to conditioning technique. The latter is a result of directional change, that is given rise by anti-windup compensation. The two issues are therefore connected.
- Generalised conditioning technique AWC (GCT)
  The restoration of the consistency is performed by modifying the filtered reference vector, i.e. computing the so-called feasible filtered reference vector,
  $$\begin{aligned} \underline{v}_t &= (I - Q(q^{-1})R(q^{-1}))\underline{u}_t + \\ &\quad + T_{2,0}\underline{r}_{f,t} + (T_2(q^{-1})L(q^{-1}) - T_{2,0})\underline{r}^r_{f,t} + \\ &\quad - Q(q^{-1})S(q^{-1})\underline{y}_t\,. \quad (6) \\ \underline{r}_{f,t} &= Q(q^{-1})L(q^{-1})^{-1}T_1(q^{-1})\underline{r}_t\,, \quad (7) \\ \underline{r}^r_{f,t} &= \underline{r}_{f,t} + T_{2,0}^{-1}(\underline{u}_t - \underline{v}_t)\,, \quad (8) \end{aligned}$$
  Where $T = T_2 T_1$ with monic T_1, and nonsingular $T_{2,0}$.

4 DEFINITION OF WINDUP PHENOMENON IN MULTIVARIABLE SYSTEMS

Currently, one can meet the following definition of windup phenomenon in multivariable systems with its connections to directional change (Walgama and Sternby, 1993):

Solving the windup phenomenon problem does not mean that constrained control vector is of the same direction as computed control vector.

On the other hand, avoiding directional change in control enables one to avoid windup phenomenon.

In further parts of this paper, it has been shown where the latter definition holds, and in what cases it is invalid.

5 DIRECTIONAL CHANGE PHENOMENON, AN EXAMPLE

Let us suppose that two-input two-output system is not coupled and both loops are driven by separate controllers (with no cross-coupling). The system output $\underline{y}_t$ is to track reference vector comprising two sinusoid waves. It corresponds in the (y_1, y_2) plane to drawing a circular shape.

As it can be seen in the Fig. 1a, the unconstrained system performs best, whereas in the case of cut-off saturation of both elements of control vector (Fig. 1b) the tracking performance is poor. In the application for, e.g., shape-cutting performance of the system from Fig. 1c is superior. Nevertheless, it is to be borne in mind that the system is always perfectly decoupled.

6 PLANT MODEL, CONTROL PROBLEM

The following multivariable CARMA plant model will be of interest

$$A(q^{-1})\underline{y}_t = B(q^{-1})\underline{u}_{t-d}\,, \tag{9}$$

with left co-prime polynomial matrices

$$A(q^{-1}){=}I{+}\begin{bmatrix} -1.4 & -0.1 \\ 0.1 & -1.0 \end{bmatrix}q^{-1}{+}\begin{bmatrix} 0.49 & 0 \\ 0 & 0.25 \end{bmatrix}q^{-2},$$

$$B(q^{-1}) = I + \mathrm{diag}\{0.5, 0.5\}\,q^{-1}$$

and $d = 1$.

The plant is cross-coupled and comprises fourth-order matrices in the transfer matrix representation (being stable and minimumphase).

6.1 RST Controller (a Posteriori AWC)

It is assumed that the plant is controlled by a multivariable pole-placement controller with characteristic polynomial matrix

$$A_M(q^{-1}) \;\; = \;\; I + \mathrm{diag}\{-0.5, -0.5\}\,q^{-1}\,.$$

The controller is given in RST structure with polynomial matrices $R(q^{-1})$ and $S(q^{-1})$ resulting from Diophantine equation

$$A(q^{-1})R(q^{-1}) + q^{-d}B(q^{-1})S(q^{-1}) = A_M(q^{-1})A_o(q^{-1})\,, \tag{10}$$

with $A_o(q^{-1}) = I - 0.2Iq^{-1}$, $T(q^{-1}) = KA_o(q^{-1})$, $A_M(1) = B(1)K$.

Having imposed control limits upon the RST controller requires implementing a posteriori AWC techniques in order to restore good performance quality.

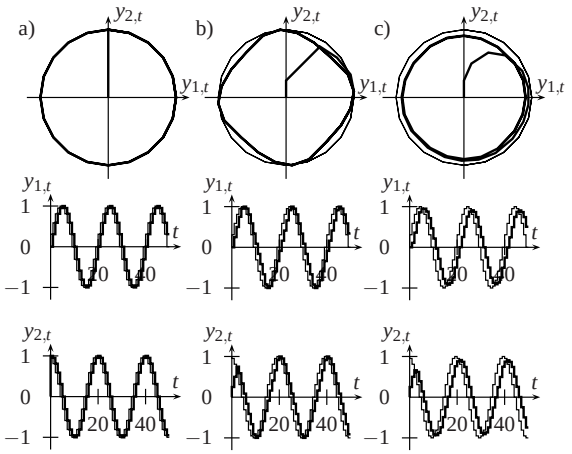


Figure 1: a) unconstrained system, b) cut-off saturation, c) direction-preserving saturation.

6.2 Optimised Controller (a Priori AWC)

In comparison, the RST pole-placement controller performance will be compared with a priori AWC controller, namely multivariable pole-placement controller utilising the theory of predictive control and convex optimisation techniques.

In order to enable such a comparison, the predictive controller has been deprived of all its advantages – the prediction horizon has been chosen as one step, thus the optimal constrained control vector is searched (Horla, 2006a; Horla, 2006b)

$$\underline{u}_t^{\star} : J_t(\underline{u}_t^{\star}) = \inf_{\underline{u}_t \in \mathscr{D}(J_t)} \{J_t(\underline{u}_t)\}\,,$$

where its jth component has been symmetrically constrained

$$|u_{j,t}| \;\; \leq \;\; \alpha_j\,,$$

where $\underline{u}_t = \begin{bmatrix} u_{1,t} & u_{2,t} & \cdots & u_{m,t} \end{bmatrix}^T$.

The performance index has been chosen as a sum of squared tracking errors resulting from a reference model output

$$J_t \;\; = \;\; \left\| \underline{r}_{M,t+d} - \underline{\hat{y}}_{t+d} - \underline{\hat{\hat{y}}}_{t+d} \right\|_2^2\,, \tag{11}$$

where one-step ($d = 1$) prediction of system output comprises as in (11) forced and free-response output vectors.

The performance index can be rewritten into quadratic form

$$J_t = \left(G\underline{u}_t + \underline{\hat{\hat{y}}}_{t+d} - \underline{r}_{M,t+d} \right)^T \left(G\underline{u}_t + \underline{\hat{\hat{y}}}_{t+d} - \underline{r}_{M,t+d} \right), \tag{12}$$

and the optimisation can be performed with the use of its linear matrix inequality (LMI) form with the last two LMIs responsible for control constraints, as below

$$\begin{aligned} \min \quad & \gamma \\ \text{s.t.} \quad & \begin{bmatrix} I & \star \\ \underline{u}_t^T (G^T G)^{1/2} & \begin{pmatrix} \gamma - (\underline{r}_{M,t+d} - \underline{\hat{\hat{y}}}_{t+d})^T \times \\ \times (\underline{r}_{M,t+d} - \underline{\hat{\hat{y}}}_{t+d}) + \\ +2(\underline{r}_{M,t+d} - \underline{\hat{\hat{y}}}_{t+d})^T G\underline{u}_t \end{pmatrix} \end{bmatrix} \geq 0\,, \\ & \mathrm{diag}\{\alpha_1 - u_{1,t}, \ldots, \alpha_m - u_{m,t}\} \geq 0\,, \\ & \mathrm{diag}\{\alpha_1 + u_{1,t}, \ldots, \alpha_m + u_{m,t}\} \geq 0\,, \end{aligned} \tag{13}$$

where $\star$ detones a symmetrical entry, and G is an impulse-response matrix.

7 SIMULATION STUDIES

The simulations have been performed for two controllers: for a pole-placement controller with a group of a posteriori AWCs and LMI-based predictive pole-placement controller.

In order to evaluate control performance connected with anti-windup compensation performance the following performance indices have been introduced

$$J = \frac{1}{N}\sum_{i=1}^{2}\sum_{t=1}^{N}|r_{i,t}-y_{i,t}|, \quad (14)$$

$$\overline{\varphi} = \frac{1}{N}\sum_{t=1}^{N}|\varphi(\underline{v}_t)-\varphi(\underline{u}_t)| \ \ [^\circ], \quad (15)$$

where (14) corresponds to mean absolute tracking error on both outputs and (15) is a mean absolute direction change in between computed and constrained control vector.

The control vector has been constrained in all cases to $\alpha_1 = \pm 0.2$ on the first input and $\alpha_2 = \pm 0.3$ for the second output. The reference vector is a square-wave signal of amplitude ± 1 and simulation horizon $N = 150$.

Performance indices have been given in the Tab. 1.

Table 1: Performance indices for a) cut-off saturation, b) direction-preserving saturation.

		–	*DB*	*G*	*CT*	*MCT*	*GCT*
a)	J	1.1539	1.1539	1.1539	1.1539	1.1536	1.1516
	$\overline{\varphi}$	0.9003	0.9004	1.1861	1.1862	1.1093	1.5495
		–	*DB*	*G*	*CT*	*MCT*	*GCT*
b)	J	1.1772	1.1672	1.1677	1.1677	1.1670	1.1655
	$\overline{\varphi}$	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

As it can be seen from Tab. 1a, and Figs. 4, 5, cut-off saturation causes directional change, which is visible during reference vector changes. It is necessary to alter the decoupling of the plant, in order to restore high control performance. As it can be seen, the greater the mean absolute direction change, the lesser the performance index is. Thus, it can be said, that directional change supports anti-windup compensation.

On the other hand, as in Tab. 1b, and Figs. 6, 7, direction-preserving saturation does not cause directional change. Preserving a constant direction causes performance indices to increase, dependless of the method of anti-windup compensation. The coupling is clearly visible during tracking when reference vector changes. Constant direction prevents the controller from decoupling the plant – performance is inferior.

In order to stipulate the differences in between saturation methods, two GCT-AWC cases have been chosen and compared with LMI-based approach.

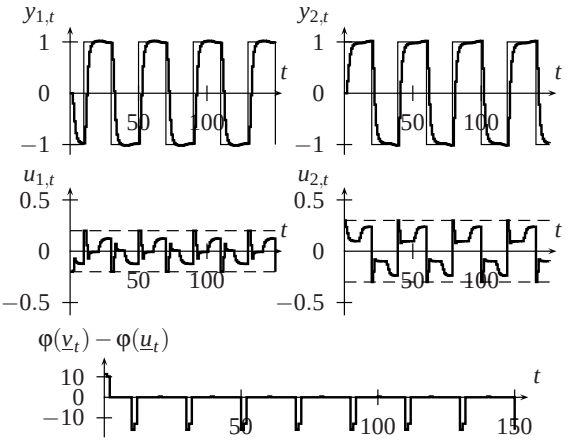


Figure 2: Overall performance for cut-off saturation with GCT-AWC.

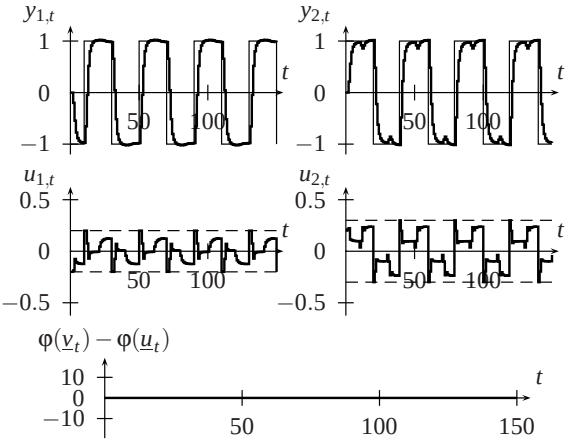


Figure 3: Overall performance for direction-preserving saturation with GCT-AWC.

In the Fig. 8 one can see a priori anti-windup compensator performance, where only feasible control actions are generated. The performance indices in such a case are of the best values, i.e. $J = 1.0450$, $\overline{\varphi} = 64.7063^\circ$.

Having compared Figs. 2–8 it can be said, that in order to achieve the best performance one has to alter the direction of a computed control vector. The greater the directional change is, the better the control performance.

For a priori AWC, visible changes in control direction result from decoupling phase, i.e. whenever control vector encounters constraints it has to be constrained in such a way as to achieve the high control performance (the last plot corresponds to the angle

difference in between control vector computed in the unconstrained case using optimisation algorithm, and constrained a priori AWC control vector).

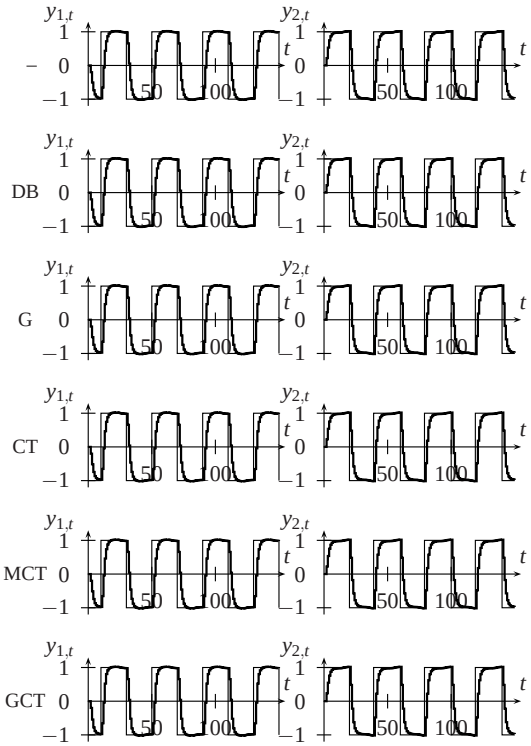


Figure 4: Tracking performance for cut-off saturation.

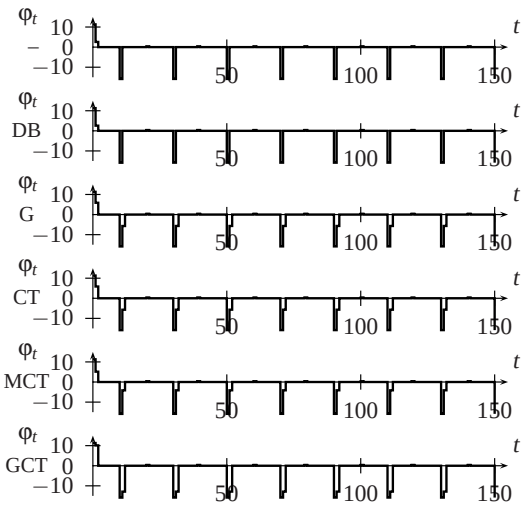


Figure 5: Directional change for cut-off saturation.

On the other hand, having constrained the control vector alters decoupling, thus its direction has to be additionally altered, what is mostly visible in Fig. 8.

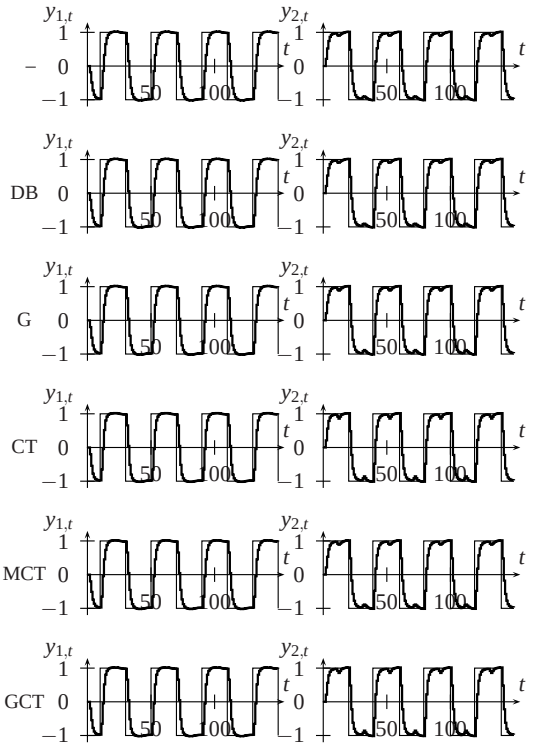


Figure 6: Tracking performance for direction-preserving saturation.

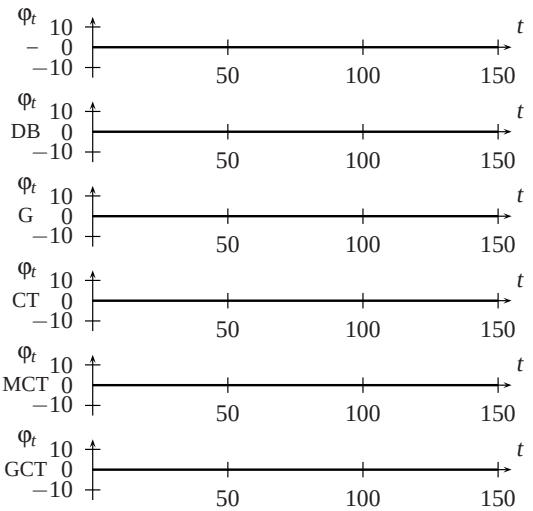


Figure 7: Directional change for direction-preserving saturation.

Finally, in the Fig. 9 it has been shown that both a priori and a posteriori AWCs need to alter direction of controls in order to restore high quality of tracking performance. In addition, a priori AWC has no fixed structure, nor decoupling compensator, thus changes

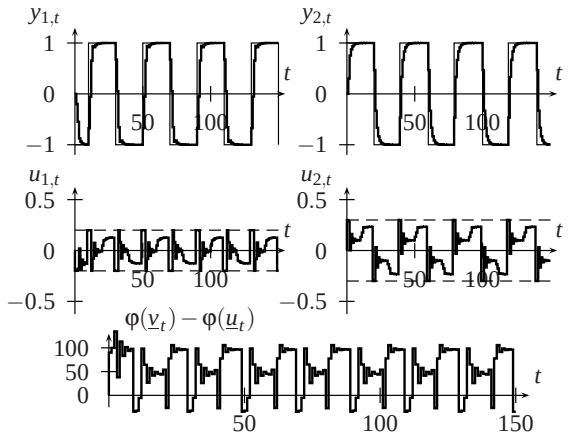


Figure 8: Overall performance for LMI-based control with a priori AWC, $J = 1.0450$, $\overline{\varphi} = 64.7063^\circ$.

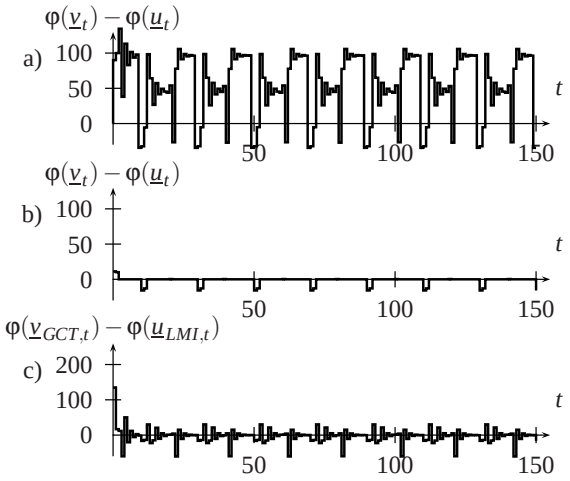


Figure 9: A comparison of direction changes a) a priori AWC, b) a posteriori GCT-AWC with cut-off saturation, c) a priori AWC vs. GCT-AWC with cut-off saturation.

in control direction must be greater than in the case of GCT-AWC with cut-off saturation.

The comparison of angle difference in between unconstrained control vector computed for GCT case and constrained control vector computed by a priori AWC, shows that approximately a priori AWC acts in the direction of unconstrained controller with GCT-AWC, i.e. both of them try to get the system's performance as close as possible to the performance of ideal pole-placement subject to no constraints.

Nevertheless, the simulations have shown that in order to obtain a good performance one has to change direction of control vector. Without the latter one will observe coupling.

8 SUMMARY – NEW DEFINITION OF WINDUP PHENOMENON IN MULTIVARIABLE SYSTEMS

One can formulate a new definition:

*Solving the windup phenomenon problem does not **have to** mean that constrained control vector is of the same direction as computed control vector **if cross-coupling is present in the control system.***

*On the other hand, avoiding directional change in control enables one to avoid windup phenomenon **if and only in the plant is perfectly decoupled or is not coupled at all.*** (what due to the constraints is hardly ever met)

Such a definition definitely changes the way one should look at windup phenomenon and its connection with directional change problem.

REFERENCES

Horla, D. (2004). Direction alteration of control vector and anti-windup compensation for multivariable systems (in Polish). *Studies in Automation and Information Technology*, 28/29:53–68.

Horla, D. (2006a). LMI-based multivariable adaptive predictive controller with anti-windup compensator. In *Proceedings of the 12th IEEE International Conference MMAR*, pages 459–462, Miedzyzdroje.

Horla, D. (2006b). Standard vs. LMI approach to a convex optimisation problem in multivariable predictive control task with a priori anti-windup compensator. In *Proceedings of the 18th ICSS*, pages 147–152, Coventry.

Peng, Y., Vrančić, D., Hanus, R., and Weller, S. (1998). Anti-windup designs for multivariable controllers. *Automatica*, 34(12):1559–1565.

Walgama, K. and Sternby, J. (1993). Contidioning technique for multiinput multioutput processes with input saturation. *IEE Proceedings-D*, 140(4):231–241.

IMAGE PREPROCESSING FOR CBIR SYSTEM

Tatiana Jaworska
Systems Research Institute Polish Academy of Sciences, Newelska 6 St, 01-447 Warsaw, Poland
Tatiana.Jaworska@ibspan.waw.pl

Keywords: Content-based image retrieval (CBIR), image preprocessing, image segmentation, clustering, object extraction, texture extraction, discrete wavelet transformation.

Abstract: This article describes the way in which image is prepared for content-based image retrieval system. Our CBIR system is dedicated to support estate agents. In our database there are images of houses and bungalows. All efforts have been put into extracting elements from an image and finding their characteristic features in the unsupervised way. Hence, the paper presents segmentation algorithm based on a pixel colour in RGB colour space. Next, it presents the method of object extraction in order to obtain separate objects prepared for the process of introducing them into database and further recognition. Moreover, a novel method of texture identification which is based on wavelet transformation, is applied.

1 INTRODUCTION

Image processing for purposes of content-based image retrieval (CBIR) systems seems to be a very challenging task for the computer. Determining how to store images in big databases, and later, how to retrieve information from them, is an active area of research for many computer science fields, including graphics, image processing, information retrieval and databases.

Although attempts have been made to perform CBIR in an efficient way based on shape, colour, texture and spatial relations, it has yet to attain maturity. A major problem in this area is computer perception. There remains a big gap between low-level features like shape, colour, texture and spatial relations, and high-level features like windows, roofs, flowers, etc.

The purpose of this paper is to investigate image processing with special attention given to segmentation and selection of separate objects from the whole image. In order to achieve this aim we present two new methods: one is a very fast algorithm for colour image segmentation, and the second is a new approach to description of textured objects, using discrete wavelet transformation.

2 CBIR CONCEPTION OVERVIEW

In the last 15 years, CBIR techniques have drawn much interest, and image retrieval techniques have been proposed in context of searching information from image databases. In the 90's the Chabot project at UC Berkeley (Ogle, 1995) was initialized to study storage and retrieval of a vast collection of digitized images. Also, at IBM Almaden Research Centre CBIR was prepared by Flickner (Flickner, 1995), Niblack (Niblack, 1993). This approach was improved by Tan (Tan, 2001), Hsu (Hsu, 2000) and by Mokhtarian, F. S. Abbasi and J. Kittler (Mokhtarian, 1996) at Department of Electronics and Electrical Engineering UK.

Our CBIR system is dedicated to support estate agents. In the estate database there are images of houses, bungalows, and other buildings. To be effective in terms of presentation and choice of houses, the system has to be able to find the image of a house with defined architectural elements, for example: windows, roofs, doors, etc. (Jaworska, 2005).

The first stage of our analysis is to split the original image into several meaningful clusters; each of them provides certain semantics in terms of human understanding of image content. Then, proper features are extracted from these clusters to represent the image content on the visual perception level. In the interest of the following processes, such

as object recognition, the image features should be selected carefully. Nevertheless, our efforts have been put into extracting elements from an image in the unsupervised way.

Figure 1: Example of an original image.

3 A NEW FAST ALGORITHM FOR OBJECT EXTRACTION FROM COLOUR IMAGES

We definitely prefer unsupervised techniques of image processing. Although there are many different methods of image segmentation, we began with two well known clustering algorithms: the C-means clustering (Seber,1984), (Spath, 1985), and later developed, the fuzzy C-means clustering algorithm (FCM) (Bezdek, 1981). In our case we found clusters in the 3D colour space RGB and HSV.

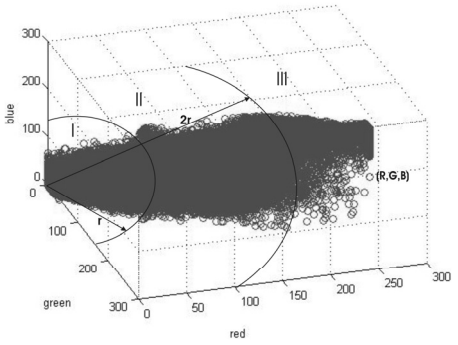


Figure 2: The way of labelling the set of pixels. Regions I, II, III show pixel brightness and the biggest value of triple (R,G,B) determines its colour.

Unfortunately, results were unsatisfying. After examining the point distribution in these both spaces (for all images) it turned out that points created one tight set. In figure 2 such a set is exemplified in RGB space but points distribution in HSV space is similar.

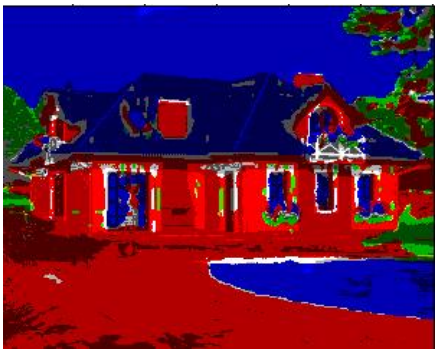


Figure 3: 12 cluster segmentation of fig. 1 obtained by using the 'colour' algorithm.

These results forced us to work out a new algorithm which uses colour information about a single point to greater extent than the C-means algorithm does. With the aim of labelling a pixel we chose the biggest value from the triple (R,G,B) and we defined it as a cluster colour. In this way we obtained three segments – red, green and blue and for better result we divided each colour into three shades, according to the darkness of colour shown as three regions (I, II, III) which determine point brightness. The idea of the segmentation is illustrated in figure 2. The radius $r = \sqrt{R_{\max}^2 + G_{\max}^2 + B_{\max}^2}/3$ of the dividing sphere was counted in Euclidean measure, where $R_{\max} = G_{\max} = B_{\max} \# 255$. Moreover, we added three segments: black, grey and white for pixels for whom R=G=B according to their region (I, II, III). We called this algorithm 'colour one'.

Figure 3 presents the image shown in fig. 1 divided into 12 clusters using the above-described algorithm.

4 OBJECT EXTRACTION ON THE BASE OF THE NEW ALGORITHM

Based on this segmentation separate objects are obtained. As an object we understand an image of architectural element such as roof, chimney, door, window, etc.

After performing the extraction of objects, the following features for these objects were counted: average colour (shown in fig. 4), texture parameters, region-based shape descriptors, contour based shape descriptors and location in the image as a region-based representation.

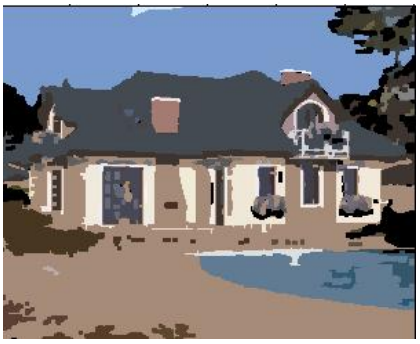


Figure 4: Objects from fig. 2 presented in their average colours.

5 THE DETERMINATION OF TEXTURE PARAMETERS

Figure 5: Example of an original image where the roof is a textured surface.

The texture information presented in images is one of the most powerful additional tools available. There are many methods which can be used for texture characterization. Unfortunately, they are mostly useless for our purpose.

One of them is the two-dimensional frequency transformation. For our aims we could apply as well the classical Fourier transformation as several spatial-domain texture-sensitive operators, for instance, the Laplacian 3x3 or 5x5, the Gaussian 5x5, Hurst, Haralick, or Frei and Chen (Russ, 1995). Regrettably, all of them are useful for relatively small neighbourhoods.

The other method of texture recognition for monochromatic image is the histogram thresholding. Unfortunately, it can be used mainly for distinguishing 2-3 textured regions. There also exists the two-dimensional histogram of pixel pairs proposed by Haralick in 1973 (Haralick, 1973).

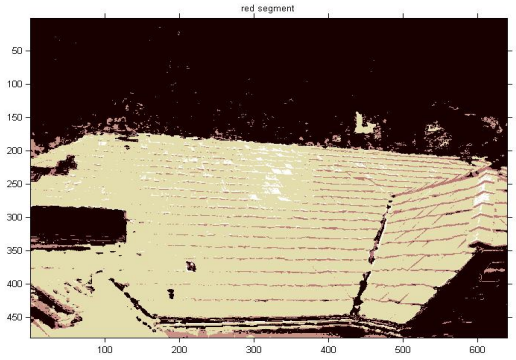


Figure 6: The red segment (in three levels of brightness) extracted from the whole segmentation from fig. 8.

The next methods are the transformation domain approaches. In 2001 Balmelli and Mojsilović (Balmelli, 2001) proposed the wavelet domain for texture and pattern using statistical features only for regular textures and geometrical patterns. So far only Lewis and Fauzi manage to perform an automatic texture segmentation for CBIR based on discrete WT (DWT) (Fauzi, 2006).

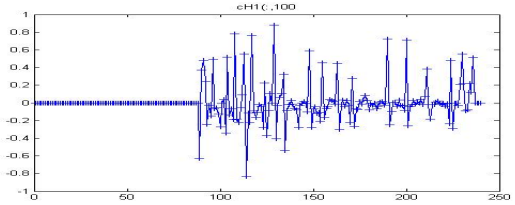


Figure 7: Horizontal wavelet coefficients presented along the 100th column of the image transform (for the Haar wavelet, where j=1). Numbers of the Haar wavelets for the first level of multiresolution analysis are on the horizontal axis and values of coefficients cH1 are on vertical axis.

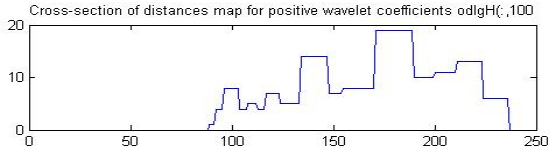


Figure 8: Cross-section through the 100th column of the distances map for positive horizontal wavelet coefficients. Numbers of the Haar wavelets for the first level of multi-resolution analysis are on the horizontal axis and distances between the maximal wavelet coefficients are on vertical axis.

In our work we decided to use the Fast Wavelet Transform (FWT). It is efficient and productive enough for frequent use for our purpose.

One of the most important features of details is their directionality. If we use this feature and compute the convolution of an image consisting of

regular tiles or bricks and relevant wavelet, we obtain a 2D transform whose maximum values are placed in the connection spots among these tiles or bricks.

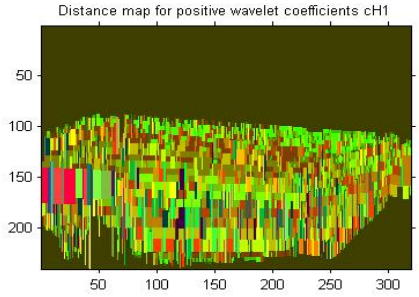


Figure 9: Distance map for positive horizontal wavelet coefficients cH1. There are wavelet numbers on both axes.

Therefore, we have applied the Haar wavelet to the roof region shown in fig. 7. Then, we obtained three matrices of details d_1^1, d_1^2 and d_1^3. The cross-section through the 100th column of the horizontal details matrix d_1^1 (cH1) is presented in figure 8. Maxima and minima in this figure are equivalent to connections between tiles in fig. 7. Having computed horizontal details, we have measured distances between maxima for each column of this matrix (shown in fig. 8) and we have measured distances between minima for each column of this matrix. We have located one threshold on the level of 1% of the maximum value of the whole matrix and we have measured distances between positive coefficients on that level and we have done analogically for negative coefficients. It has turned out that these distances which are equivalent to the size of tiles are good distinctive parameters for textured region.

After counting the distances we have created two distance maps for all positive and negative horizontal coefficients. Figure 9 presents one of these distance maps. Analogical procedure has been carried out for vertical wavelet coefficients cV1. Basing on the above distance maps we can estimate that the size of tiles.

6 CONCLUSIONS

To sum up, this paper shows how to extract elements from images in the unsupervised way and analyze objects parameters. We have focused on the description of texture parameters because it was the most difficult task. The achieved results indicate that it is possible to separate objects in the image with acceptable accuracy for further interpretation in the unsupervised way. In computer terms, objects are recognized by finding the above-mentioned features of each object and a new object is classified to one of the previous created classes. So far, we have no interpretation which of these objects are doors, windows, etc. At present, the database structure is being prepared. This structure will cover all elements necessary for image content analysis; namely basic object features as well as logical and spatial relations.

REFERENCES

Bezdek, J. C., 1981. Pattern Recognition with Fuzzy Objective Function Algorithms, *Plenum Press*, New York.

Fauzi, M., Lewis, P., 2006. Automatic texture segmentation for content-based image retrieval application, *Pattern Analysis and Applications*, Springer-Verlag, London, (in printing).

Flickner, M., Sawhney, H., et al., 1995. Query by Image and Video Content: The QBIC System, *IEEE Computer*, Vol. 28, No. 9, pp. 23-32.

Haralick, R. M., Shanmugan, K., Dinstein, I., 1973. Texture Features for Image Classification, *IEEE Transactions of Systems, Man and Cyberntics*, *SMC-3*, pp. 610-621.

Hsu, W., Chua, T. S., Pung, H. K., 2000. Approximation Content-based Object-Level Image Retrieval, *Multimedia Tools and Applications*, Vol. 12, Springer Netherlands, pp. 59-79.

Jaworska, T., Partyka, A., 2005. Research: Content-based image retrieval system [in Polish], *Report RB/37/2005*, Systems Research Institute, PAS.

Mokhtarian, F., Abbasi, S., Kittler J., 1996. Robust and Efficient Shape Indexing through Curvature Scale Space, *Proc. British Machine Vision Conference*, pp. 53-62.

Niblack, W., Flickner, M., et al., 1993. The QBIC Project: Querying Images by Content Using Colour, Texture and Shape, *SPIE*, Vol. 1908, pp. 173-187.

Ogle, V., Stonebraker, M., 1995. CHABOT: Retrieval from a Relational Database of Images, *IEEE Computer*, Vol. 28, No 9, pp. 40-48.

Russ, J. C., 1995 The image processing. Handbook, *CRC*, London, pp. 361-385.

Seber, G., 1984. Multivariate Observations, *Wiley*.

Spath, H., 1985. Cluster Dissection and Analysis: Theory, FORTRAN Programs, Examples, translated by J. Goldschmidt, *Halsted Press*, pp. 226.

Tan, K-L., Ooi, B. Ch., Yee, Ch. Y., 2001. An Evaluation of Color-Spatial Retrieval Techniques for Large Image Databases, *Multimedia Tools and Applications*, Vol. 14, Springer Netherlands, pp. 55-78.

USE A NEURAL NETWORKS TO ESTIMATE AND TRACK THE PN SEQUENCE IN LOWER SNR DS-SS SIGNALS

Tianqi Zhang[1], Shaosheng Dai[1]
[1]InstituteSchool of Communication and Information Engineering / Institute of Signal Processing and System On Chip (ISPSOC), Chongqing University of Posts and Telecommunications (CQUPT), Chongqing 400065, China
zhangtianqi@tsinghua.org.cn

Zhengzhong Zhou[2], Xiaokang Lin[3]
[2] School of Communication and Information Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu 610054, China
[3] Graduate School at Shenzhen of Tsinghua University, Shenzhen 518055, China

Keywords: Generalized Hebbian algorithm (GHA), neural network (NN), direct sequence spread spectrum (DS-SS) signals, pseudo noise (PN) sequence.

Abstract: This paper proposes a modified Sanger's generalized Hebbian algorithm (GHA) neural network (NN) method to estimate and track the pseudo noise (PN) sequence in lower signal to noise ratios (SNR) direct sequence spread spectrum (DS-SS) signals. The proposed method is based on eigen-analysis of DS-SS signals. The received signal is firstly sampled and divided into non-overlapping signal vectors according to a temporal window, which duration is a periods of PN sequence. Then an autocorrelation matrix is computed and accumulated by these signal vectors one by one. The PN sequence can be estimated and tracked by the principal eigenvector of autocorrelation matrix in the end. But the eigen-analysis method becomes inefficiency when the estimated PN sequence becomes longer or the estimated PN sequence becomes time varying. In order to overcome these shortcomings, we use a modified Sanger's GHA NN to realize the PN sequence estimation and tracking from lower SNR input DS-SS signals adaptively and effectively.

1 INTRODUCTION

Since the direct sequence spread spectrum (DS-SS, DS) signals have the distinguished capability of anti-jamming and lower probability interception, the DS signals have used broadly in communication, radar, telemetry and telecommand etc for a long time. Usually, the spread spectrum receiver has to perform synchronization before it can start the despreading operation. For the case of DS, this entails establishing complete knowledge of the pseudo noise (PN) sequence and the timing. Synchronization is performed in two stages. The first stage of coarse synchronization is known as PN acquisition and the final stage of maintaining the fine synchronization is called PN tracking. While PN tracking forms an important part of DS synchronization, PN acquisition is a more challenging problem.

Conventional acquisition techniques (Simnon et al., 1994) rely on the knowledge of the internal algebraic structure of the PN spreading sequence to establish synchronization. While they demonstrate good acquisition performance in low noise environments, they tend to break down in environments with high levels of noise and interference because of a high false alarm rate. Furthermore, reliable algebraic techniques for synchronization have yet to be developed for nonlinear codes, or codes with unknown code structure, chip constellations, and residual delay. Additionally, the PN sequences of DS signal have the distinguished function of keeping secrecy. If you have no knowledge of the PN sequence, you could not demodulate the transmitted message symbols generally.

A method of autocorrelation and cyclic autocorrelation was proposed to de-spread the DS

signal (French et al., 1986), which can extract a differentially-encoded estimate of the underlying message sequence from a modulation-on-symbol DS signal (where the spreading PN sequence repeats once per message symbol) on the basis of the periodic structure of these signals. This method attempts to overcome some of these disadvantages by making no assumptions about the internal algebraic structure of the PN spreading sequence. They can operate in the presence of arbitrary delay and for arbitrary codes or chip constellations. Because some spectral correlation computations are required, it is difficult to carry out in real-time. Furthermore, it does only de-spread the DS signal without the PN sequence, but it doesn't utilize or analyze any signal structure information. So far, most of DS packet radio and military systems often require frequent, fast and robust synchronization. Blind estimation and tracking of the PN spreading sequence without the a priori knowledge of its structure and timing is useful in achieving these objectives.

The signal subspace analysis and relational techniques, introduced in (Zhang et al., 2005) (Simic et al., 2005) (Zhan et al., 2005), is precisely such a technique. It is based on the signal subspace analysis of DS signal, and estimates the PN spreading sequence blindly by exploiting cyclostationarity property and eigenstructure of the DS signal. The technique provides perfect estimates of the PN spreading sequence under the assumptions of infinite time-averaging in the presence of arbitrary levels of temporally-white background noise. But the methods proposed in (Zhang et al., 2005) (Simic et al., 2005) (Zhan et al., 2005) belong to a batch method, when the number of samples in a period of observation window becomes too large or the estimated PN sequence becomes time-varying, the computation of matrix decomposition may not be feasible in practice.

This paper proposes an unsupervised adaptive approach of Sanger's generalized Hebbian algorithm (GHA) neural networks (NN) to PN sequence blind estimation and adaptive tracking. It needs the first and second principal component vectors associated with the largest and second largest eigenvalue respectively; and it can deal with too long sampling signal vectors and time-varying cases.

2 SIGNAL MODEL

The base band DS signal $x(t)$ corrupted by the white Gaussian noise $n(t)$ with the zero mean and σ_n^2 variance can be expressed as (French et al., 1986) (Zhang et al., 2005) (Simic et al., 2005) (Zhan et al., 2005)

$$x(t) = s(t - T_x) + n(t) \quad (1)$$

Where $s(t) = d(t)p(t)$ is the DS signal, $p(t) = \sum_{j=-\infty}^{\infty} p_j q(t - jT_c)$, $p_j \in \{\pm 1\}$ is the periodic PN sequence, $d(t) = \sum_{k=-\infty}^{\infty} m_k q(t - kT_0)$, $m_k \in \{\pm 1\}$ is the symbol bits, uniformly distributed with $E[m_k m_l] = \delta(k - l)$, $\delta(\cdot)$ is the Dirac function, $q(t)$ denotes a pulse chip. Where $T_0 = NT_c$, N is the length of PN sequence, T_0 is the period of PN sequence, T_c is the chip duration, T_x is the random time delay and uniformly distributed on the $[0, T_0]$. According to the above, the PN sequence and synchronization are required to de-spread the received DS signals. But in some cases, we only have the received DS signals. We must estimate the signal parameters firstly (We assume that T_0 and T_c had known in this paper), and then estimate the PN sequence and synchronization.

3 SUBSPACE ANALYSIS BASED ON K-L TRANSFORMATION

The received DS signal is sampled and divided into non-overlapping temporal windows, the duration of which is T_0. Then one of the received signal vector is

$$\mathbf{X}(k) = \mathbf{s}(k) + \mathbf{n}(k), \quad k = 1,2,3,\cdots \quad (2)$$

Where $\mathbf{s}(k)$ is the k-th vector of useful signal, $\mathbf{n}(k)$ is the additive white Gaussian noise vector. The dimension of vector $\mathbf{X}(k)$ is $N = T_0 / T_c$. If the random time-delay is T_x, $0 \le T_x < T_0$, $\mathbf{s}(k)$ may contain two consecutive symbol bits, each modulated by a period of PN sequence, i.e.

$$\mathbf{s}(k) = m_k \mathbf{p}_1 + m_{k+1} \mathbf{p}_2 \quad (3)$$

Where m_k and m_{k+1} are the two consecutive symbol bits, $\mathbf{p}_1$ ($\mathbf{p}_2$) is the right (left) part of the PN sequence waveform.

According to K-L transformation, we normalize $\mathbf{p}_i$ by $\mathbf{u}_i = \mathbf{p}_i / \|\mathbf{p}_i\|$, $i = 1,2$

$$\mathbf{u}_i^T \mathbf{u}_j = \delta(i-j) \quad , \quad i,j = 1,2 \tag{4}$$

Where $\mathbf{u}_1$ and $\mathbf{u}_2$ are ortho-normal vectors, $\delta(\cdot)$ is a Dirac function. From $\mathbf{u}_1$ and $\mathbf{u}_2$, we have

$$\mathbf{X}(k) = m_k \|\mathbf{p}_1\| \mathbf{u}_1 + m_{k+1} \|\mathbf{p}_2\| \mathbf{u}_2 + \mathbf{n}(k) \tag{5}$$

The autocorrelation matrix of $\mathbf{X}(k)$: $\mathbf{R}_X$ may be estimated as

$$\hat{\mathbf{R}}_X(M) = \frac{1}{M} \sum_{i=1}^{M} \mathbf{X}(i) \mathbf{X}^T(i) \tag{6}$$

Assume $\mathbf{s}(k)$, $\mathbf{n}(k)$ are mutually independent, substitute Eq.(5) into Eq.(6) yields

$$\begin{aligned} \mathbf{R}_X &= \hat{\mathbf{R}}_X(\infty) = \bar{\mathbf{U}}_s \mathbf{\Lambda}_s \bar{\mathbf{U}}_s^T + \bar{\mathbf{U}}_n \mathbf{\Lambda}_n \bar{\mathbf{U}}_n^T \\ &= \sigma_n^2 \left\{ \left(\gamma_{SNR} \cdot \frac{T_0 - T_x}{T_c} \right) \cdot \bar{\mathbf{u}}_1 \bar{\mathbf{u}}_1^T + \left(\gamma_{SNR} \cdot \frac{T_x}{T_c} \right) \cdot \bar{\mathbf{u}}_2 \bar{\mathbf{u}}_2^T + \mathbf{I} \right\} \end{aligned} \tag{7}$$

Where $\mathbf{I}$ is an identity matrix of dimension $N \times N$, the expectation of m_k is zero. The variance of m_k is σ_m^2, the symbol is uncorrelated from each other. The energy of PN sequence is $E_p \approx T_c \|\mathbf{p}\|^2$, the variance of $\mathbf{s}(k)$ is $\sigma_s^2 = \sigma_m^2 E_p / T_0$, $\gamma_{SNR} = \sigma_s^2 / \sigma_n^2$. The row vectors of $\bar{\mathbf{U}}_s$ and $\bar{\mathbf{U}}_n$ are corresponding to the eigenvectors of eigenvalue $\lambda_{R1} = \left[1 + \gamma_{SNR} \cdot (T_0 - T_x) / T_c \right] \sigma_n^2$, $\lambda_{R2} = \left(1 + \gamma_{SNR} \cdot T_x / T_c \right) \sigma_n^2$ and σ_n^2 , and exist $\lambda_{R1} \geq \lambda_{R2} > \sigma_n^2$. It is clear that the eigenvalues of $\mathbf{R}_X$ are dependent on T_x . It is shown in (Anderson, 1963) that the estimated principal eigenvectors have the following behavior:

$$\|\bar{\mathbf{u}}_i - \mathbf{u}_i\| = O\left(\sqrt{\log\log M / M} \right), \ i = 1,2,\cdots,K \ .$$

Therefore, $M \to \infty$, there always exists $\bar{\mathbf{u}}_i = \mathbf{u}_i$, $i = 1,2,\cdots,K$.

When $T_x \neq 0$, the biggest eigenvalue is λ_{R1}, the sign of its corresponding eigenvector $\mathbf{p}_1 = \text{sign}(\bar{\mathbf{u}}_1)$. The second biggest eigenvalue is λ_{R2} and the sign of its corresponding eigenvector $\mathbf{p}_2 = \text{sign}(\bar{\mathbf{u}}_2)$. We can recover a period PN sequence from $\mathbf{p} = \mathbf{p}_2 + \mathbf{p}_1 = \text{sign}(\bar{\mathbf{u}}_2) + \text{sign}(\bar{\mathbf{u}}_1)$. When $T_x = 0$, λ_{R1} and $\mathbf{p}_1 = \text{sign}(\bar{\mathbf{u}}_1)$ which denote a period of PN sequence.

Because the accumulation of $\mathbf{R}_X$ estimation by Eq. (6) is a de-noise process, we can estimate the PN sequence by decomposition of $\hat{\mathbf{R}}_X$ even when γ_{SNR} is lower. However, the memory size and computational speed will become problems when N becomes bigger. Additionally, it is difficult to use this batch method to realize the PN tracking of DS signals. Since we would like to track slowly varying parameters, we must form a moving average estimate of the correlation matrix based on the J most recent observations

$$\hat{\mathbf{R}}_X(i,J) = \frac{1}{J} \sum_{j=i-J+1}^{i} \mathbf{X}(j) \mathbf{X}^T(j) \tag{8}$$

It is well known (Anderson, 1963) that the maximum likelihood estimate of the eigenvalues and associated eigenvectors of $\mathbf{R}_X$ is just the eigenvalue decomposition of $\hat{\mathbf{R}}_X(i,J)$. But there are a lot of difficulties in this tracking process by eigenvalue decomposition for it's a batch method. In the following context, we will propose to use the PCA NN to solve these problems.

4 IMPLEMENTATION OF A MODIFIED SANGER'S GHA NEURAL NETWORKS

According to the result of subspace analysis of DS signals based on K-L transformation, we'll have to extract the first and second principal eigenvectors before realizing the whole PN sequence estimation. A two-layer PCA NN is used to estimate the PN sequence in DS signal blindly as in Fig.1. The number of input neurons is given by $N = T_0 / T_c$.

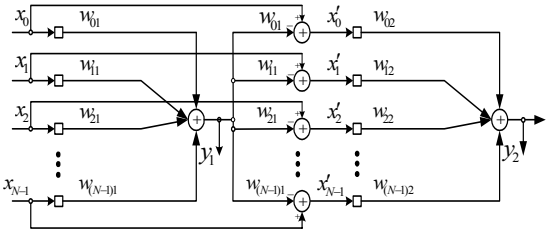


Figure 1: Neural Networks.

Assume $T_x \neq 0$, one of the received signal vectors is

$$\mathbf{X}(t)=\mathbf{X}(k)=\left[x(t),x(t-T_C),\cdots,x\left[t-(N-1)T_C\right]\right]^T \quad (9)$$
$$=\left[x_0(t),x_1(t),\cdots,x_{N-1}(t)\right]^T$$

Where $\{x_i(t)=x(t-iT_C),i=0,1,\cdots,N-1\}$ are sampled by one point per chip. The synaptic weight vector is

$$\mathbf{w}_j(t)=\left[w_{0j}(t),w_{1j}(t),\cdots,w_{(N-1)j}(t)\right]^T \quad (10)$$

Where the sign of $\{w_{ij}(t)$, $i=0,1,\cdots,N-1$, $j=1,2\}$ denotes the 1st and 2nd i-th bit of estimated PN sequence. The output layer of NN has only two neurons, its output is

$$y_j(t)=\sum_{i=0}^{N-1}w_{ij}(t)x_i(t),\quad j=1,2 \quad (11)$$

The synaptic weight $w_{ij}(t)$ is adapted in accordance with a general form of Hebbian learning, as shown by

$$\mathbf{w}_j(t+1)=\mathbf{w}_j(t)+\beta_j y_j(t)\left[\mathbf{X}(t)-\sum_{k=1}^{j}y_k(t)\mathbf{w}_k(t)\right] \quad (12)$$

Where β_j are the positive step-size parameters. In order to achieve good robust convergence performance, we modified β_j in learning rule Eq.(12) of Sanger's GHA as follows

$$\beta_j=1/d_j(t+1),$$
$$d_j(t+1)=B_j d_j(t)+y_j^2(t),\ j=1,2 \quad (13)$$

Where B_j, $j=1,2$, are two positive constants (usually less than 1). Where the Sanger's generalized Hebbian algorithm (GHA) of Eq.(12) for layer of j neurons includes the algorithm of original Hebbian for a single neuron as a special case ,that is, $j=1$.

For a heuristic understanding of how the Sanger's GHA actually operates, we use matrix notation to rewrite the version of the algorithm defined in Eq.(12) as follows

$$\mathbf{w}_j(t+1)=\mathbf{w}_j(t)+\beta_j y_j(t)\left[\mathbf{X}'(t)-y_j(t)\mathbf{w}_j(t)\right] \quad (14)$$

Where

$$\mathbf{X}'(t)=\mathbf{X}(t)-\sum_{k=1}^{j-1}y_k(t)\mathbf{w}_k(t) \quad (15)$$

The vector $\mathbf{X}'(t)$ represents a modified form of the input vector. Provided that the first neuron has already converged to the first principal component, the second neuron sees an input vector $\mathbf{X}'(t)$ from which the first eigenvector of the correlation matrix $\mathbf{R}_X$ has been removed. The second neuron therefore extracts the first principal component of $\mathbf{X}'(t)$, which is equivalent to the second principal component of the original input vector $\mathbf{X}(t)$.

The neuron-by-neuron description above is intended merely to simplify the explanation. In practice, all the neurons in this modified generalized Hebbian algorithm tend to converge together. There is a convergence theorem in (Sanger, 1989) (Haykin, 1999) which can guarantee the convergence of the modified Sanger's GHA NN. It guarantees the GHA NN to find the first j eigenvectors of the correlation matrix $\mathbf{R}_X$. Equally important is the fact that we do not need to compute $\mathbf{R}_X$. Rather, the first j eigenvectors of $\mathbf{R}_X$ are computed by the algorithm directly from the input signal. The resulting computational savings can be enormous especially if the dimensionality N of the input space is very large, and the required number of the eigenvectors associated with the j largest eigenvalues of the $\mathbf{R}_X$ is a small fraction of N . This provides best advantage to track the time-varying PN spreading sequence of DS signals adaptively.

5 SIMULATIONS

The experiments mainly focus on the NN implementation. We get principal eigenvectors and performance curves.

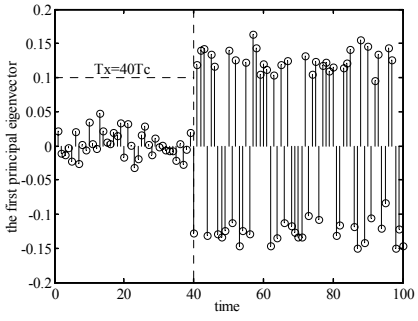


Figure 2: The estimated 1st principal eigenvector.

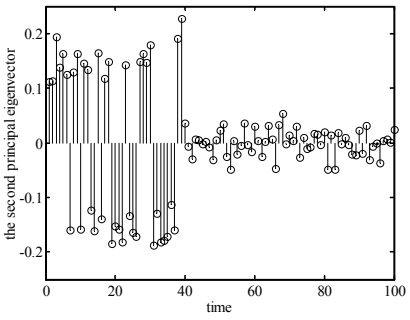


Figure 3: The estimated 2nd principal eigenvector.

Fig.2 and Fig.3 denote the first and second principal eigenvector with N=100bit at Tx=0.4T0. From them, we may estimate the parameter Tx and reconstruct the original PN sequence.

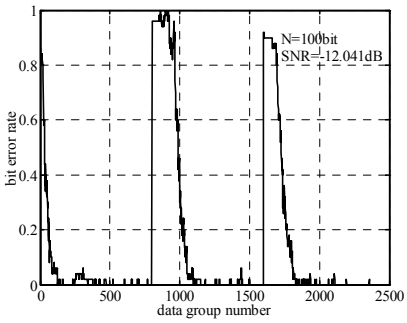


Figure 4: Tthe performance curves of PN tracking.

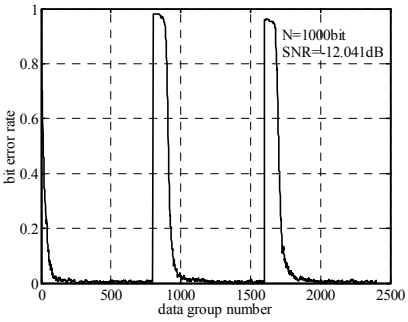


Figure 5: The performance curves of PN tracking.

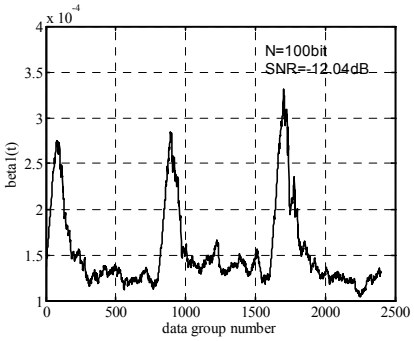


Figure 6: The curve of $\beta_1(t)$.

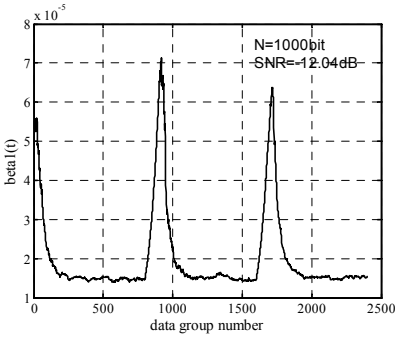


Figure 7: the curve of $\beta_1(t)$.

Fig.4-5 show the tracking performance of the NN under *SNR=-12.04dB* when the length of PN sequence is *N=100bit* and *N=1000bit* respectively. Fig.6-7 show the curves of step-size $\beta_1(t)$ when the case of *N=100bit, SNR=-12.04dB* and *N=1000bit, SNR=-12.04dB*, respectively. Under the same parameters except the length and content of PN sequence, we study the convergence behavior of the NN in signal scenarios with sudden PN sequence changes. We see in Fig.4-7 that when the PN sequence is longer, the convergence and tracking performance is better.

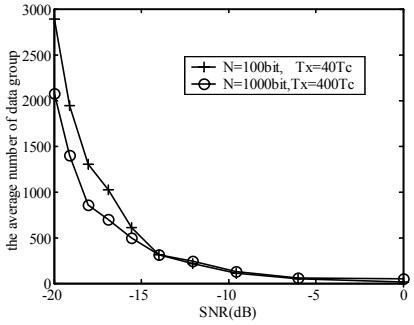


Figure 8: The performance curves of PN estimation.

Fig.8 denotes the performance curves of PN sequence estimation. It shows the time taken for the NN to perfectly estimate the PN sequence for lengths of *N=100bit* and *N=1000bit* at $T_x/T_0=0.4$. Under the same condition, when the longer the PN sequence is, the better the performance is.

6 CONCLUSIONS

A modified Sanger's GHA NN technique for blind estimation and adaptive tracking of PN sequence of DS signals is developed and demonstrated. The technique, referred to here as the modified Sanger's GHA NN algorithm, exploits the subspace analysis based on K-L transformation of the DS signal to blindly estimate and adaptively track the spreading code and can further despread the underlying message sequence, without knowledge of the content of the PN code or message sequences. The technique is applicable to arbitrary spreading codes and message sequences, and can operate in environments containing arbitrary levels of additive white Gaussian noise in theory.

The technique is demonstrated for the length of PN code *N=100bit* and *1000bit* DS-SS signal received in *-20 dB* to *0 dB* of additive white Gaussian noise. It is shown that the technique can blindly estimate and adaptively track the PN sequence in the presence of strong additive white Gaussian noise. In (Simic et al., 2005) Simic used the method of eigen-analysis to achieve –5dB of the *SNR* threshold, moreover, in (Zhan et al., 2005) Zhan use the method of matrix to achieve –12dB *SNR* threshold, but we can realize threshold of $SNR = -20.0dB$ easily here, hence the performance of the methods in this paper is more better. The convergence time of the algorithm for PN sequence perfect estimation is also shown to be competitive with conventional despreading techniques (which require knowledge of the spreading code) such as delay-lock loops.

These results show that modified Sanger's GHA NN technique can provide a promising alternative to existing despreading algorithms. The algorithm can be applicable to signals with short code lengths, such as commercial communication signals. The algorithm can be also applicable to signals with longer code lengths, such as military communication signals. It can be further used in management and scout of DS communications.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China (No.60602057), the Natural Science Foundation of Chongqing University of Posts and Telecommunications (CQUPT) (No.A2006-04 , No.A2006-86), the Natural Science Foundation of Chongqing Municipal Education Commission (No.KJ060509), and the Natural Science Foundation of Chongqing Science and Technology Commission (No. CSTC2006BB2373).

REFERENCES

M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread Spectrum Communications Handbook.* New York: McGraw-Hill, 1994.

C. A. French and W. A. Gardner, "Spread spectrum despreading without the code," *IEEE Trans. Cornmun.,* vol. COM-34, pp. 404-408, Apr. 1986.

Tianqi Zhang, Xiaokang Lin and Zhengzhong Zhou, "Blind Estimation of the PN Sequence in Lower SNR DS/SS Signals , " *IEICE Transaction On Communications*, Vol.E88-B, No.7, JULY, 2005, pp. 3087-3089.

Simic, S. and Zejak, A., "Blind Estimation of the Code Sequence in Spread Spectrum Radar," *the 7th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services, IEEE-TELSIKS, 2005*. Vol.2, 28-30 Sept. 2005, pp: 485 – 490.

Zhan, Y., Cao Z., and Lu J., "Spread-spectrum sequence estimation for DSSS signal in non-cooperative communication systems," *IEE Proc.-Commun*, Vol.152, No.4, 2005, pp.476-480.

T.D. Sanger, "Optimal unsupervised learning in a single-layer linear feedforward neural networks," *Neural Networks*, vol.3, pp.459-473, 1989.

S. Haykin, *Neural Networks—A Comprehensive Foundation*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.

T.W. Anderson. Asymptotic theory for principal component analysis. *Ann. Math. Statist.*, 1963, 35: 1296-1303.

ON THE JOINT ESTIMATION OF UNKNOWN PARAMETERS AND DISTURBANCES IN LINEAR STOCHASTIC TIME-VARIANT SYSTEMS

Stefano Perabò and Qinghua Zhang
IRISA, INRIA Rennes, Campus universitaire de Beaulieu, Avenue du General Leclerc, 35042 Rennes Cedex, France
sperabo@irisa.fr, zhang@irisa.fr

Keywords: Fault detection, parameters estimation, linear stochastic time varying systems, adaptive signal processing.

Abstract: Motivated by fault detection and isolation problems, we present an approach to the design of unknown parameters and disturbances estimators for linear time-variant stochastic systems. The main features of the proposed method are: (a) the joint estimation of parameters and disturbances can be carried out; (b) it is a full-stochastic approach: the unknown parameters and disturbances are random quantities and prior information, in terms of means and covariances, can be easily taken into account; (c) the estimator structure is not fixed *a priori*, rather derived from the optimal infinite dimensional one by means of a sliding window approximation. The advantages with respect to the widely used *parity space* approach are presented.

1 INTRODUCTION

The following discrete time linear stochastic system is considered in this brief paper:

$$\mathrm{x}_{k+1} = A_k \mathrm{x}_k + B_k \mathrm{u}_k + \Psi_k \mathrm{p} + E_k \mathrm{d}_k + \mathrm{w}_k \quad (1a)$$

$$\mathrm{y}_k = C_k \mathrm{x}_k + \mathrm{v}_k \quad (1b)$$

for $k \geq 0$, with $A_k \in \mathbb{R}^{n\times n}$, $B_k \in \mathbb{R}^{n\times m}$, $\Psi_k \in \mathbb{R}^{n\times q}$, $E_k \in \mathbb{R}^{n\times f}$ and $C_k \in \mathbb{R}^{l\times n}$ known time-variant matrices. The vector sequences $\{\mathrm{x}_k\}$, $\{\mathrm{u}_k\}$ and $\{\mathrm{y}_k\}$ denote respectively the state, input and output stochastic processes. The sequences $\{\mathrm{w}_k\}$ and $\{\mathrm{v}_k\}$ are assumed to be zero mean, white and uncorrelated wide-sense stochastic processes, with $\mathbb{E}[\mathrm{w}_k \mathrm{w}_k^T] = Q_k$ and $\mathbb{E}[\mathrm{v}_k \mathrm{v}_k^T] = R_k \succ O$ (positive definite), where $\mathbb{E}[\cdot]$ denotes the mathematical expectation operator. The initial condition x_0 has known mean $\mathbb{E}[\mathrm{x}_0] = \mu_0$ and covariance $\mathbb{E}[(\mathrm{x}_0 - \mu_0)(\mathrm{x}_0 - \mu_0)^T] = P_0$. Both the initial condition x_0 and the input process $\{\mathrm{u}_k\}$ are assumed uncorrelated with the noise sequences.
The term $E_k \mathrm{d}_k$ accounts for unknown disturbances acting on the system or faults, whence the sequence $\{\mathrm{d}_k\}$ is an unknown (and uncontrolled) input modeled as a wide-sense stochastic process, not necessarily stationary. The disturbances are further assumed uncorrelated with the initial state, the noise and the input processes, respectively x_0, $\{\mathrm{w}_k\}$, $\{\mathrm{v}_k\}$ and $\{\mathrm{u}_k\}$. Finally, the term $\Psi_k \mathrm{p}$ can account for the occurrence of parametric faults in the system (for instance with the meaning that when p is zero no faults are present) or for constant parameters that need to be estimated on-line. Here p is a random variable uncorrelated with the noise, input and disturbance processes.

The problem to be solved is the following: find for each $N \geq 0$ the *minimum variance unbiased linear estimators* of the disturbances sequence $\mathrm{d}_0^{N-1} = \{\mathrm{d}_k : 0 \leq k \leq N-1\}$ and of the parameters p, given the input and output sequences u_0^{N-1} and y_0^N, and the conditions guaranteeing the uniqueness of the corresponding estimates. These estimators will be denoted respectively by $\hat{\mathrm{d}}_{k|N}$ and $\hat{\mathrm{p}}_{|N}$ (since p does not depend on time).

The following two related problems will also be discussed in this paper. First, how to weaken the uniqueness conditions by considering the quantities $\hat{\mathrm{d}}_{k|N+D}$ for $0 \leq k \leq N-1$ and some appropriate delay $D > 0$, which will be called, with an abuse of terms, "delayed estimators". Second, how to *recursively* and reliably compute the estimates $\hat{p}_{|N+D}$ and $\hat{d}_{k|N+D}$ once sample paths (measurements) u_0^{N+D-1} and y_0^{N+D} of the input and output processes are becoming available (by convention, italic characters will denote samples from the corresponding random variables which, in-

stead, will be denoted by roman characters).

The solution proposed in this work shares many similarities with the so called *parity space* method (Chow and Willsky, 1984; Gustafsson, 2001) which finds wide application in fault detection problems. However it has some advantageous features that will be presented at the end of the exposition.

Once the disturbances and parameters estimates have been computed, state estimation becomes straightforward and can also be easily performed *on demand*. This topic is discussed in (Perabò and Zhang, 2007).

2 BASIC EQUATIONS FOR ESTIMATION

Pretend for a while that the parameters and the disturbances sequence are known quantities, i.e. as if they were inputs of the system described by (1), and assume the following:

Assumption 1. (A_k, C_k) is *uniformly completely observable* and $(A_k, Q_k^{1/2})$ is *uniformly completely reachable*.

Assumption 2. The parameters p and the disturbance sequence $\{\mathrm{d}_k\}$ are uncorrelated from the initial state x_0 and the noise sequences $\{\mathrm{w}_k\}$ and $\{\mathrm{v}_k\}$.

Hence there is *no feedback* from the output to the parameters and disturbances (see (Gevers and Anderson, 1982) for details) and by applying well known results of the linear estimation theory (Kailath et al., 2000), the following *innovation representation* of the output process $\{\mathrm{y}_k\}$ can be derived:

$$\hat{\mathrm{x}}^*_{k+1|k} = A_k\hat{\mathrm{x}}^*_{k|k-1} + B_k\mathrm{u}_k + \Psi_k\mathrm{p} + E_k\mathrm{d}_k + K_k\mathrm{e}^*_k \quad (2a)$$

$$\mathrm{y}_k = C_k\hat{\mathrm{x}}^*_{k|k-1} + \mathrm{e}^*_k, \quad (2b)$$

the recursion being initiated setting $\hat{\mathrm{x}}^*_{0|-1} = \mathrm{x}_0$, where $\hat{\mathrm{x}}^*_{k+1|k}$ is the one step *minimum variance unbiased linear predictor* of the state. Each term of the *innovation sequence* $\{\mathrm{e}^*_k\}$ has zero mean and covariance Λ_k given by the recursive solution of the same Riccati equation which is solved in the standard Kalman filter (i.e. with no disturbances and unknown parameters). With respect to this one, however, the superscript * in (2) emphasizes that the "estimates" $\{\hat{x}^*_{k+1|k}\}$ cannot be computed because the realizations p and $\{d_k\}$, of p and $\{\mathrm{d}_k\}$ respectively, are not really available. Also the gains K_k are computed exactely as in the Kalman filter. By defining recursively the quantities,

$$\Upsilon_0 = O \quad \Upsilon_{k+1} = (A_k - K_kC_k)\Upsilon_k + \Psi_k \quad (3a)$$

$$\mathrm{s}_0 = 0 \quad \mathrm{s}_{k+1} = (A_k - K_kC_k)\mathrm{s}_k + E_k\mathrm{d}_k \quad (3b)$$

$$\mathrm{z}_0 = \mathrm{x}_0 \quad \mathrm{z}_{k+1} = (A_k - K_kC_k)\mathrm{z}_k + B_k\mathrm{u}_k + K_k\mathrm{y}_k \quad (3c)$$

and by using (2b) it is not difficult to check that the following is true:

$$C_k\mathrm{s}_k + C_k\Upsilon_k\mathrm{p} + \mathrm{e}^*_k = \mathrm{y}_k - C_k\mathrm{z}_k. \quad (4)$$

Note that a realization of the sequence $\{\mathrm{z}_k\}$ can be computed from available data only, i.e. system matrices, input and output sequences. As a matter of fact, (3c) is exactly the Kalman filter equation that would be obtained if $\mathrm{p} \equiv 0$ and $\mathrm{d}_k \equiv 0$ for all k.
It is possible to arrange in matrix form the set of equations obtained from (4) when $k = 1, 2, \ldots, N$. For example for $N = 4$ one obtains

$$\begin{bmatrix} C_4\Phi_1^4E_0 & C_4\Phi_2^4E_1 & C_4\Phi_3^4E_2 & C_4E_3 & C_4\Upsilon_4 \\ C_3\Phi_1^3E_0 & C_3\Phi_2^3E_1 & C_3E_2 & O & C_3\Upsilon_3 \\ C_2\Phi_1^2E_0 & C_2E_1 & O & O & C_2\Upsilon_2 \\ C_1E_0 & O & O & O & C_1\Upsilon_1 \end{bmatrix} \begin{bmatrix} \mathrm{d}_0 \\ \mathrm{d}_1 \\ \mathrm{d}_2 \\ \mathrm{d}_3 \\ \mathrm{p} \end{bmatrix} + \begin{bmatrix} \mathrm{e}^*_4 \\ \mathrm{e}^*_3 \\ \mathrm{e}^*_2 \\ \mathrm{e}^*_1 \end{bmatrix} = \begin{bmatrix} \mathrm{y}_4 - C_4\mathrm{z}_4 \\ \mathrm{y}_3 - C_3\mathrm{z}_3 \\ \mathrm{y}_2 - C_2\mathrm{z}_2 \\ \mathrm{y}_1 - C_1\mathrm{z}_1 \end{bmatrix}, \quad (5)$$

where the transition matrices Φ_h^k are defined by

$$\Phi_h^h = I, \qquad \Phi_h^{k+1} = (A_k - K_kC_k)\Phi_h^k. \quad (6)$$

For an arbitrary N, left multiply the above system by the block diagonal matrix $\mathrm{blkdiag}\{\Lambda_N^{-1/2}, \ldots, \Lambda_1^{-1/2}\}$ in such a way that the covariance of the zero mean vector $\mathrm{e}^* = \mathrm{vec}[\Lambda_N^{-1/2}\mathrm{e}^*_N \ \ldots \ \Lambda_1^{-1/2}\mathrm{e}^*_1]$ is equal to the identity matrix. A system of the form

$$\mathsf{A}\mathsf{g} + \mathsf{e}^* = \mathsf{r} \quad (7)$$

is thus obtained, where the matrix $\mathsf{A} \in \mathbb{R}^{lN\times(fN+q)}$ has the same structure as in (5), $\mathsf{g} = \mathrm{vec}[\mathrm{d}_0 \ \ldots \ \mathrm{d}_{N-1} \ \mathrm{p}]$ is the unknown term, and the vector $\mathsf{r} = \mathrm{vec}[\mathrm{r}_N \ \ldots \ \mathrm{r}_1]$ contains the computable *residuals*

$$\mathrm{r}_k = \Lambda_k^{-1/2}(\mathrm{y}_k - C_k\mathrm{z}_k). \quad (8)$$

If $\mathrm{d}_k \equiv 0$ for each k and $\mathrm{p} \equiv 0$, then $\mathsf{r} = \mathsf{e}^*$, i.e. the vector of residuals has zero mean and its covariance equals the identity matrix. Any statistical test indicating a deviation from this condition can be used to detect the presence of non-null disturbances and/or parameters.

Since samples of r are available but instead e cannot be observed, the most appealing approach to estimate g is to compute its minimum variance linear estimator $\hat{\mathsf{g}}$ given the random vector r. Thanks to the Assumption 2, the following holds:

$$\mathbb{E}[\mathrm{e}^*_k\mathrm{d}_h^T] = O \quad \mathbb{E}[\mathrm{e}^*_k\mathrm{p}^T] = O \quad \forall k, h \geq 0. \quad (9)$$

As a result, g and e^* in (7) are in fact uncorrelated. Provided that *prior information* on the random vector g is given in terms of its mean μ_{g} and covariance Σ_{g} (assume Σ_{g} invertible and the factorization $\Sigma_{\mathrm{g}}^{-1} = \mathsf{B}^T\mathsf{B}$), a straightforward application of linear estimation formulas shows that $\hat{\mathrm{g}}$ and the covariance of the error $\tilde{\mathrm{g}} = \mathrm{g} - \hat{\mathrm{g}}$ can be obtained from

$$\left(\mathsf{A}^T\mathsf{A} + \mathsf{B}^T\mathsf{B}\right)(\hat{\mathrm{g}} - \mu_{\mathrm{g}}) = \mathsf{A}^T(\mathrm{r} - \mathsf{A}\mu_{\mathrm{g}}) \qquad (10a)$$

$$\Sigma_{\tilde{\mathrm{g}}} = \left(\mathsf{A}^T\mathsf{A} + \mathsf{B}^T\mathsf{B}\right)^{-1}. \qquad (10b)$$

One could suspect, at this point, that the *information* about the unknown terms which is available from knowledge of the input and output sequences, is not fully exploited if the only quantities that are used for the estimation of the disturbances and parameters are the residuals defined in (8). However, as long as linear estimators are considered, it is possible to prove that the proposed method is *optimal* in the sense that, by estimating g from (7) (instead of a different linear relation with the measurable sequences $\{\mathrm{u}_0^{N-1}, \mathrm{y}_0^N\}$) one in fact minimizes the estimation error variance.

When sample paths of the input and output sequences, say $\{u_0^{N-1}\}$ and $\{y_0^N\}$, are available, one is faced to the problem of computing numerically the estimate $\hat{g} = \mathrm{vec}[\hat{d}_{0|N} \ \ldots \ \hat{d}_{N-1|N} \ \hat{p}_{|N}]$ from the vector r denoting the realization of r. To this end, the availability or lack of prior information makes a difference. In the following the latter case is discussed.

3 NO PRIOR INFORMATION

3.1 Estimability Conditions

The absence of prior information about g can be dealt with by setting $\mu_{\mathrm{g}} = 0$ and letting $\Sigma_{\mathrm{g}} \to \infty$ (or equivalently $\Sigma_{\mathrm{g}}^{-1} \to 0$) which corresponds to a very large uncertainty. Formula (10a) becomes $\left(\mathsf{A}^T\mathsf{A}\right)\hat{\mathrm{g}} = \mathsf{A}^T\mathrm{r}$ which is the system of *normal equations* for computing the unique *least squares solution* of

$$\mathsf{A}g = r. \qquad (11)$$

in the unknown g, provided that the matrix A has full column rank. From a practical point of view, It should be noted that the proposed method requires simply checking the rank of matrices and solving least squares problems, for which efficient numerical tools are readily available. But, unfortunately, finding general estimability condition in analytic form, is a very complex task. The following is not difficult to prove:

Proposition 1. *For a given $N \geq 1$, the estimates $\hat{p}_{|N}$ and $\hat{d}_{k|N}$ for $0 \leq k \leq N-1$ are unique if and only if the matrix* A *in* (11) *has full column rank. Moreover, the uniqueness holds only if the following necessary conditions are satisfied:*

$$\text{(C1)} \quad \mathrm{rank}\begin{bmatrix} E_0 & O & \cdots & O & \Psi_0 \\ O & E_1 & \cdots & O & \Psi_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ O & O & O & E_{N-1} & \Psi_{N-1} \end{bmatrix} = rN + q \quad (12a)$$

$$\text{(C2)} \quad \mathrm{rank}\left(\sum_{k=1}^{N} \Upsilon_k^T C_k^T C_k \Upsilon_k\right) = q. \qquad (12b)$$

If $\mathrm{rank}(E_k) = f$ *for all $k \geq 0$ and* (C1) *is true for a value $N = N_{min}$, then it is satisfied for all values $N \geq N_{min}$. Analogously, if* (C2) *is true for a value $N = N_{min}$, then it is satisfied for all values $N \geq N_{min}$.*

3.2 Delayed Estimation

Consider first the case when there are *no unknown parameters* ($q = 0$). A *sufficient* (but *not necessary*) condition to ensure that A has full column rank for all $N \geq 1$, hence the uniqueness of the estimates $\hat{d}_{k|N}$ for $0 \leq k \leq N-1$, is the following:

$$\text{(C3)} \qquad \mathrm{rank}(C_{k+1}E_k) = f \qquad \forall k \geq 0. \qquad (13)$$

However, when (C3) is not satisfied, it could still be possible to compute, for some delay $D > 0$, unique *delayed estimates* $\hat{d}_{k|N+D}$ for $0 \leq k \leq N-1$. To exemplify what has been just asserted, consider the case $C_{k+1}E_k = O$ and thus (C3) is not satisfied (this situation may happen typically when C_{k+1} and E_k have both some zero entries, for example $C_{k+1} = [1\ 0]$ and $E_k = [0\ 1]^T$). Then the zero blocks appear in the term Ag in (7) as shown in the following scheme (suppose, for example, that $N = 4$):

$$\begin{array}{r|c|c|} & & \\ 5 & \times & \times & \times & * & O \\ \hline N=4 & \times & \times & * & O & O \\ 3 & \times & * & O & O & O \\ 2 & * & O & O & O & O \\ 1 & O & O & O & O & O \end{array} \begin{bmatrix} \mathrm{d}_0 \\ \mathrm{d}_1 \\ \mathrm{d}_2 \\ \mathrm{d}_3 \\ \hline \mathrm{d}_4 \end{bmatrix}.$$

It is evident that d_{N-1} (d_3 in the example above) is not estimable from measurements collected till time N (in other words $\hat{d}_{3|4}$ is not unique). However, if the blocks marked with a $*$, i.e. the matrices $C_{k+2}\Phi_{k+1}^{k+2}E_k$ in (7), have full column rank, it is sufficient to add the measurements at time $N+1$ (at time 5 to continue the example) so that the unique estimates $\hat{d}_{k|N+1}$ for $k = 0, \ldots, N-1$ and, in particular $\hat{d}_{N-1|N+1}$ (in the example $\hat{d}_{3|5}$), could be computed. The above argument can be generalized as follows: if for some $D > 0$ the conditions

$$\text{(C4a)} \quad \mathrm{rank}(C_{k+D+1}\Phi_{k+1}^{k+D+1}E_k) = f \ \ \forall k \geq 0 \quad (14a)$$

$$\text{(C4b)} \quad \begin{bmatrix} C_{k+D}\Phi_{k+1}^{k+D}E_k \\ \cdots \\ C_{k+2}\Phi_{k+1}^{k+2}E_k \\ C_{k+1}E_k \end{bmatrix} = O \qquad (14b)$$

are satisfied, then the estimates $\hat{d}_{k|N+D}$ for $0 \leq k \leq N-1$ are unique (even if A has not full rank).

When there are unknown parameters ($q > 0$), the conditions in (13) or (14) are no longer sufficient and, in general, the rank of the matrix A has to be checked numerically. However note the following result:

Proposition 2. *(a) Assuming that condition* (C3) *in* (13) *is satisfied, if the estimates $\hat{p}_{|N}$ and $\hat{d}_{k|N}$ for $0 \leq k \leq N-1$ are unique (i.e. the matrix A has full column rank) for a value $N = N_{min}$, then they are unique also for all $N \geq N_{min}$.*
(b) Analogously, assuming that conditions (C4) *in* (14) *are satisfied, if the delayed estimates $\hat{p}_{|N+D}$ and $\hat{d}_{k|N+D}$ for $0 \leq k \leq N-1$ are unique for a value $N = N_{min}$, then they are unique also for all $N \geq N_{min}$.*

3.3 Approximate Recursive Estimation

In order to compute the estimates from (11), a growing size least squares problem as to be solved as N increases. Observe, however, that the upper left blocks of the matrix A tend to zero as N grows, because the uniform observability and reachability assumption guarantees that the transition matrices Φ_h^k defined in (6) tend to the null matrix as the difference $k-h \to \infty$. Hence, it is natural to consider an approximate problem by replacing A with $\mathsf{A}+\mathsf{E}$, where E annihilates the blocks $\Lambda_k^{-1/2} C_k \Phi_h^k E_{h-1}$ such that $k-h \geq L \geq L_{min}$, where $L_{min} \geq 1$ is the minimum value guaranteeing that $\mathrm{rank}(\mathsf{A}) = \mathrm{rank}(\mathsf{A}+\mathsf{E})$ for all N, so that the estimability properties of the original problem are conserved also in the approximate one. Obviously, the accuracy of the approximate solution increases as $L \to \infty$. The system $(\mathsf{A}+\mathsf{E})g = r$ has thus the *banded* structure shown in the following scheme (for $N = 5$ and $L = 3$):

$$\begin{bmatrix} \times & \times & \times & \times & \\ \times & \times & \times & & \times \\ \boxed{\times} & \boxed{\times} & \boxed{\times} & & \boxed{\times} \\ & \boxed{\times} & \boxed{\times} & & \boxed{\times} \\ & & \boxed{\times} & & \end{bmatrix} \cdot \begin{bmatrix} \boxed{d_0} \\ \boxed{d_1} \\ \boxed{d_2} \\ d_3 \\ d_4 \\ \boxed{p} \end{bmatrix} = \begin{bmatrix} r_5 \\ r_4 \\ \boxed{r_3} \\ \boxed{r_2} \\ \boxed{r_1} \end{bmatrix}$$

In the above, also an initial data window has been indicated with a solid line box. Using the numerical techniques described for example in (Björck, 1996, Chapter 6.2), this approximate least squares problem can then be solved recursively using a sliding window procedure.

3.4 Comparison with the Parity Space Approach

In the parity space method, the parameters and disturbances are estimated from a set of relations which can be cast in the form $\bar{\mathsf{A}}\mathsf{g} + \mathsf{w} = \bar{\mathsf{r}}$. The matrix $\bar{\mathsf{A}}$ differs from A in (7) only because the transition matrices Φ_h^k defined in (6) are replaced by $\Gamma_h^k = A_{k-1} \ldots A_{h+1} A_h$. Moreover, the covariance of the noise term w does not equal the identity matrix and the residuals $\bar{\mathsf{r}}$ are built in a different way.
The approach proposed here is new in that it makes explicit reference to the innovation representation of the system (1), with the following advantages:

(a) The components of the noise term e^* are independent and normalized, while an important drawback of the parity space approach is that the covariance of the noise term w has to be *whitened* before computing the least squares estimate, thus increasing the computational load, especially for large scale problems.

(b) If the matrices A_k are not stable, as it can happen typically in control problems, the matrix $\bar{\mathsf{A}}$ could be largely ill-conditioned, thus making numerically harder the process of computing reliably the estimate, especially for large window sizes.

(c) The initial condition x_0 affects the residuals r through the sequence $\{\mathrm{z}_k\}$. However the transition matrices Φ_h^k are stable. Hence the effect of the initial condition is asymptotically forgot as $k \to +\infty$. As a consequence, when using the sliding window estimation procedure, one has not to take care of the estimation or rejection of the state at the initial time of the window as happens for the parity space approach (Törnqvist and Gustafsson, 2006).

REFERENCES

Björck, A. (1996). *Numerical Methods for Least Squares Problems*. SIAM.

Chow, E. Y. and Willsky, A. S. (1984). Analytical redundancy and the design of robust failure detection systems. *IEEE T. Automat. Contr.*, AC-29(7):603–614.

Gevers, M. R. and Anderson, B. D. O. (1982). On jointly stationary feedback-free stochastic processes. *IEEE T. Automat. Contr.*, AC-27(2):431–436.

Gustafsson, F. (2001). *Adaptive Filtering and Change Detection*. John Wiley & Sons, Ltd.

Kailath, T., Sayed, A. H., and Hassibi, B. (2000). *Linear Estimation*. Prentice Hall.

Perabò, S. and Zhang, Q. (2007). Adaptive observers for linear time-variant stochastic systems with disturbances. In *Proceedings of the European Control Conference 2007*, Kos, Greece. (accepted for publication).

Törnqvist, D. and Gustafsson, F. (2006). Eliminating the initial state for the generalized likelihood ratio test. In *Proceedings of the 6^{th} IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pages 643–648, Beijing, P. R. China.

SEARCHING AND FITTING STRATEGIES IN ACTIVE SHAPE MODELS

Jianhua Zhang
College of Information Engineering, Zhejiang University of Technology, Hangzhou, China
Jx.zhangjianhua@gmail.com

S. Y. Chen
Department of Informatics, University of Hamburg, Germany
sy@ieee.org

Sheng Liu, Qiu Guan, Haihong Wu
College of Information Engineering, Zhejiang University of Technology, Hangzhou, China

Keywords: Active Shape Model, shape parameter, self-adjustment, active searching, image fitting, outlying points.

Abstract: The Active Shape Model (ASM) is an ever-increasingly important method for object modelling, shape recognition, and image localization. However, when the target is not clarity, or the initial model placed very far from the target, ASM may have problem to locate on an acceptable result. In this paper, a new strategy is proposed on the ASM searching and fitting procedure, which forms an active searching method. Using this new strategy, the influence of the clarity of the target and initial position of the model is reduced and the result of the fitting is more accuracy. Experiments and results show that the new strategy are effective for improving the performance of the image fitting.

1 INTRODUCTION

The Active Shape Model (ASM), firstly proposed by Cootes et.al. (Cootes et.al, 1995), is an important method for modelling of a deformable model, image fitting, shape recognition, and shape localization.

For the traditional ASM, it performs successfully when the target image is clear and the initial position of the model closes to the target. However, when the target is not clarity, for example, the beard in a facial image or a weak boundary in medical images, or the initial model placed very far from the target, ASM may have problem to locate on an acceptable result. Fig. 1 illustrates a few situations of these scenes.

Figure 1: Problems occurred in ASM fitting to an image (the left is the initial pose and the right is the fitting result).

Some improvements were put forward to ASM in recent years. (Bruijne et al. 2001,Wan et al. 2005,Wang et al. 2002). However, ASMs still need an appropriate initial position and clear target images. Li and Chutatape presented the self-adjusting weight in the transformation from

shape space to image space and the exclusion of outlying points in obtaining shape parameters (Li and Chutatape 2003). This modification is more robust and converge faster than the conventional ASM in the case where the edge of optic disk is weak or occluded by blood vessels, but it is uneasy to extend to other cases.

In this paper, a novel method is presented that avoid the influence of the targe and initial position of model by using the minimize square error(MSE) obtained by an equation as defined like (Cootes et al.1995):

$$f(d) = (h(d) - \bar{y}_j)^T C_{y_j}^{-1} (h(d) - \bar{y}_j) \quad (1)$$

If the MSEs of some points are too large, we ignore these points when the shape and pose parameters are attained. The organization of the remainder paper is as follows. Section 2 outlines the standard ASM procedure. Some new strategies are developed and described in Section 3. Practical experimental and results are given in Section 4 and a conclusion is followed in Section 5.

2 THE ACTIVE SHAPE MODEL

ASM have been helpful for image fitting, shape recognition, and shape localization. Because models are built by the training set, instance of an ASM can only deform in the ways found in its training set. The local structures are also considered by ASM through modelling the gray-level information of each landmark.

2.1 The Point Distribution Model (PDM)

PDM is built to describe both typical shape and typical variability by disposing a training set which we have chosen. Firstly, landmarks are labelled on each image in the training set by hand.

After all images in the training set have been labeled, they must be aligned with respect to a set of axes. The required alignment is achieved by scaling, rotating, and translating the training shapes in order that they agree as closely as possible.

And then we capture the statistics of the set of aligned shape. In previous steps, labeling and aligning shapes, the mean shape is obtained by the equation which defined as following:

$$\bar{X} = \frac{1}{N}\sum_{i=1}^{N} X_i \quad (2)$$

where the N is the number of the shapes in the training set. The X_i is a vector which represents the coordinates of landmarks in the i-th shape:

$$X_i = [x_{i1}, y_{i1}, x_{i2}, y_{i2}, \ldots, x_{im}, y_{im}] \quad (3)$$

In Eq.(3), m is the number of the points on one shape. Afterwards, we calculate the covariance of the training set as following:

$$S = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - \bar{X})(X_i - \bar{X})^T \quad (4)$$

And then the eigenvectors ϕ_i and corresponding eigenvalues λ_i are computed. Now we can approximate any shapes in the training set, X, by using

$$X \approx \bar{X} + \Phi b \quad (5)$$

where $\Phi = (\phi_1 | \phi_2 \ldots | \phi_n)$ and b is a N dimensions vector which can be given by

$$b = \Phi^T (X - \bar{X}) \quad (6)$$

For reducing the dimensions of the model and variations, the Principal Components Analysis (PCA) is employed. Thus we get the t dimensions vector b. Eq (5) can be modified as following:

$$X \approx X' = \bar{X} + Pb \quad (7)$$

where b is the t dimensions vector, and P is the corresponding eigenvectors. And now we can present a new shape that is similar to the shape in the training set as the following:

$$X = \bar{X} + dX_i \quad (8)$$

where $dX_i = Pb$. And

$$b = P^T (X - \bar{X}) \quad (9)$$

2.2 Extract Gray-level Information and Image Fitting

After the PDM is built, the new points are found by modelling gray-level appearance on the target image which present the object and transform the model into a new better location.

We consider the gray-level values along a line passing through the landmark in perpendicular to the boundary formed by the landmark and its neighbours. Gray-level profile g_{ij} is extracted from n_p pixels that are centred at the landmark for each landmark point j in the image i of the training set. We get the gray-level profile g_{ij} as following:

$$g_{ij} = [g_{ij0}, g_{ij1}, \ldots, g_{ij\,np-1}]^T \quad (10)$$

$$dg_{ij} = [g_{ij1} - g_{ij0}, g_{ij2} - g_{ij1}, \ldots, g_{ij\,np-1} - g_{ij\,np-2}]^T \quad (11)$$

Here, the mean values are calculated as following :

$$\bar{y}_{ij} = \frac{1}{N}\sum_{i=1}^{N} dg_{ij} \quad (12)$$

Now the gray-level information has been modelled, and for each landmark, there is a certain profile $\bar{y}_j$. To transform the mean shape into target object , the target points responded to the points in the model must be found, according to the modelled grey-level information, then the model transforms to the new model that form by target points, but this transformation is restricted by the shape parameters b which is defined in Eq(9). In this way, the new shape will not bring too large distortion to represent the object shape in almost situation. However, in the standard ASM, there are some instances that the new shape will occur the too large distortion and this distortion will lead the fitting process to a failure result (Ghassan Hamarneh).

3 SELECTION CRITERION

For fitting the images into a good shape model, we must have a good strategy to exclude some outlying points and select good target points. This is done with a MSE criterion. When we calculate the shape and pose parameters, such as the scalars $ds, d\theta, dt$, we need to move our current estimate X_i as close as possible to $X_i + dx_i$. Within this process, however, if some points do not match the target object well or their movements keep away from the target object, it will lead to the bad direction in the image fitting, when current estimate X_i closes to $X_i + dx_i$. For avoiding this situation, we consider to exclude those points which are denoted as the outlying points. The initial shape outlying these dissociated points is denoted as X_i', and the target shape is denoted as $(X_i + dx_i)t$. In this paper, we implement such a strategy as following:

(1) Firstly, the MSE of each point is calculated by eq(1). And the msei is defined as:

$$mse_i = f(d) = (h(d) - \bar{y}_j)^T C_{y_j}^{-1} (h(d) - \bar{y}_j) \quad (13)$$

(2) When the MSE of each point, mse_i, is obtained, we sign the point that the mse_i value is large than h (e.g. h=1.5) times of the mean of all the mse_i. And then we exclude these points and get the new initial shape X_i' and the new target shape $(X_i + dx_i)'$.

(3) Then X_i' is aligned to $(X_i + dx_i)t$ and obtain the shape and pose parameters, $ds, d\theta, dt$.

(4) The shape parameters are calculated without the influence of those dissociated points and they can be used to transform X_i into new shape.

Fig. 2 illustrates that the new shape is affected by excluding the outlying points. It shows that the new shape (green line) with outlying points excluded is obviously improved since the dissociated points do not involved in shape formation anymore.

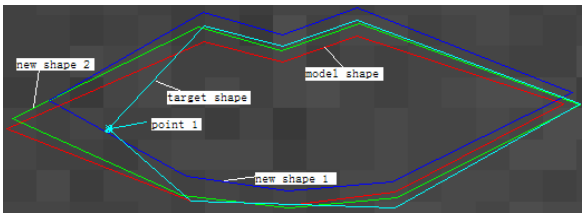


Figure 2: The red line marked by 'model shape' is the initial shape. The cyan line marked by 'target shape' is the target shape. The green line marked by 'new shape 2' is the new shape fitted from no outlying points. The blue line marked by 'new shape 1' is the new shapefitted with the outlying points. And the point marked by 'point 1' is an example of outlying points that should be excluded.

4 EXPERIMENTS

4.1 Data Set

To evaluate our method, 400 facial images are used to build the PDM in experiments. On facial images, we labelled the lip with eight landmarks for each image. Fig. 4 illustrates these landmarks.

4.2 Experimental Result

In the experiments, we adopt the leave-one-out strategy in order to evaluate the performance more accurately and sufficiently. When each facial image is been fitting, the remaining 399 facial images are utilized to establish the PDM. And the same way is performed in anklebone images. Fig. 4 illustrates the search result of the new strategies and the traditional ASM.

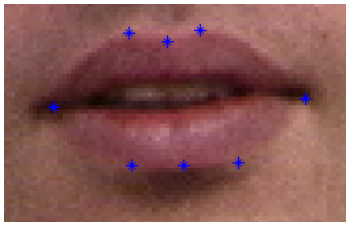


Figure 3: landmarks of the facial image.

(a) (b) (c)

Figure 4: Comparison of the searching results. Column (a) is the standard model and its initial place. (b) Fitting results with the standard ASM. (c) Fitting results with new strategy.

5 CONCLUSION

In this paper, to enhance the robustness and accuracy of image fitting, we propose a new strategy on the Active Shape Model (ASM) method. The main advantages are obvious from observation of practical experiments. For example, according to the MSE that is obtained at the process of the image fitting, the outlying points whose corresponding MSE are too large is excluded for forming a new shape. These outlying points are brought by those target images that are not clarity with some interferential object and the new strategy can avoid effectively the influence of outlying points. By comparison with practical implementation, the proposed strategy works satisfactorily.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China [NSFC-60405009, 60605013], [ZJNSF-Y105101, Y104185], and a grant for Key Research Items from the Dept of Science and Technology of Zhejiang Province [2006C21002]. S. Y. Chen is a research fellow of the Alexander von Humboldt Foundation, Germany.

REFERENCES

T. F. Cootes, C.J.Taylor, D. Cooper, and J. Graham.: Active shape models--their training and application. Computer vision and image understanding, 61(1): pp38-59, 1995.

Marleen de Bruijne, Bram van Ginneken, Wiro J. Niessen, and Max A. Viergever: Active shape model segmentation using a non-linear appearance model: application to 3D AAA segmentation. IEEE Transactions on Medical Imaging, 2001

Kwok-Wai Wan, Kin-Man Lam, Kit-Chong Ng: An accurate active shape model for facial feature extraction. Pattern Recognition Letters 26 (2005) 2409 –2423

Wei Wang, Shiguang Shan, Wen Gao, Bo Cao, Baocai Yin: An Improved Active Shape Model for Face Alignment. Multimodal Interfaces, 2002. Proceedings. Fourth IEEE International Conference on

Huiqi Li, Opas Chutatape: Boundary detection of optic disk by a modified ASM method. Pattern Recognition 36 (2003) 2093 – 2104

Lu Huchuan, Shi Wengang: Accurate Active shape model for face alignment. Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05)

Chai Xiujuan, Shan Shiguang, Gao Wen, Chen Xilin: Example-Based Learning for Automatic Face Alignment. Journal of Software(2005) Vol.16,No.5

A. Hill, T.F. Cootes, C.J. Taylor: Active Shape Models and the shape approximation problem. Image and Vision Computing 14 (1996) 601-607

Ghassan Hamarneh.Active shape models, modeling shape variations and gray level information and an application to image search and classification [EB/OL].http://www.ae.chalmers.se/ ~ jessi/

HUMAN-SCALE VIRTUAL REALITY CATCHING ROBOT SIMULATION

Ludovic Hamon, François-Xavier Inglese and Paul Richard
Laboratoire d'Ingénierie des Systèmes Automatisés, Université d'Angers
62 Avenue Notre Dame du Lac, 49000 Angers, France
ludovic.hamon@univ-angers.fr, inglese@istia.univ-angers.fr, paul.richard@univ-angers.fr

Keywords: Virtual reality, large-scale virtual environment, human-robot interaction, catching.

Abstract: This paper presents a human-scale virtual reality catching robot simulation. The virtual robot catches a ball that users throw in its workspace. User interacts with the virtual robot using a large-scale bimanual haptic interface. This interface is used to track user's hands movements and to display weight and inertia of the virtual balls. Stereoscopic viewing, haptic and auditory feedbacks are provided to improve user's immersion and simulation realisms.

1 INTRODUCTION

Roboticians tried to solve the problem of moving object catching (dynamic problem) while basing themselves on the use a priori of the trajectory of the object to limit the calculating time.

Most of the proposed methods rest generally on the following stages:

1) the detection of the ball,
2) the determination since it is in flight,
3) the follow-up and the prediction of its trajectory
4) the economic planning and the execution of a movement of interception.

Indeed, the prediction of balls trajectories in a controlled environment (no wind, etc.) is based on a priori knowledge of characteristics of this type of movement and, on the collection of information about the displacement of the ball, before beginning to make a prediction on the trajectory followed by the object.

Virtual Reality (VR) is a computer-generated immersive environment with which users have real-time interactions that may involve visual feedback, 3D sound, haptic feedback, and even smell and taste (Burdea, 1996 ; Richard, 1999 ; Bohm, 1992 ; Chapin, 1992 ; Burdea, 1993 ; Sundgren, 1992 ; Papin, 2003). By providing both multi-modal interaction techniques and multi-sensorial immersion, VR presents an exciting tool for simulation of (real) human – (virtual) robot interaction or cooperation. However, this requires a large-scale Virtual Environments (VEs) that provide efficient and multi-modal interaction techniques including multi-sensorial feedbacks.

2 UMAN-SCALE VE

Our multi-modal VE is based on the SPIDAR interface (Figure 1). In this system, a total of 8 motors for both hands are placed as surrounding the user (Sato, 2001). Motors set up near the screen and behind the user; drive the strings (strings between hands and motors) attachments. One end of string attachment is wrapped around a pulley driven by a DC motor and the other is connected to the user's hand.

By controlling the tension and length of each string attachment, the SPIDAR-H generates an appropriate force using four string attachments connected to a hand attachment. Because it is a string-based system, it has a transparent property so that the user can easily see the virtual world.

It also provides a space where the user can freely move around. The string attachments are soft, so there is no risk of the user hurting himself if he

would get entangled in the strings. This human-scale haptic device allows the user to manipulate virtual objects and to naturally convey object physical properties to the user's body. Stereoscopic images are displayed on a retro-projected large screen (2m x 2,5m) and viewed using polarized glasses. A 5.1 immersive sound system is used for simulation realism, auditory feedback and sensorial immersion. Olfactory information can be provided using a battery of olfactory displays.

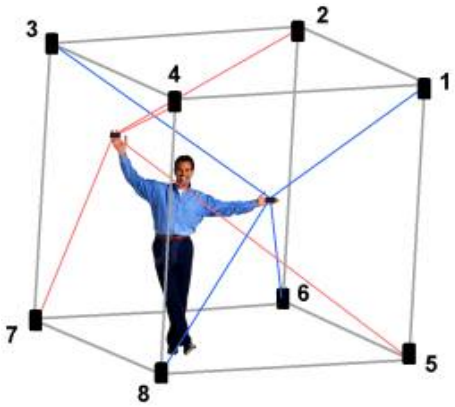


Figure 1: Workspace of the SPIDAR device.

3 CATCHING SIMULATION

3.1 Virtual Room

The virtual room in which simulation takes place is a right-angled parallelepiped which consists of a ground, a left wall and a right wall. The ceiling is left open. A wood texture was added on each face to increase the depth-of-field perception, as well as the ball shadow.

This virtual room contains objects such as a virtual ball, virtual hands (right and left), and a virtual robot (a Kuka KR6).

All calculation are made in cartesian co-ordinates X, Y, Z, according to an orthonormed reference frame whose origin O is located at the middle of the floor. The Z axis is directed towards the user. The Y axis is directed upwards. The X axis is directed towards the right compared to the user view.

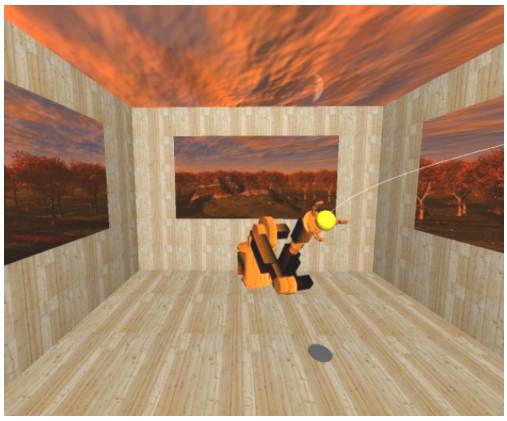


Figure 2: Snapshot of the robot reaching for the ball.

3.2 Robot Modelling

The robot closed here is Kuka KR6 model. It is an arm manipulator with 6 degrees of freedom, having only rotoids axes. It is placed at the bottom of the virtual room. Each part of the model was modelled in Discreet 3D Studio Max 7.0 and then imported into OpenGL. The robot consists of 6 rotoïds axes whose angles are respectively q1, q2, q3, q4, q5, q6.

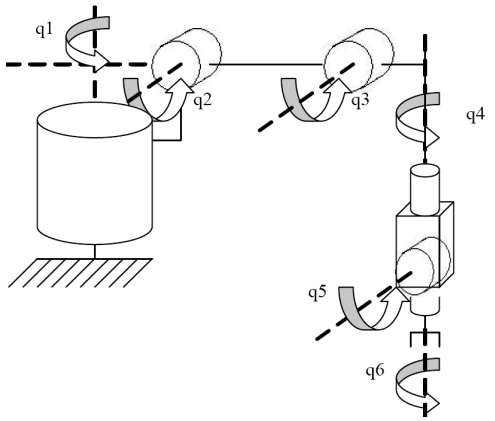


Figure 3: Illustration of the parameters used for the geometrical modelling of the Kuka KR6 robot.

To be able to animate each robot part, elementary geometrical operations such as translations and rotations around the frame reference will be used.

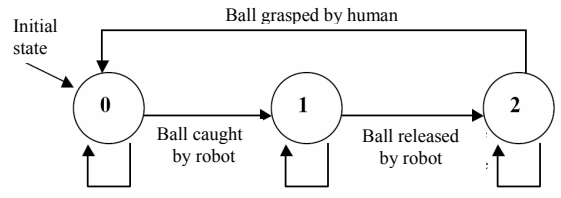


Figure 4: Finite state machine of the robot.

The virtual robot is subjected to the finite state machine given in figure 4. The various states are defined as follows:

State 0: the robot tries to catch the ball if in its workspace.

At the beginning of simulation, the robot waits until the ball is seized by the human operator, via the virtual hand, or till an external force is emitted on the ball, to return to state 0.

State 1: the robot catches the ball if in state 0.

State 2: the robot releases the ball automatically, after a certain amount of time, and returns in its initial configuration.

The robot waits until the ball is grasped by the user (using one of the virtual hand) or till an external force is emitted on the ball to return to state 0. Once the ball is caught, the robot automatically drops the ball and the simulation is reinitialised in its initial configuration.

The virtual ball is represented by a sphere and has a given mass "m", a given radius "R" and a given velocity "Vb" (or rather a Velocity Vector).

Assimilated to a single point which is the centre of the sphere, the ball is animated according to the fundamental law of dynamics: F=mA, i.e. the sum of the external forces F applied to the ball, is equal to the mass of the ball multiplied by acceleration.

Thus, the animation engine of the ball uses the following formulas:

Force = truncate(Force, max_force)
Acceleration = Force/m
Velocity = Velocity + Acceleration
Velocity = truncate(Velocity, max_velocity)
Position = Position + Velocity
Or Force=(Fx,Fy,Fz) , Acceleration=(Ax,Ay,Az) ,
Velocity = (Vx , Vy , Vz) ,
Position (Px , Py , Pz)

"max_force" is defined by the developer. It represents the maximum force that could be applied to the ball. Similarly, "max_velocity" represents the maximum velocity that could be set to the ball. Thus one truncates the force by "max_force" and velocity by "max_velocity" to avoid reaching a force or velocity of oversized magnitudes.

In this way, a new position of the ball could be calculated at any moment (more precisely according to the main loop of the simulation), when the ball is free (not caught by the robot or grasped by the user).

The ball is subjected to the finite state machine given in fig.5.

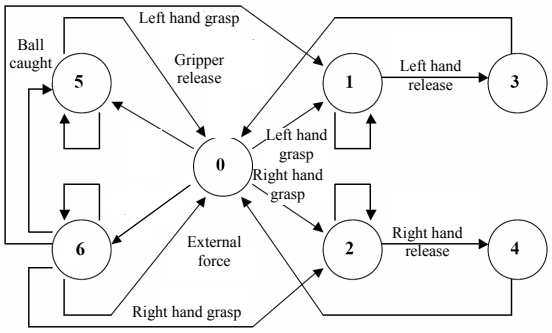


Figure 5: Finite state machine of the ball.

The various states are defined as follows:

State 0: the ball is free and is thus subjected to the animation engine described before.

State 1: the ball is caught by the left hand. The position of the ball is therefore directly linked to the position of this hand.

State 2: the ball is caught by the right hand. The position of the ball is therefore directly linked to the position of this hand.

State 3: the ball is released by the left hand. The position of the ball is no more established by the hand, but rather by the animation engine. The external Forces vector is equal, at this moment, to the hand velocity vector Vmx, Vmy, Vmz.

State 4: the ball is released by the right hand. The position of the ball is no more established by the hand, but rather by the animation engine. The external Forces vector is equal, at this moment, to the hand velocity vector Vmx, Vmy, Vmz.

State 5: the ball is caught by the robot. The position of the ball is no more established by the animation engine, but rather is a function of the robot gripper position.

State 6: the ball lies on the ground or closed to the ground. The Velocity vector magnitude is close to zero. The ball automatically moves to state 6, which is the end state and is immobilized on the ground.

User's hands position is tracked using the SPIDAR device.
A gain parameter between the user hand movements and the virtual hands can be introduced in order to enable him to increase his workspace. For example, it can be tuned so that the user can reach any location of the virtual room without moving too far from the centre of the SPIDAR frame of reference.

The closing of the virtual hand is carried out by the closing of a 5dt wireless data glove worn by the user (http://www.5dt.com). This could also be achieved using wireless mousses integrated to the SPIDAR device.

Each virtual hand is subjected to the finite state machine given in fig.6. The different states are defined as follows:

State 0: the left (respectively right) hand is open: it cannot grasp the ball.
State 1: the left (respectively right) hand is closed: it can grasp the ball if the latter is in state 0 or 6 or 1 (respectively 2).

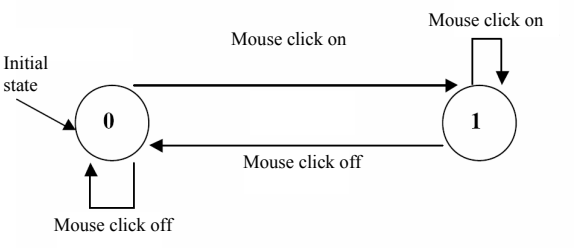


Figure 6: Finite state machine for both hands.

To do the ball grasping, a sphere of detection is used. Its size is defined by the designer and it is invisible during simulation. If the ball and the sphere are in contact, it is considered that the ball is seized, and the position of the ball is readjusted according to the hand.

3.3 Ball Launching

The virtual ball is thrown by the human operator, which can grasp and move it using the virtual hands. Once the ball is grasped, a method to launch the ball, corresponding to the animation engine, is proposed and validated. This method allows efficient velocity transfer of a user hand to the virtual ball.

To do this, hand velocity must be calculated. Thus an array of size S (S being defined by the designer), is created and is used to record the hand position at each loop cycle of the main program loop.

Fig. 7 illustrates an example with an array of size S = 4. At the initialisation, the array is empty.

This method is easy to implement is and is not CPU-time consuming. It gives good results to reproduce realistic "launched balls". However, this requires an optimisation of the size (S) of the array. One can also divide this subtraction by a time "T", function of times to which were recorded the last entered position, and the position in the past entered, with an aim obviously of standardizing speed compared to reality.

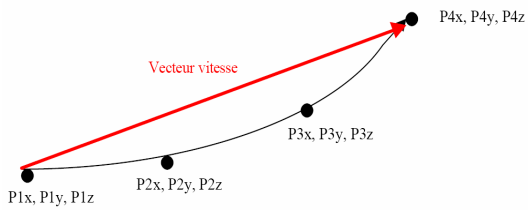


Figure 7: Illustration of the method proposed to efficiently launch the virtual ball with an array size S=4.

3.4 Ball Catching

Ball catching is achieved using a detection sphere of predefined size and invisible during the simulation. If the ball and the sphere are in contact, it is considered that the ball is caught. Then the ball position is readjusted according to the robot gripper position.

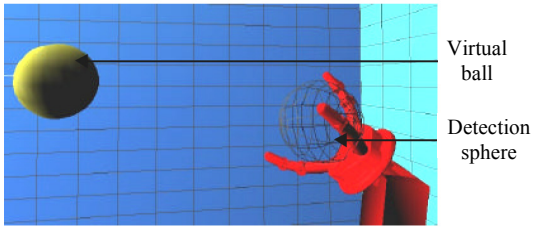


Figure 8: Illustration of the algorithm used for ball catching by the robot gripper.

This requires knowing both the cartesian position of the gripper according to the 6 angles q1, q2, q3, q4, q5, q6 and the dimensions of each part of the robot. The gripper is subjected to the finite state machine illustrated on fig. 9.

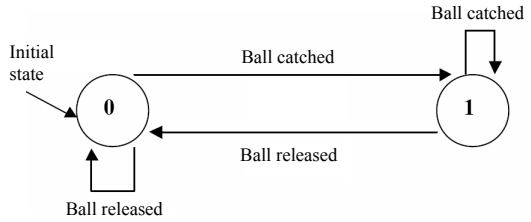


Figure 9: Finite state machine for the gripper.

The states of the gripper are defined as follows:

State 0: the gripper is open; the grip is open when the ball is not caught.

State 1: the gripper is closed; the grip is closed when the ball is caught.

In order for the robot to catch the ball, it is necessary to know: (1) the cartesian position of the gripper at any moment according to the 6 angles q1, q2, q3, q4, q5, and q6 of the robot and, (2) the Cartesian space which the robot can reach (workspace). This is given by the direct geometrical model defined by X=f(Q), with X=(x,y,z) and Q=(q1,q2,q3,q4,q5,q6).

It is also necessary to know the value of the 6 angles of the robot, according to the Cartesian position of the gripper (X, Y, Z). The inverse geometrical model can obtain these.

It is thus a question of determining the articular coordinates Q making it possible to obtain a desired location for the gripper specified by the operational coordinates.

Here, we are confronted with a system of 3 equations with 6 unknown variables. To solve this system, the method proposed by Paul (1981) was used. This method allows obtaining the whole solutions set, when they exist.

In our simulation, the robot always faces the ball.
However, it will carry out a catching movement towards the ball only if the latter is in its workspace, defined by the whole set of points in the Cartesian space that the robot gripper can reach.

Under the hypotheses that the robot can reach all the points of its workspace at any time, and that there is no constraint on the rotation angles of the joint, the workspace of the robot is a TORE defined by equation 3.

$$\left(\sqrt{(x^2 + z^2)} - A \right)^2 + y^2 = R^2 \qquad (3)$$

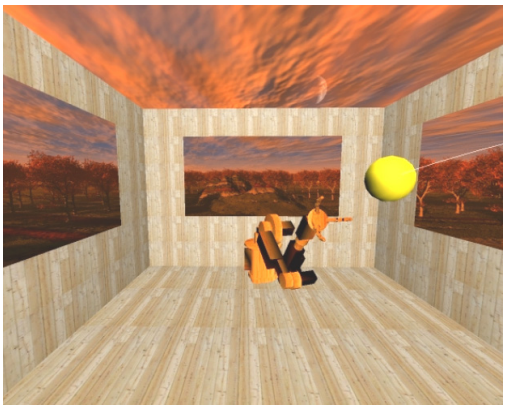


Figure 10: Snapshot of the robot oriented towards the ball.

Figure 11: Snapshot of the robot realising the ball.

4 CONCLUSION

We present a human-scale virtual reality catching robot simulation.. The user interacts with a virtual robot by throwing virtual balls towards it, using a large-scale bimanual haptic interface. The interface is used to track user's hands movements and to display various aspects of force feedback associated mainly with contact, weight, and inertia. We presented the robot modelling, as well as the ball launching and catching procedures.

REFERENCES

Inglese, F.-X., Lucidarme, Ph., Richard, P., Ferrier, J.-L., 2005. PREVISE : A Human-Scale Virtual Environment with Haptic Feedback. In *Proceedings of ICINCO 2005*. Barcelona, Spain, pp. 140-145.

Burdea, G. , Coiffet, Ph.., Richard, P., 1996. Integration of multi-modal I/Os for Virtual Environments. In *International Journal of Human-Computer Interaction (IJHCI), Special Issue on Human-Virtual Environment Interaction*. March, (1), pp. 5-24.

Richard, P., Coiffet, Ph., 1999. Dextrous haptic interaction in Virtual Environments: human performance evaluation. In *Proceedings of the 8th IEEE International Workshop on Robot and Human Interaction*. October 27-29, Pisa, Italy, pp. 315-320.

Bohm, K., Hubner, K., Vaanaen, W., 1992. GIVEN: Gesture driven Interactions in Virtual Environments. A Toolkit Approach to 3D Interactions. In *Proceedings of Interfaces to Real and Virtual Worlds*. Montpellier, France, March, pp. 243-254.

Chapin, W., Foster, S., 1992. Virtual Environment Display for a 3D Audio Room Simulation. *In Proceedings of SPIE Stereoscopic Display and Applications*. Vol.12.

Burdea, G., Gomez, D., Langrana, N., 1993. Distributed Virtual Force Feedback. In *Proceedings of IEEE Workshop on Force Display in Virtual Environments and its Application to Robotic Teleoperation*. Atlanta, May 2.

Sundgren, H., Winquist, F., Lundstrom, I., 1992. Artificial Olfactory System Based on Field Effect Devices. In *Proceedings of Interfaces to Real and Virtual World.* Montpellier, France, pp. 463-472, March.

Papin, J.-P., Bouallagui, M., Ouali, A., Richard, P., Tijou, A., Poisson, P., Bartoli, W., 2003. DIODE: Smell-diffusion in real and virtual environments. In *Proceedings of the 5th International Conference on Virtual Reality.* Laval, France, pp.113-117, May 14-17.

Bowman, D.A., Kruijff, E., LaViola, J.J., Poupyrev, I., 2004. *3D User Interfaces: Theory and Practice.* Addison Wesley / Pearson Education.

Richard, P., Birebent, G., Burdea, G., Gomez, D., Langrana, N., Coiffet, Ph., 1996. Effect of frame rate and force feedback on virtual objects manipulation. In *Presence - Teleoperators and Virtual Environments*. MIT Press, 15, pp. 95-108.

Bouguila, L., Ishii, M., Sato, M. , 2000. A Large Workspace Haptic Device For Human-Scale Virtual Environments. In *Proceedings of the Workshop on Haptic Human-Computer Interaction*. Scotland.

Sato, M., 2001. Evolution of SPIDAR, In Proceedings of the 3rd International Virtual Reality Conference. Laval, May, France.

Richard, P., 1981. *Robot Manipulators--Mathematics, Programming, and Control*, MIT Press.

A LOCAL LEARNING APPROACH TO REAL-TIME PARAMETER ESTIMATION
Application to an Aircraft

Lilian Ronceray, Matthieu Jeanneau
Stability and Control Department, Airbus France, Toulouse, France
Lilian.L.Ronceray@airbus.com

Daniel Alazard
Supaéro, Toulouse, France

Philippe Mouyon
Département Commande des Systèmes et Dynamique du Vol, ONERA, Toulouse, France

Sihem Tebbani
Département Automatique, École Supérieure d'Électricité, Gif-sur-Yvette, France

Keywords: Local learning, radial-basis neural networks, real-time parameter estimation.

Abstract: This paper proposes an approach based upon local learning techniques and real-time parameter estimation, to tune an aircraft sideslip estimator using radial-basis neural networks, during a flight test. After a presentation of the context, we recall the local model approach to radial-basis networks. The application to the estimation of the sideslip angle of an aircraft, is then described and the various results and analyses are detailled at the end before suggesting some improvement directions.

1 INTRODUCTION

In the aeronautical field, although most of the useful parameters (like inertial data, airspeed) are calculated directly using probes, it is often relevant to use estimators to consolidate the information, thus increasing the redundancy of the aircraft systems or to replace the probes in order to save weight. It then becomes critical to have these estimators tuned in the early days of the flight tests of a new aircraft, when it is only known by an inaccurate numerical model.

The problem that is dealt with in this paper, is that of retuning a specific estimator in real-time (or near real-time) using flight test data. We must then take into account that during flight tests, the aircraft flies around in small regions of the flight domain, yielding a strong locality constraint on the retuning.

The general idea will be to make a combined use of local learning and real-time parameter estimation techniques to tune the estimator, in a small neighbouring of its input space, without degrading its performance in other regions.

2 ABOUT NEURAL NETWORKS

In this section, some generalities about RBF networks are recalled.

2.1 General Description

Such a network is composed by a set of N local estimators $\{\widehat{f}_i(\mathbf{x})\}_{i=1}^N$, defined in the neighbouring of some points $\{\mathbf{c}_i\}_{i=1}^N$ in an input space $\mathcal{I}$ (Murray-Smith, 1994).

The resulting global estimation $\widehat{\xi}$ is the weighted sum of the outputs of all local estimators, for a query point $\mathbf{x} \in \mathcal{I}$ (see equation 1).

The weighting φ_i of the local estimation $\widehat{f}_i(\mathbf{x})$ is a function of $\frac{\|\mathbf{x}-\mathbf{c}_i\|}{\sigma_i}$ for all $i = 1 \dots N$. φ_i is then considered as a radial-basis function. The resulting output is then:

$$\widehat{\xi} = \sum_{i=1}^{N} \varphi_i \widehat{f}_i(\mathbf{x}) \tag{1}$$

2.2 Local Parameters Computation

The local models $\widehat{f}_i(\mathbf{x})$ can either be constants in the neighbouring of the centers, or affine models in the inputs, or any nonlinear model.

From now on, we will show how the parameters of the local models are computed in the particular case where these models are linear in the inputs (Haykin, 1999):

$$\widehat{f}_i(\mathbf{x}) = \mathbf{x}^T \theta_i \quad \forall \mathbf{x} \in I$$

The problem to solve is set as follows: given a set of K different points $\{\mathbf{x}_k \in \mathbb{R}^{m_0}\}_{k=1}^K$ and a corresponding set of K reference values to estimate $\{\xi_k \in \mathbb{R}\}_{k=1}^K$, find a function $\widehat{\xi}$ such that :

$$\widehat{\xi}(\mathbf{x}_k) = \xi_k, \quad \forall k = 1 \ldots K \tag{2}$$

The RBF technique consists in choosing a function $\widehat{\xi}$ that has the following form :

$$\widehat{\xi}(\mathbf{x}) = \sum_{i=1}^{N} \varphi_i \mathbf{x}^T \widehat{\theta}_i = \sum_{i=1}^{N} \psi_i(\mathbf{x})^T \widehat{\theta}_i \tag{3}$$

$$= \psi(\mathbf{x})^T \widehat{\theta} \tag{4}$$

where N is the chosen number of local estimators, and with $\psi_i = \varphi_i \mathbf{x}$.

With $\Psi = \{\Psi_{ki}\} = \{\psi_i(\mathbf{x}_k)\}$, the interpolation condition (2) can be written as a linear system :

$$\Psi^T \widehat{\theta} = \xi \tag{5}$$

In order to find a solution to equation (5) and as Ψ is not a square matrix, we must verify that $\Psi\Psi^T$ is nonsingular, which can be done using Michelli's theorem (Michelli, 1986). A solution $\widehat{\theta}$ satisfying the interpolation condition (2), can then be found using least square optimization :

$$\widehat{\theta}^* = \left(\Psi\Psi^T\right)^{-1} \Psi\xi \tag{6}$$

2.3 Selection of the Centers

A simple solution is to make a regular gridding on the normalized input space. As minimum and maximum variations of the input parameters are known, the input space can be normalized. The gridding is then made on a unitary hypercube.

The issue is that we face the curse of dimensionality though the dimension of each network's input space is smaller than 5. However, some physical considerations, depending on the considered application, may help reducing the number of neurons by making a "truncated" hypercube.

For instance, in our application, some parameters have a dependency in Mach number and angle of attack α. As the aircraft is not designed to fly at both high Mach and high α, we may remove the corresponding part of the domain.

3 APPLICATION

The application we considered here is the estimation of the sideslip angle β of a civilian aircraft, which is the angle between the aircraft longitudinal axis and the direction of flight (Russell, 1996).

To do so, we have a formula for the estimation of β, based on equation (7) that describes the aircraft lateral force equation where we neglect the longitudinal coupling terms and equation (8) which is a classical decomposition of the lateral force coefficient (Boiffier, 1998):

$$mg \cdot Ny_{cg} = P_d S Cy - F_{eng,y} \tag{7}$$

$$Cy = Cy_\beta \beta + \Delta Cy_\beta^{NL} + \frac{l}{V_{tas}}\left(Cy_r r + Cy_p p\right) + \Delta Cy_{\delta r} + \Delta Cy_{\delta p} \tag{8}$$

where Ny_{cg} denotes the lateral load factor, P_d the dynamic pressure, $F_{eng,y}$ the projection of thrust on the lateral axis, β the sideslip angle, p the rolle rate, r the yaw rate, δp the ailerons deflection, δr the rudder deflection, $Cy_\star$ the Cy gradient w.r.t. $\star$ (β, p or r), $\Delta Cy_\star$ the Cy effect due to $\star$ (δp, δr or β), l the mean aerodynamic chord and V_{tas} the true airspeed velocity.

An approximation of the aircraft sideslip can then be deduced :

$$\widehat{\beta} = -\left[\frac{1}{Cy_\beta}\right]\frac{Mg}{P_d S} Ny_{cg} - \left[\frac{\Delta Cy_{\delta p}}{Cy_\beta}\right] - \left[\frac{\Delta Cy_{\delta r}}{Cy_\beta}\right] - \delta_{HL}\left[\frac{\Delta Cy_\beta^{NL}}{Cy_\beta}\right] - \frac{l}{V_{tas}}\left(\left[\frac{Cy_p}{Cy_\beta}\right] p + \left[\frac{Cy_r}{Cy_\beta}\right] r\right)$$

The key points treated in the sequel are the definition of the architecture and the initialisation of the neural networks from a given set of simulated data, and the in-flight tuning of these networks.

3.1 Rbf Networks Applied to Sideslip Estimation

We will start by noticing that the expression (9) is linear in ratios of aerodynamic coefficients.

In order to ease the reader's effort, the following notations are introduced. Let M denote the number of unknown ratios of aerodynamic coefficients, $\widehat{\xi}^m$ the

m-th unknown ratio with $m = 1 \dots M$ and y^m its attached auxiliary measurement. (9) will then be written as:

$$\widehat{\beta} = \sum_{m=1}^{M} y^m \cdot \widehat{\xi}^m \tag{9}$$

The m-th unknown ratio is modelled by a RBF network with N^m linear local estimators $\widehat{f}_{i,m}(\mathbf{x})$. Its input space $\mathcal{I}^m$ is a subset of the flight enveloppe variables.

The $\widehat{\xi}^m$ depends on the following variables : α, Mach number, P_d, δr and δp, the last two being only used respectively for $\Delta Cy_{\delta r}$ and $\Delta Cy_{\delta p}$.

According to equation (3), estimated sideslip can then be rewritten as :

$$\begin{aligned} \widehat{\beta} &= \sum_{m=1}^{M} y^m \psi^{mT} \widehat{\theta}^m = \sum_{m=1}^{M} \zeta^{mT} \widehat{\theta}^m \\ &= \zeta^T \Theta \end{aligned} \tag{10}$$

A linear expression of the estimated sideslip is thus obtained, allowing the recursive least algorithm (RLS) algorithm to be directly applied. For a complete formulation of the RLS algorithm and the related criterion, one may refer to (Labarrère et al., 1993).

3.2 The Process in Details

The process will be divided into three main parts.

Initialization: the optimal network structure must be found : RBF, distance function, feature scaling (transformation on the input space), centers location, and smoothing parameter σ_i (Atkeson et al., 1997).

A direct offline learning: where each network is trained individually on a database of aerodynamic coefficient values, computed by the numerical model. A particular attention will be paid to the norm of the local parameter vector, which is an indicator of the generalization performance of the network.

An indirect online learning: where the learning criterion is no longer the error of the networks outputs but the error between the estimated sideslip and the true sideslip. All the networks are trained at the same time and must achieve both local performance, i.e. on the considered flight point and global performance, i.e. on the whole flight domain.

3.3 Analysis

The structure of the neural networks is a key element in the outcome of the process and requires an analysis of the networks' offline performance, with respect to the various degrees of freedom available for the networks' structure.

For clarity reasons, we will only show the performance of the ($\Delta Cy_{\delta r}$) network in the sequel.

Prior to the analysis, the input space of the network is normalized.The effect of the smoothing parameter σ on the learning performance and the Euclidean norm of the parameter vector, which gives an idea of the network's capacity to generalize, will be studied. Both the Euclidean and Infinity norm will be used as distance functions and inverse multiquadrics as RBF ($f : x \mapsto \frac{1}{\sqrt{x^2+1}}$). We perform an offline learning for each value of σ and compute the relative error on the whole training data :

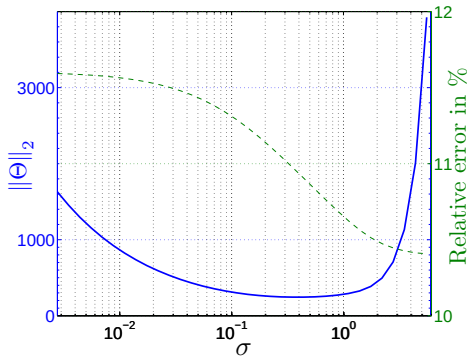


Figure 1: Influence of σ using $\|.\|_2$.

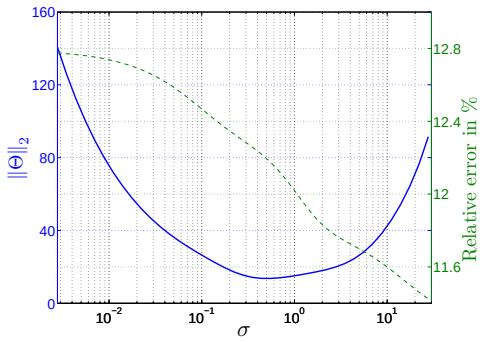


Figure 2: Influence of σ using $\|.\|_\infty$.

The plain line represents the norm of the parameter vector and the dashed line the relative error.

For both norms, we can see a behaviour that can be compared with the overfitting phenomenon with multilayer perceptrons (Haykin, 1999; Dreyfus et al., 2004). We could name this "over-covering", as it seems that local models are strongly interfering with each other, hence over-compensating their interaction. σ must then be chosen that reaches a compro-

mise between a decent performance and a relatively small norm for the parameter vector.

In terms of compared performance, the Infinity norm allows better generalization, as the norm reaches lower values for a quite similar performance. This can be justified by figure 3: the covering of 2D normalized space is presented using gaussian kernels with Euclidean (solid line circles) and Infinity norm (dotted lines squares).

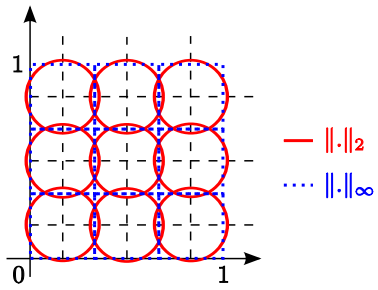


Figure 3: Input space covering using $\|.\|_2$ and $\|.\|_\infty$.

3.4 Implementation

We then chose the structure of our networks : linear local models, inverse multiquadrics as RBF[1], optimal smoothing parameter according to the previous analysis, normalisation of the input space as feature scaling and a regular gridding to locate the centers.

To test our method, we used flight tests recordings to be as close as possible to real conditions.

3.5 Results

The obtained results are presented in this section, from learning on the pre-flight test identification data to the online tuning during simulated flight tests (in-flight recorded data is fed through the estimator).

About the offline part, as it is basic least squares optimization, we will only say that the points were generated randomly using an inaccurate numerical model of the aircraft.

For the online adaptation, we will present results in clean configuration for two distinct flight points (FP1 and FP2) on a steady sideslip maneuver.

First, estimations without and then with the RLS algorithm are presented. The dot-dashed line represents the real sideslip at the center of gravity, the dashed line the estimated sideslip computed by the current method and the solid line the estimated sideslip computed by our estimator (figures 4 and 5).

The aim is to compare the performance of the existing sideslip estimator which uses interpolated approximate values for the aerodynamic coefficients.

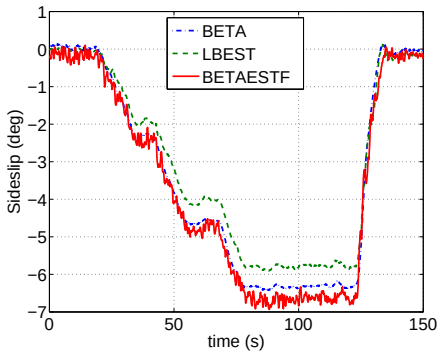


Figure 4: Flight point 1 - Fixed estimator.

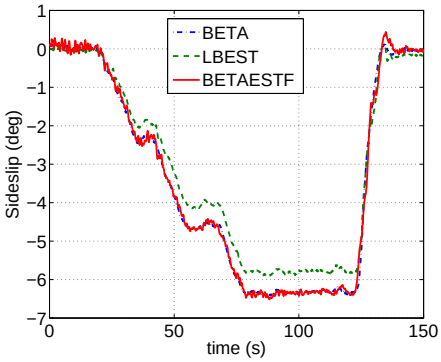


Figure 5: Flight point 1 - RLS estimator.

The results are quite satisfactory because better estimation than the existing estimator is achieved. The noise we can see on both figures comes from the *Ny* sensor and the derivation *p* and *r*.

The issue is the generalization and is emphasized by the following procedure. The estimator is tuned on FP1 then on FP2. Local performance is achieved for both flight points as shown in figure 5 for FP1. The estimator is then verified on FP1. We can see on figure 6 that the original performance has been degraded. The learning algorithm does not tune locally enough and impacts the whole flight domain.

[1]They are better suited for an implementation on an embedded computer

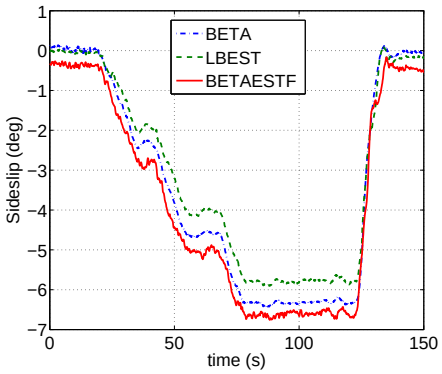


Figure 6: Generalization from FP2 to FP1.

4 CONCLUSION

Throughout this paper, we studied the application of RBF-based neural networks on the estimation of an aircraft sideslip and tried to find a method to tune it in real-time during a simulated flight test.

Though such networks have interesting local properties, some improvements are required on the different steps of the process, mainly on the generalization performance. Work is currently on-going about using total least squares algorithm instead of the classical least squares (Huffel and Vandewalle, 1991; Björck, 1996) and their recursive form (Boley and Sutherland, 1993). Other work directions will be investigated :

- allowing directional forgetting in the RLS algorithm (Kulhavy and Kárny, 1984)
- reducing numerical complexity
- adaptive filtering on the estimator output to soften input noise effects

REFERENCES

Atkeson, C., Moore, A., and Schaal, S. (1997). Locally weighted learning. *AI Review*, 11:11–73.

Björck, A. (1996). *Numerical Methods for Least Squares Problems*. S.I.A.M., first edition.

Boiffier, J.-L. (1998). *The Dynamics of Flight, The Equations*. Wiley.

Boley, D. L. and Sutherland, K. T. (1993). Recursive total least squares: An alternative to the discrete kalman filter. Technical Report TR 93-32, Computer Science Dpt, University of Minnesota.

Dreyfus, G., Martinez, J.-M., Samuelides, M., Gordon, M., Badran, F., Thiria, S., and Hérault, L. (2004). *Réseaux de neurones, méthodologies et applications*. Eyrolles, second edition.

Haykin, S. (1999). *Neural Networks, a comprehensive foundation*. Prentice-Hall, second edition.

Huffel, S. V. and Vandewalle, J. (1991). *The Total Least Squares Problem : Computational Aspects and Analysis*. S.I.A.M., first edition.

Kulhavy, R. and Kárny, M. (1984). Tracking of slowly varying parameters by directional forgetting. *Preprints of the 9th IFAC World Congress*, X:78–83.

Labarrère, M., Krief, J.-P., and Gimonet, B. (1993). *Le Filtrage et ses Applications*. Cépaduès.

Michelli, C. A. (1986). Interpolation of scattered data : Distance matrices and conditionally positive definite functions. *Constructive Approximation*, 2:11–22.

Murray-Smith, R. (1994). Local model networks and local learning. In *Fuzzy-Duisburg*, Duisburg.

Russell, J. B. (1996). *Performance and Stability of Aircraft*. Arnold.

SPECIAL SESSION ON FRACTIONAL ORDER SYSTEMS

CHAIR:
SAMIR LADACI

SOLUTION OF THE FUNDAMENTAL LINEAR FRACTIONAL ORDER DIFFERENTIAL EQUATION

A. Charef, M. Assabaa and Z. Santouh
Université Mentouri de Constantine
Département d'Electronique
Route Ain El-bey - Constantine 25000 - Algeria
E-Mail : afcharef@yahoo.com

Keywords: Fractional order differential equations, Fractional power zero, Irrational transfer function, Rational function.

Abstract: This paper provides a solution of the fractional order system represented by the fundamental linear fractional order differential equation, namely, $(\tau_0)^m \frac{d^m x(t)}{dt^m} + x(t) = e(t)$ whose transfer function is given by $G(s) = \frac{X(s)}{E(s)} = \frac{1}{[1+(\tau_0 s)^m]}$ for $0 < m < 2$. Simple methods of approximation, for a given frequency band, of the transfer function of this fractional order system by a rational function are presented. Analytical impulse and step responses of this system are derived. Illustrative examples are presented to show the exactitude of the approximation methods.

1 INTRODUCTION

In the recent decades the concepts of fractional order derivatives and integrals has been arisen in various areas of the engineering fields (Torvik,1984), (Ichise, 1971), (Sun, 1983), (Cole, 1941), (Davidson, 1950). Theses fractional concepts have been generally used to model physical systems, leading to the formulation of the linear fractional order differential equations. So, the dynamic systems described by this type of fractional differential equation are called fractional linear systems. With the growing number of applications system and control fields (Manabe, 1961), (Oustaloup, 1983), (Charef, 1992), (Podlubny, 1994), (Miller, 1993), (Hartley, 1998), (Petras, 2002), it is important to establish a clear system theory for these fractional order systems, so they may be accessible to the general engineering community.

The fundamental linear fractional order differential equation, defined in (Petras et al., 2002), is represented by the following equation:

$$(\tau_0)^m \frac{d^m x(t)}{dt^m} + x(t) = e(t), \quad \text{for } 0 < m < 2 \qquad (1)$$

The transfer function of this type of fractional order systems is given by the following irrational function:

$$G(s) = \frac{X(s)}{E(s)} = \frac{1}{[1+(\tau_0 s)^m]}, \quad \text{for} \quad 0 < m < 2 \qquad (2)$$

In this paper an effective and easy to use methods are presented for the approximation by a rational function, for a given frequency band, of the transfer function of the fundamental linear fractional order differential equation. Analytical impulse and step responses of this system are also derived. Illustrative examples are presented to show the exactitude and the usefulness of the approximation methods.

2 RELAXATION FRACTIONAL ORDER SYSTEM

2.1 Definition

Relaxation fractional order system is defined in this context as the fundamental linear fractional order differential equation of equation (1) with the transfer function of equation (2) for $0 < m < 1$.

2.2 Rational Function Approximation

In dielectric studies, Cole and Cole (Cole, 1941) observed that dispersion/relaxation data measured

from a large number of materials can be modeled by the following function:

$$G(s) = \frac{1}{[1+(\tau_0 s)^m]}, \quad \text{for} \quad 0 < m < 1 \tag{3}$$

It is also known that the distribution of relaxation times function $H(\tau)$ can be derived directly from the original transfer function as (MacDonald, 1987):

$$G(s) = \int_0^\infty \frac{H(\tau)}{1+s\tau} d\tau \tag{4}$$

Cole and Cole (Cole, 1941) applied the above method to find the distribution of relaxation times function $H(\tau)$ for their model of equation (3) to be :

$$G(s) = \frac{1}{[1+(\tau_0 s)^m]} = \int_0^\infty \frac{H(\tau)}{1+s\tau} d\tau, \text{ for } \quad 0 < m < 1 \tag{5}$$

with

$$H(\tau) = \frac{1}{2\pi}\left[\frac{\sin[(1-m)\pi]}{\cosh[m\log(\frac{\tau}{\tau_0})] - \cos[(1-m)\pi]}\right] \tag{6}$$

The method of approximation began by sampling the distribution of relaxation times function $H(\tau)$ of equation (6) for a limited frequency band of approximation of practical interest $[0, \omega_H]$ at logarithmically equidistant points τ_i as follows (Sun, 1992):

$$H(\tau) \cong H_s(\tau) = \sum_{i=1}^{2N-1} H(\tau_i)\delta(\tau - \tau_i) \tag{7}$$

and the points τ_i are such that:

$$\tau_i = \tau_0(\lambda)^{N-i} \quad \text{for i = 1,2, . . . , 2N-1} \tag{8}$$

with τ_N occurring at the characteristic relaxation time τ_0, and λ, a constant positive real number greater than unity, is chosen such that:

$$\lambda = \frac{\tau_i}{\tau_{i+1}} \quad \text{for} \quad i = 1,2, \ldots, 2N-1 \tag{9}$$

Substituting equation (7) into equation (5), we obtain:

$$G(s) \cong \int_0^\infty \frac{\sum_{i=1}^{2N-1} H(\tau_i)\delta(\tau - \tau_i)}{1+s\tau} d\tau = \sum_{i=1}^{2N-1} \frac{H(\tau_i)}{1+s\tau_i} \tag{10}$$

Hence, we can write that:

$$G(s) = \frac{1}{[1+(\tau_0 s)^m]} \cong \sum_{i=1}^{2N-1} \frac{k_i}{\left(1+\frac{s}{p_i}\right)} \tag{11}$$

where the p_i 's are the poles of the approximation which are given as:

$$p_i = \frac{1}{\tau_i} = (\lambda)^{(i-N)} p_0 \quad \text{, for} \quad i = 1,2,...,2N-1 \tag{12}$$

such that $p_0 = 1/\tau_0$ and $\lambda = p_{i+1}/p_i$, the k_i 's are the residues of the poles which are given from equation (6), for i = 1,2,...,2N-1, as:

$$k_i = \frac{1}{2\pi}\left[\frac{\sin[(1-m)\pi]}{\cosh[m\log(\frac{\tau_i}{\tau_0})] - \cos[(1-m)\pi]}\right] \tag{13}$$

and for an approximation frequency ω_{max} which can be chosen to be $1000\omega_H$, with $[0, \omega_H]$ is the frequency band of practical interest, the number N is determined as follows:

$$N = \text{Integer}\left[\frac{\log(\tau_0\omega_{max})}{\log(\lambda)}\right] + 1 \tag{14}$$

2.3 Time Responses

From equation (11), we have that:

$$G(s) = \frac{X(s)}{E(s)} = \frac{1}{[1+(\tau_0 s)^m]} \cong \sum_{i=1}^{2N-1} \frac{k_i}{\left(1+\frac{s}{p_i}\right)} \tag{15}$$

so,

$$X(s) = \frac{E(s)}{[1+(\tau_0 s)^m]} \cong \sum_{i=1}^{2N-1} \frac{k_i}{\left(1+\frac{s}{p_i}\right)} E(s) \tag{16}$$

for $e(t) = \delta(t)$ the unit impulse $E(s) = 1$, we will have

$$X(s) = \sum_{i=1}^{2N-1} \frac{k_i}{\left(1 + \frac{s}{p_i}\right)} \tag{17}$$

thus, the impulse response can be obtained as:

$$x(t) = \sum_{i=1}^{2N-1} k_i p_i \exp(-p_i t) \tag{18}$$

For e(t) = u(t) the unit step E(s) = 1/s, will be:

$$X(s) = \sum_{i=1}^{2N-1} \frac{k_i}{\left(1 + \frac{s}{p_i}\right)} \frac{1}{s} = \sum_{i=1}^{2N-1} k_i \left(\frac{1}{s} - \frac{1}{s + p_i}\right) \tag{19}$$

thus, the step response can be obtained as:

$$x(t) = \sum_{i=1}^{2N-1} k_i \left(1 - \exp(-p_i t)\right) \tag{20}$$

2.4 Illustrative Example

For illustration purpose let's take a numerical example for a relaxation fractional order system represented by the fundamental linear fractional order differential equation with m = 0.65 and $\tau_0 = 10$ as:

$$(10)^{0.65} \frac{d^{0.65} x(t)}{dt^{0.65}} + x(t) = e(t)$$

its transfer function is given by:

$$G(s) = \frac{1}{1 + (10s)^{0.65}}$$

For a frequency band $[0, \omega_H] = [0, 100$ rad/s], the approximation frequency $\omega_{max} = 1000\omega_H = 100000$ rad/s, $p_0 = 0.1$ rad/s and the ratio $\lambda = 4$, the number N, the poles p_i and the residues k_i of the approximation can be easily calculated from section (II.2) as: N=10, $p_i = (4)^{(i-N)} p_0$, for i = 1,2,...,19, and

$$k_i = \frac{1}{2\pi}\left[\frac{\sin[(1-m)\pi}{\cosh[m\log((4)^{(10-i)})] - \cos[(1-m)\pi]}\right]$$

Figures (1) and (2) show the Bode plots of the relaxation fractional order system transfer function and its proposed rational function approximation. We can easily see that they are all quite overlapping over the frequency band of interest. Figures (3) and (4) show respectively the impulse and the step responses of this fractional order system obtained from its proposed rational function approximation.

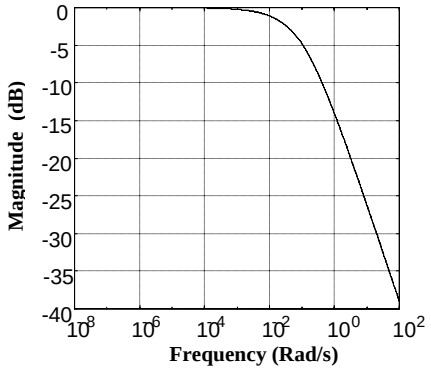


Figure 1: Magnitude of the Bode plot.

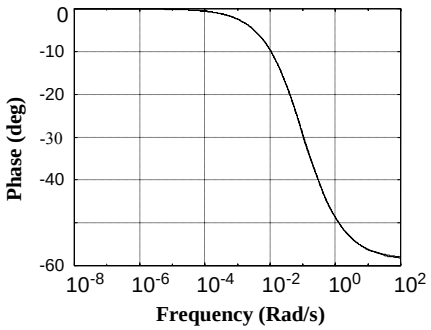


Figure 2: Phase of the Bode plot.

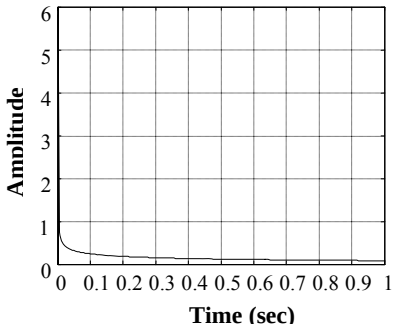


Figure 3: Impulse response.

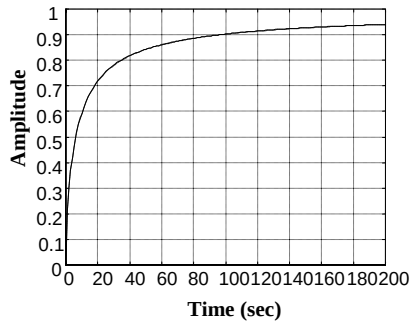


Figure 4: Step response.

3 OSCILLATION FRACTIONAL ORDER SYSTEM

3.1 Definition

Oscillation fractional order system is defined in this context as the fundamental linear fractional order differential equation of equation (1) with the transfer function of equation (2) for $1 < m < 2$.

3.2 Rational Function Approximation

First, the transfer function of the oscillation fractional order system is modeled as:

$$G(s)=\frac{1}{[1+(\tau_0 s)^m]}\cong\frac{(1+\tau_0 s)^{(2-m)}}{(\tau_0 s)^2+2\zeta(\tau_0 s)+1}=G_N(s)G_D(s) \quad (21)$$

$$G_N(s)=(1+\tau_0 s)^{(2-m)} \quad (22)$$

is a fractional power zero (FPZ) with $0 < (2-m) < 1$

$$G_D(s)=\frac{1}{(\tau_0 s)^2+2\zeta(\tau_0 s)+1} \quad (23)$$

is a regular second order system. It can be easily shown that:

for $\omega << 1/\tau_0$, $\quad |G(j\omega)|=1\cong 1$

for $\omega >> 1/\tau_0$, $\quad |G(j\omega)|=\frac{1}{(\omega\tau_0)^m}\cong\frac{(\omega\tau_0)^{(2-m)}}{(\omega\tau_0)^2}=\frac{1}{(\omega\tau_0)^m}$

for $\omega = 1/\tau_0$, $\quad |G(j\omega)|=\left|\frac{1}{(1+j^m)}\right|\cong\frac{|(1+j)^{(2-m)}|}{|j2\zeta|}$

$$|G(j\omega)|=\frac{1}{\sqrt{[(1+\cos(\frac{\pi}{2}m))^2+(\sin(\frac{\pi}{2}m))^2]}}\cong\frac{(\sqrt{2})^{2-m}}{2\zeta} \quad (24)$$

In order that the two sides of equation (24) were equal, the damping ratio ζ of the regular second order system must be given as:

$$\zeta=\sqrt{\frac{[1+\cos(\frac{\pi}{2}m)]}{2^{m-1}}} \quad (25)$$

To represent the oscillation fractional order system by a rational transfer function instead of the irrational function of equation (2), we have to approximate the FPZ of equation (22) by a rational one in a frequency band $[0, \omega_H]$. The method of approximation of the FPZ consists of approximating its 20(2-m) dB/dec slope on the Bode plot by a number of zig-zag lines with alternate slopes of 20 dB/dec and 0 dB/dec corresponding to alternate zeros and poles on the negative real axis of the s-plane such that $z_0 < p_0 < z_1 < p_1 < . \ . \ . < z_N < p_N$. Hence, we can write that:

$$G_N(s)=(1+\tau_0 s)^{(2-m)}\cong\frac{\prod_{i=0}^{N}\left(1+\frac{s}{z_i}\right)}{\prod_{i=0}^{N}\left(1+\frac{s}{p_i}\right)} \quad (26)$$

So, equation (21) can be rewritten as:

$$G(s)=\frac{1}{[1+(\tau_0 s)^m]}\cong\frac{\prod_{i=0}^{N}\left(1+\frac{s}{z_i}\right)}{\prod_{i=0}^{N}\left(1+\frac{s}{p_i}\right)}\left[\frac{1}{(\tau_0 s)^2+2\zeta(\tau_0 s)+1}\right] \quad (27)$$

As the same idea of the method used to approximate the fractional power pole (Charef, 1992), the approximation of the ZPF began with a specified approximation error y in dB and an approximation frequency band ω_{max} which can be $100\omega_H$, then the parameters a, b, z_0 , p_0 and N of the approximation can be easily determined as follows:

$$a=10^{\left\lfloor\frac{y}{10[1-(2-m)]}\right\rfloor},\ b=10^{\left\lfloor\frac{y}{10(2-m)}\right\rfloor},\ z_0=\frac{1}{\tau_0}10^{\left\lfloor\frac{y}{20(2-m)}\right\rfloor}$$

$$p_0=az_0,\ \text{and}\ N=\text{Integer}\left[\frac{\log\left(\frac{\omega_{max}}{z_0}\right)}{\log(ab)}\right]+1$$

Hence, the zeros z_i's and the poles p_i's of equation (27) can then be derived from the above parameters for i=0,1,…,N as: $z_i=z_0(ab)^i$ and $p_i=p_0(ab)^i$. Then, equation (27) can be rewritten as:

$$G(s)=\frac{1}{[1+(\tau_0 s)^m]}=\frac{\prod_{i=0}^{N}\left(1+\frac{s}{z_0(ab)^i}\right)}{\prod_{i=0}^{N}\left(1+\frac{s}{p_0(ab)^i}\right)}\left[\frac{1}{(\tau_0 s)^2+2\zeta(\tau_0 s)+1}\right] \quad (28)$$

3.3 Time Responses

By partial fraction expansion of the rational function of equation (28) it is possible to represent the transfer function of the oscillation fractional order system by a linear combination of elementary simple functions, that is:

$$G(s) = \sum_{i=0}^{N} \frac{k_i}{\left(1 + \frac{s}{p_0(ab)^i}\right)} + \frac{As + B}{(\tau_0 s)^2 + 2\zeta(\tau_0 s) + 1} \qquad (29)$$

where the k_i (i=0,1, …, N) are the residues of the poles which can be calculated as:

$$k_i = \frac{\prod_{j=0}^{N}\left[1 - a(ab)^{(i-j)}\right]}{\prod_{\substack{j=0 \\ i \neq j}}^{N}\left[1 - (ab)^{(i-j)}\right]} \left\{ \frac{1}{\left(\tau_0 p_0 (ab)^i\right)^2 - 2\zeta\left(\tau_0 p_0 (ab)^i\right) + 1} \right\} \qquad (30)$$

and the constants A and B can also be calculated as:

at s = 0, $G(0) = B + \sum_{i=0}^{N} k_i = 1$, then $B = 1 - \sum_{i=0}^{N} k_i$, also

$$\lim_{s \to \infty} sG(s) = 0 = \frac{A}{\tau_0^2} + \sum_{i=0}^{N} k_i p_0 (ab)^i \text{, then } A = -\tau_0^2 \sum_{i=0}^{N} k_i p_0 (ab)^i$$

We will then have that:

$$G(s) = \frac{X(s)}{E(s)} = \sum_{i=0}^{N} \frac{k_i}{\left(1 + \frac{s}{p_0(ab)^i}\right)} + \frac{As + B}{(\tau_0 s)^2 + 2\zeta(\tau_0 s) + 1} \qquad (31)$$

$$X(s) = \sum_{i=0}^{N} \frac{k_i}{\left(1 + \frac{s}{p_0(ab)^i}\right)} E(s) + \frac{As + B}{(\tau_0 s)^2 + 2\zeta(\tau_0 s) + 1} E(s) \qquad (32)$$

for e(t) = δ(t) the unit impulse E(s) = 1, the impulse response of this system is given as:

$$x(t) = \sum_{i=0}^{N} k_i p_0 (ab)^i \exp\left(-p_0 (ab)^i t\right) + C \exp\left(-\frac{\zeta}{\tau_0} t\right) \sin\left(\frac{\sqrt{1-\zeta^2}}{\tau_0} t + \Phi\right) \qquad (33)$$

where the constants C and Φ are given as (17):

$$C = \frac{B}{\tau_0} \sqrt{\frac{A^2 - 2AB\zeta\tau_0 + (B\tau_0)^2}{(B\tau_0)^2\left(1 - \zeta^2\right)}}$$

$$\Phi = \operatorname{arctg}\left(\frac{A\sqrt{1-\zeta^2}}{B\tau_0 - A\zeta}\right)$$

Now, for e(t) = u(t) the unit step E(s) = 1/s, equation (32) we will be

$$X(s) = \sum_{i=0}^{N} \frac{k_i}{\left(1 + \frac{s}{p_0(ab)^i}\right)} \frac{1}{s} + \frac{As + B}{(\tau_0 s)^2 + 2\zeta(\tau_0 s) + 1} \frac{1}{s} \qquad (34)$$

the step response of this system can be obtained as:

$$x(t) = 1 - \sum_{i=0}^{N} k_i \exp\left(-p_0 (ab)^i t\right) + C_1 \exp\left(-\frac{\zeta}{\tau_0} t\right) \sin\left(\frac{\sqrt{1-\zeta^2}}{\tau_0} t + \Phi_1\right) \qquad (35)$$

where the constants C_1 and Φ_1 are given as (Kuo, 1987):

$$C_1 = B \sqrt{\frac{A^2 - 2AB\zeta\tau_0 + (B\tau_0)^2}{(B\tau_0)^2\left(1 - \zeta^2\right)}}$$

$$\Phi_1 = \operatorname{arctg}\left(\frac{A\sqrt{1-\zeta^2}}{B\tau_0 - A\zeta}\right) - \operatorname{arctg}\left(\frac{\sqrt{1-\zeta^2}}{-\zeta}\right)$$

3.4 Illustrative Example

Let's take a numerical example for an oscillation fractional order system represented by the following fundamental linear fractional order differential equation with m = 1.7 and $\tau_0 = 0.1$ as:

$$(0.1)^{1.7} \frac{d^{1.7} x(t)}{dt^{1.7}} + x(t) = e(t)$$

its transfer function is given by:

$$G(s) = \frac{1}{1 + (0.1s)^{1.7}}$$

First, G(s) is modeled by the following function:

$$G(s) = \frac{1}{[1 + (0.1s)^{1.7}]} = \frac{(1 + 0.1s)^{(0.3)}}{(0.1s)^2 + 0.52(0.1s) + 1}$$

For a frequency band of practical interest $[0, \omega_H]$ = [0, 1000 rad/s], the approximation of the fractional

power zero $(1+0.1s)^{(0.3)}$ by a rational function is given as:

$$(1+0.1s)^{(0.3)} = \frac{\prod_{i=0}^{N}\left(1+\frac{s}{z_0(ab)^i}\right)}{\prod_{i=0}^{N}\left(1+\frac{s}{p_0(ab)^i}\right)}$$

for an approximation error y = 1 dB and an approximation frequency band $\omega_{max} = 100\omega_H$ = 100000 rad/s, the parameters a, b, z_0 , p_0 and N of the above equation can be easily calculated as follows : a = 1.389, b = 2.154, z_0 = 14.678 rad/s, p_0 = 20.395 rad/s and N = 9, so:

$$(1+0.1s)^{(0.3)} = \frac{\prod_{i=0}^{9}\left(1+\frac{s}{14.678(2.993)^i}\right)}{\prod_{i=0}^{9}\left(1+\frac{s}{20.395(2.993)^i}\right)}$$

then, we will have that:

$$G(s) = \frac{\prod_{i=0}^{9}\left(1+\frac{s}{14.678(2.993)^i}\right)}{\prod_{i=0}^{9}\left(1+\frac{s}{20.395(2.993)^i}\right)} \frac{1}{(0.1s)^2+0.52(0.1s)+1}$$

Figures (5) and (6) show the Bode plots of the system transfer function and its proposed rational function approximation. Figures (7) and (8) show respectively the impulse and the step responses of the system obtained from its proposed rational function approximation.

4 CONCLUSION

In this paper I have presented some effective methods for approximating the irrational function given by $G(s) = \frac{1}{[1+(\tau_0 s)^m]}$, for 0 < m < 2, representing the transfer function of the fundamental linear fractional order differential equation $(\tau_0)^m \frac{d^m x(t)}{dt^m} + x(t) = e(t)$ by a rational function, in a given frequency band. The impulse and step responses of this type of systems are derived. Illustrative examples have been treated to demonstrate the usefulness of the approximation methods.

Theses approximations can very suitable for analysis, realization and implementation of fractional order systems. The expressions for characteristics and usual time and frequency specifications can also be derived.

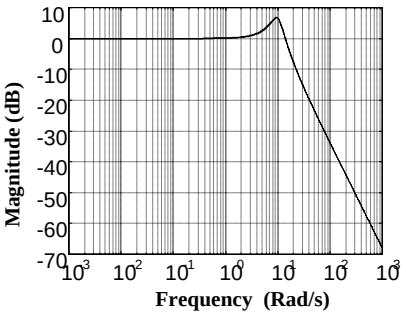


Figure 5: Magnitude Bode plot.

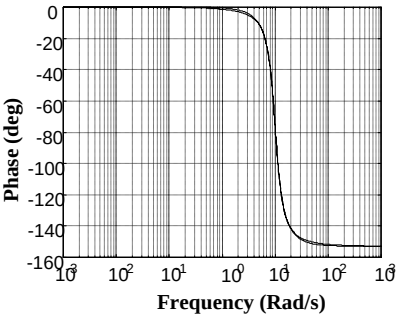


Figure 6: Phase of the Bode plot.

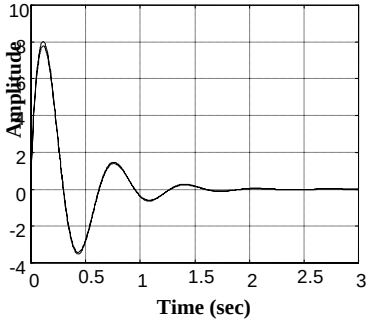


Figure 7: Impulse response.

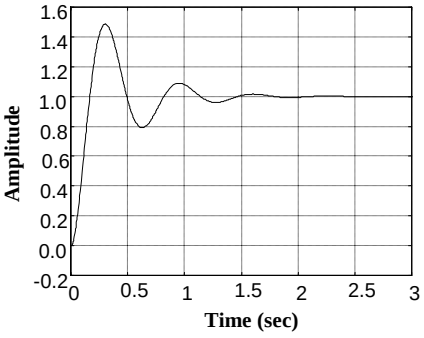


Figure 8: Step response.

REFERENCES

Torvik, P.J. and Bagley, R. L.,1984, 'On the Appearance of the Fractional Derivative in the Behavior of Real Materials,' Transactions of the ASME, vol. 51.

Ichise, M., Nagayanagi, and Y., Kojima, T., 1971 "An Analog Simulation of Non-Integer Order Transfer Functions for Analysis of Electrode Processes," J. of Electro-analytical Chemistry, vol. 33.

Sun, H.H. and Onaral, B., 1983, 'A Unified Approach to Represent Metal Electrode Polarization,' IEEE Transactions on Biomedical Engineering, vol. 30.

Cole, K.S. and Cole, R.H., 1941, 'Dispersion and absorption in dielectrics, alternation current characterization,' Journal of Chem. Physics vol. 9.

Davidson, D. and Cole, R., 1950, 'Dielectric relaxation in glycerine,' J. Chem. Phys., vol.18.

Manabe, S., 1961, 'The Non-Integer Integral and its Application to Control Systems,' ETJ of Japan, vol. 6, N° 3-4.

Oustaloup, A., 1983, Systèmes Asservis Linéaires d'Ordre Fractionnaire : Théorie et Pratique-, Editions Masson, Paris.

Charef, A., Sun, H. H., Tsao, Y.Y., and Onaral, B., 1992, 'Fractal system as represented by singularity function,' IEEE Transactions on Automatic Control, Vol. 37, N°9.

Podlubny, I., 1994, 'Fractional-order Systems and fractional-Order Controllers,' UEF-03-94 Slovak Academy of Science, Kosice.

Miller, K.S. and Ross, B., 1993, An Introduction to the Fractional Calculus and Fractional Differential Equations, John Wiley & Sons Inc., New-York.

Hartley, T.T. and Lorenzo C. F., 1998, 'A solution of the fundamental linear fractional order differential equation,' NASA TP-1998-208693, December 1998

Petras, I., Podlubny, I., O'Leary, P., Dorcak, L., and Vinagre, B. M., 2002, 'Analogue Realization of Fractional Order Controllers,' Fakulta Berg , TU Kosice.

MacDonald, J.R., 1987, Impedance spectroscopy, John Wiley, New York.

Sun, H. H., Charef, A., Tsao, Y.Y., and Onaral, B., 1992, 'Analysis of Polarization Dynamics by Singularity Decomposition Method,' Annals of Biomedical Engineering, Vol. 20.

Kuo, Benjamin C., 1987, Automatic control systems, Englewood Cliffs, Prentice-Hall, Englewood Cliffs, New Jersey.

ROBUST ADAPTIVE CONTROL USING A FRACTIONAL FEEDFORWARD BASED ON SPR CONDITION

Samir Ladaci, Jean Jacques Loiseau
IRCCyN, Ecole Centrale de Nantes, 1, rue de la Noë, BP: 92101 Nantes, 44321, France
{*Samir.Ladaci,Jean-Jacques.Loiseau*}*@irccyn.ec-nantes.fr*

Abdelfatah Charef
Département d'Electronique, Université Mentouri, Route de Ain Elbey, Constantine 25000, Algeria
afcharef@yahoo.com

Keywords: Robust adaptive control, Fractional Adaptive Control, Model Reference Adaptive Control, Feedforward, Fractional order systems.

Abstract: This paper presents a new approach for robust adaptive control, using fractional order systems as parallel feedforward in the adaptation loop. The basic adaptive algorithm used here is Model Reference Adaptive Control (MRAC), which do not require explicit parameter identification. The problem is that such a control system may diverge when confronted with finite sensor and actuator dynamics, or with parasitic disturbances. One of the classical robust adaptive control solutions to these problems, makes use of parallel feedforward and simplified adaptive controllers based on the concept of positive realness.
This control scheme is based on the ASPR property of the plant. We show that this condition implies also robust stability in case of fractional order controllers. A simulation example of a SISO robust adaptive control system illustrates the interest of the proposed method in the presence of disturbances and noises.

1 INTRODUCTION

Adaptive control has proven to be a good control solution for the partially unknown systems or varying parameter systems. In this domain Model reference adaptive control (MRAC) became very popular since it presents a very simple algorithm with easy implementation and does not require identifiers or observers in the control loop (Astrom and Wittenmark, 1995; Landau, 1979). However such algorithm shows its limits in noisy or disturbed environment, which may make it inefficient or uncompetitive. Unfortunately very few industrial control processes are not subject to theses practical problems, which can damage the quality of product and the good process operating.
The use of simple parallel feedforward in the adaptation loop appeared as a robust solution since the 80's. Many works have used this approach towards robust control systems (Bar-Kana, 1987; Naceri and Abida, 2003). In the last decade a great interest was given to fractional order systems, which have shown good robustness performances, several robust control methods based on these systems have been developed, like CRONE Control (Oustaloup, 1991) and fractional adaptive control (Vinagre et al., 2002; Ladaci and Charef, 2006; Ladaci et al., 2007).
In this paper we present a fractional robust adaptive control solution for disturbed applications, based on the idea of Bar-kana (Bar-Kana, 1987), which uses the basic stabilizability property of the plant and simple parallel feedforward in order to satisfy the desired "almost positive realness" condition that can guarantee robust stability of the nonlinear adaptive controller.
The main contribution of this work is to improve the feedforward approach robust performances by using fractional order filters. This result is illustrated by a simulation example of a test in bad realistic conditions like finite bandwidth of actuators, input and output disturbances and no assumed natural damping.
This paper is structured as follows:
In section 2 definitions of fractional order systems are presented. Section 3 introduces the principles of robust adaptive control based on the concept of 'positive realness' condition and then the main result in fractional order case is presented in section 4. The implementation in Model Reference Adaptive Control scheme is introduced in section 5 and a simulation

example is given in section 6. The paper is concluded in section 7.

2 FRACTIONAL ORDER SYSTEMS

The analysis in Bode plot of many natural processes, like transmission lines, dielectric polarisation impedance, interfaces, cardiac rhythm, spectral density of physical wave, some types of noise (VanDerZiel, 1950; Duta and Hom, 1981), has allowed to observe a fractional slope. This type of process is known as $1/f$ process or fractional order system. During the last decade, a great interest was given by researchers to the study of these systems (Sun and Charef, 1990) and their application in control systems (Oustaloup, 1991; Hotzel and Fliess, 1997; Ladaci and Charef, 2006; Ladaci et al., 2007).
A SISO fractional order system can be represented by the following transfer function,

$$X(s) = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + ... + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + ... + a_0 s^{\alpha_0}} \quad (1)$$

Where,

- α_i, β_j: real numbers such that,

$$\begin{cases} 0 \leq \alpha_0 < \alpha_1 < ... < \alpha_n \\ 0 \leq \beta_0 < \beta_1 < ... < \beta_m \end{cases}$$

- s: Laplace operator.

for the purpose of this work, let us introduce the following definitions,

Definition 1 *The fractional order transfer function $X(s)$ given in (1) is called proper if: $\beta_m \leq \alpha_n$*
It is called stricly proper if: $\beta_m < \alpha_n$

Definition 2 *(Desoer and Vidyasagar, 1975) The fractional order transfer function Matrix $M_X(s)$ whose elements are of the form (1) is proper (strictly proper) if and only if all elements of $M_X(s)$ are bounded at ∞ (tend to zero at ∞, resp.).*

We use in the sequel a description equation into frequency domain of a single pole fractional order process, given as follows:

$$Y(s) = \frac{1}{(s + p_T)^{\alpha}} \quad (2)$$

with

- α: fractional exponent, $0 \leq \alpha \leq 1$
- p_T: fractional pole which is the cut frequency.

Many previous works have shown that fractional systems present best qualities, in response time and in transition dynamic stability (Sun and Charef, 1990). All the control theory developed by Oustaloup especially on CRONE control was based on fractional order systems robustness in presence of uncertainties and perturbations (Oustaloup, 1991).

3 CONCEPT OF POSITIVE REALNESS CONDITION

Robustness is defined relatively to a certain property and a set of models. A property (generally stability or performance level) is said to be robust if all the models belonging to the set satisfy it. Robust adaptive stabilization means that all values involved in the adaptation process namely, states, gains and errors are bounded in the presence of any bounded input commands and input or output disturbances (Bar-Kana and Kaufman, 1985; Kwan et al., 2001).

In this paper we are interested by a particular configuration of feedforward controllers combined with MRAC control and fractional order systems giving a fractional robust adaptive control method.
The use of a simple feedforward in the adaptation loop (see Figure 4) improves the robust stability of the control system. This approach is based on the concept of the "positive realness" condition (Bar-Kana, 1989); witch can guarantee stable implementation of adaptive control configuration. Let us present these definitions:

Definition 3 *The $m \times m$ transfer function matrix $G_s(s)$ is called strictly positive real (**SPR**) if (Landau, 1979; Bar-Kana, 1989):*

1. *All elements of $G_s(s)$ are analytic in $\Re(s) \geq 0$.*
2. *$G_s(s)$ is real for real s.*
3. *$G_s(s) + G_s^{T*}(s) > 0$ for $\Re(s) \geq 0$ and finite s.*

We also show that (Shaked, 1977) for a fractional order transfer function matrix $G_s(s)$,

$$G_s(s) \text{ is } \mathbf{SPR} \Leftrightarrow G_s^{-1}(s) \text{ is } \mathbf{SPR} \quad (3)$$

Indeed, by using the **SPR** property if we write (Bar-Kana, 1989),

$$G_s(s) = A + jB \Rightarrow G_s^{T*}(s) = A^T - jB^T$$

Since by definition

$$G_s(s) + G_s^{T*}(s) = A + A^T + j(B - B^T) > 0$$

we get $B = B^T$ and $A > 0$ (not necessary symmetric). Then whenever

$$\Re[G_s(s)] = A > 0$$

we get

$$G_s^{-1}(s) = (A + BA^{-1}B^T)^{-1} - jA^{-1}B(A + BA^{-1}B^T)^{-1}$$

and

$$\Re\left[G_s^{-1}(s)\right] = (A + BA^{-1}B^T)^{-1} > 0$$

which proves (3).

Definition 4 *(Bar-Kana, 1987)*
Let $G_a(s)$ be a $m \times m$ transfer matrix. Let us assume that there exists a positive definite constant gain matrix, $\tilde{K}_e$ such that the closed-loop transfer function

$$G_c(s) = \left[I + G_a(s)\tilde{K}_e\right]^{-1} G_a(s) \tag{4}$$

is **SPR**. *$G_a(s)$ is called "almost strictly positive real (***ASPR***)".*

Now if we consider a fractional order proper or strictly proper **ASPR** transfer matrix $G_s(s)$.
Then the following statements are equivalent,

$$G_s(s) = [I + G_a(s)K_e]^{-1} G_a(s) \text{ is SPR} \tag{5}$$

$$G_s(s) = [I + G_a(s)K_e]^{-1} \text{ is SPR} \tag{6}$$

$$G_s^{-1}(s) = G_a^{-1}(s) + K_e \text{ is SPR} \tag{7}$$

$$\Re\left[G_a^{-1}(s) + K_e\right]_{\Re(s)\geq 0} > 0 \tag{8}$$

$$\begin{aligned} G_s^{-1}(s) &\quad \text{is asymptotically stable and} \\ K_e &\quad \text{is sufficiently large} \end{aligned} \tag{9}$$

Because $\exists M$ such that $\Re\left[G_a^{-1}(s)\right]_{\Re(s)\geq 0} > M > -\infty$, and then any $K_e > -M$ will do (Bar-Kana, 1989).

$$\begin{aligned} G_a(s) &\quad \text{is strictly minimum phase and} \\ K_e &\quad \text{is sufficiently large} \end{aligned} \tag{10}$$

All the above algebraic manipulation, as done to obtain (3) and definitions 3 and 4, apply to fractional systems as well. Here we can generalize as fellows the result of (Bar-Kana, 1989) to the fractional order case.

Lemma 1 *Let a fractional order transfer function matrix $G_a(s)$ be* **ASPR** *and let $\tilde{K}_e$ be any gain that satisfies (4). Then $G_a(s)$ is* **SPR** *for any gain K_e that satisfies $K_e > \tilde{K}_e$.*

It is obvious that **ASPR** fractional order systems, which are minimum phase proper systems maintain stability with high gains. The high gain stability is important when nonstationary or nonlinear (adaptive) control is used, because the robustness of the control system is maintained if, due to specific operational conditions, the time-varying gains become too large.

Remarks

1. The ASPR plant must also be proper.
2. The open loop is not necessarily stable (the plant will actually be stabilized by the fictitious gain K_e), however all the zeros must be placed in the left half plane. The plant must be minimum phase to obtain positivity.
3. We can easily show (Bar-Kana, 1987) that if a system is ASPR, then it can be stabilized by any constant or time variable output gain K_e, if it is large enough, i.e. $K_e > \tilde{K}_e$.

But in this method, instead of using high gain regulation we will use a simple parallel feedforward configuration which can by a similar way satisfy the positive realness conditions.
The idea of using feedforward in parallel with the controlled plant is based on the following Lemma of Bar-Kana,

Lemma 2 *(Bar-Kana, 1989) Let the plant be described by the $m \times m$ transfer function $G_p(s)$ of order n. Let $C(s)$ be any dynamic stability output feedback controller. Then*

$$G_a(s) = G_p(s) + C^{-1}(s) \tag{11}$$

is **ASPR** *if $C^{-1}(s)$ is proper or strictly proper.*

We can adapt the proof of (Bar-Kana, 1989; Bar-Kana, 1986)) to the fractional case.

4 MAIN RESULT

At this stage we propose a fractional order feedforward configuration of the form:

$$F(s) = \frac{F_p}{\left(1 + \frac{s}{s_0}\right)^{\alpha}} \tag{12}$$

with a real fractional power $0 < \alpha < 1$, to improve the robustness of the adaptive algorithm, in presence of perturbations, as such systems do not amplify much

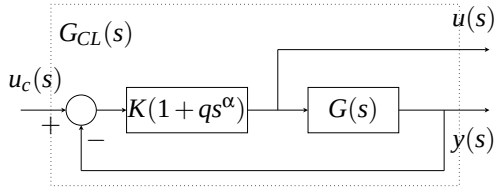


Figure 1: Closed-loop system.

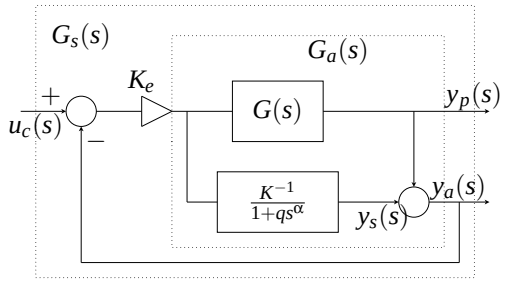


Figure 2: The fictitious SPR configuration.

these random signals. This configuration could be considered as the inverse of an improper fractional PD$^{\mu}$ controller, which was used in control systems with good proven performances (Oustaloup, 1983; Hotzel and Fliess, 1997; Podlubny, 1999).
We can formulate the main result of this paper in the following theorem.

Theorem 1 *Let $G(s)$ be any $m \times m$ strictly proper transfer matrix of arbitrary MacMillan degree. $G(s)$ is not necessarily stable or minimum phase. Let*

$$H_f(s) = K(1+qs^{\alpha}) \tag{13}$$

be some stabilizing controller for $G(s)$. Then the augmented controlled plant

$$G_a^f(s) = G(s) + H_f^{-1}(s) = G(s) + \frac{K^{-1}}{1+qs^{\alpha}} \tag{14}$$

is **ASPR**.

Proof of Theorem 1:
From definition 4, if $G_a(s)$ is **ASPR** then the closed-loop transfer function

$$G_c(s) = \left[I + G_a(s)\tilde{K}_e\right]^{-1} G_a(s)$$

is **ASPR**.

Since $H_f^{-1}(s)$ from (13) is strictly proper (relative degree $\alpha > 0$), then Lemma 2 implies that the augmented system $G_a^f(s)$ as defined in (14) is **ASPR**, which proves Theorem 1.

The stabilizing controller $H_f(s)$ can also be modelized as follows,

$$H_f(s) = K(1+qs)^{\alpha} \tag{15}$$

Figure 1 represents the feedback control system corresponding to the control (13).
From Definition 4 and the fact that the transfer function $G_a^f(s)$ is ASPR, we know that it can be stabilized by a gain $\tilde{K}_e$. Figure 2 illustrates the feedforward configuration. In addition, the stabilization is robust, it holds for any gain $K_e > \tilde{K}_e$.

Many previous works (Hotzel and Fliess, 1997; Podlubny, 1999) have proposed PD$^{\mu}$ improper controllers of the form (13):

$$C(s) = K_p + K_i s^{\alpha} \tag{16}$$

which can stabilize many realistic plants for sufficient high values of K.
A feedforward of equivalent effect is chosen as follows:

$$F(s) = C^{-1}(s) = \frac{F_p}{\left(1+\frac{s}{s_0}\right)^{\alpha}} \tag{17}$$

Where $F_p = K^{-1}$, such that the augmented plant becomes:

$$G_a(s) = G_p(s) + F(s) \tag{18}$$

As K should be very large, so F_p are small coefficients, guaranteeing that $G_a(s)$ be **ASPR**. And during the control design we can take $G_a(s) \approx G_p(s)$ as a practical approximation.

5 IMPLEMENTATION IN MRAC SCHEME

Model Reference Adaptive Control (MRAC) is one of the more used approaches of adaptive control, in which the desired performance is specified by the choice of a reference model. Adjustment of parameters is achieved by means of the error between the output of the plant and the model reference output. Let us introduce the basic ideas of this approach represented in Figure 3.

We consider a closed loop system where the controller has an adjustable parameter vector θ. A model which output is y_m specifies the desired closed loop response. Let e be the error between the closed loop system output y and the model one y_m, one possibility is to adjust the parameters such that the cost function:

$$J(\theta) = \frac{1}{2}e^2 \tag{19}$$

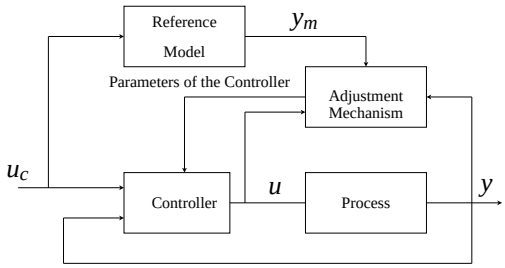


Figure 3: Direct Model Reference Adaptive Control.

be minimised. In order to make J small it is reasonable to change parameters in the direction of negative gradient J, so:

$$\frac{d\theta}{dt} = -\gamma\frac{\delta J}{\delta\theta} = -\gamma e\frac{\delta e}{\delta\theta} \tag{20}$$

or

$$\frac{d\theta}{dt} = \gamma\varphi e \tag{21}$$

where $\varphi = -\frac{\delta e}{\delta\theta}$ is the regression (or measures) vector and γ is the adaptation gain. This aproach is called *M.I.T. rule*.
The introduction of a simple feedforward in the MRAC adaptation loop us represented in figure 4 improves the robust stability performance against the controller gain fluctuations in presence of perturbation and noises (Naceri and Abida, 2003). Previous works (Sobel and Kaufman, 1986), showed that the **ASPR** property of a process, allows the implementation of very simple adaptive controllers that garantee robust stability of the closed loop in presence of bounded input or output disturbances.
The feedforward transfer function is choosen like in (12) where the gain F_p is a small coefficient.

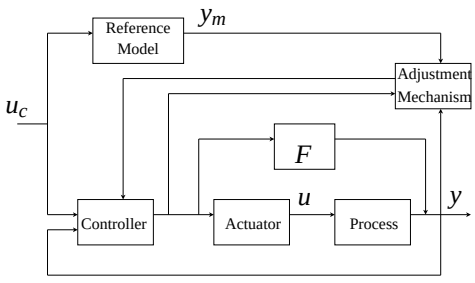


Figure 4: Simple feedforward in MRAC scheme.

6 SIMULATION EXAMPLE

Without any loss of generality we will apply this robust adaptive control method, both in the case of integer and fractional order feedforward, to a SISO model of a DC motor controlled in respect of velocity, given by:

$$G_p(z) = \frac{0.8513z + 5.099\,10^{-6}}{z^2 + 2.442\,10^{-7}z + 1.37\,10^{-11}} \tag{22}$$

with a sampling period $T_s = 0.3sec$, and an actuator model of the form:

$$A(z) = \frac{0.007667z + 0.007049}{z^2 - 1.763z + 0.7772} \tag{23}$$

The plant is subject to random input and output perturbations of amplitudes 2 and 0.05 respectively.
The reference model G_m is given by:

$$G_m(z) = \frac{0.9411z + 0.1208}{z^2 + 0.05679z + 0.005092} \tag{24}$$

6.1 Integer Order Feedforward Case

The feedforward trunsfer fuction F is given by:

$$F(z) = \frac{3.2394\,10^{-7}}{z - 0.9997} \tag{25}$$

with a regulation parameter $\gamma = 0.001$ we obtain the results of Figure 5.

6.2 Fractional Order Feedforward Case

The fractional order feedforward trunsfer function F is given in Laplace domain by:

$$F(s) = \frac{0.001}{(s + 500)^{0.6}} \tag{26}$$

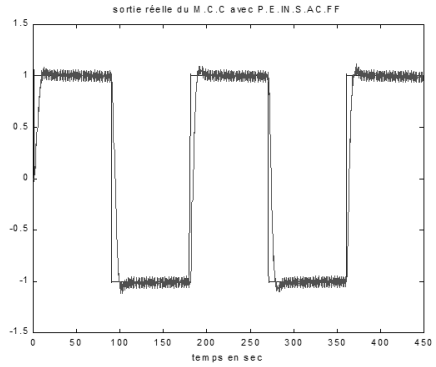
For the purpose of our approach we need to use an integer order model approximation of the fractional order feedforward model in order to implement the adaptation algorithm, for this aim we have used the so-called singularity function method (Charef et al., 1992).
The fractional transfer function (26) is approximated to a linear transfer function and sampled to give the following formula:

$$\hat{F}(z) = \frac{0.001(z - 4.78\,10^{-97})}{z^2 - 2.407\,10^{-96}z + 1.001\,10^{-207}} \tag{27}$$

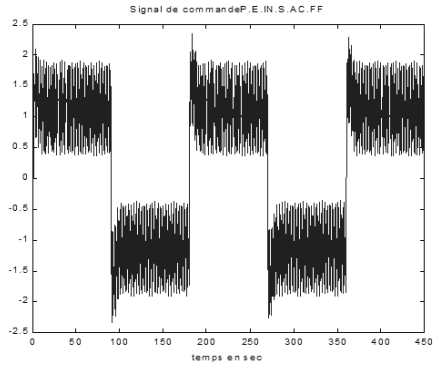
with a regulation parameter $\gamma = 0.005$, we obtain the results of Figure 6.

6.3 Remarks

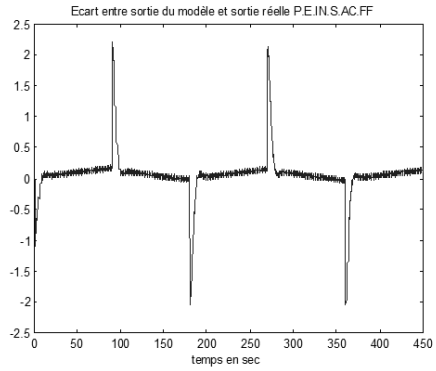
- The command signal u is more polish in the fractional case witch is a very useful property in regulation problem.



(a)

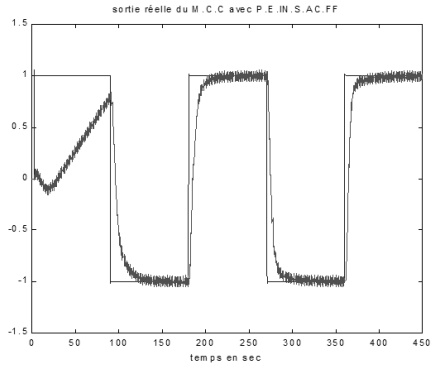


(b)

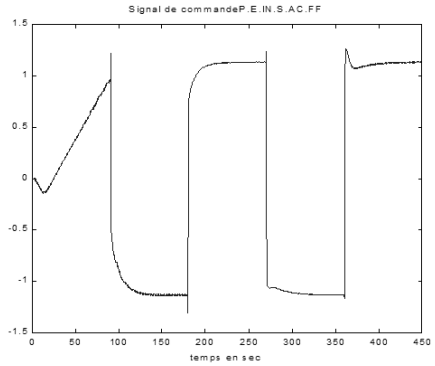


(c)

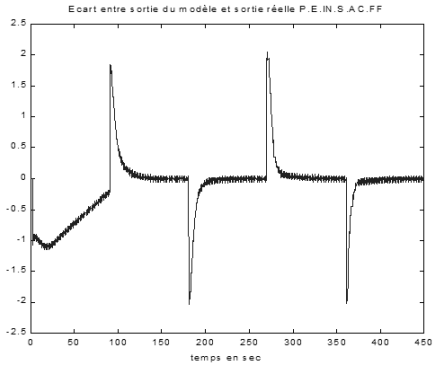
Figure 5: Process output with integer feedforward (a) Process output $y(t)$, (b) Control signal $u(t)$, (c) Error signal $e(t)$.



(a)



(b)



(c)

Figure 6: Process output with fractional feedforward (a) Process output $y(t)$, (b) Control signal $u(t)$, (c) Error signal $e(t)$.

- The proposed fractional order configuration of feedforward maintains stability and at less the same level of performances, witch confirms the interest of integrating fractional strategy in robust adaptive control.

7 CONCLUSION

In this paper we have presented a new robust adaptive control strategy, by introducing simple fractional feedforward configuration in the MRAC algorithm. The concept of positive realness condition which is the basis of this robust control strategy is extended to fractional order control systems. The idea was to take benefit of the high performance quality of fractional

order systems confirmed in many precedent research works. The stability proofs of this adaptive control scheme developed for integer order filters in control literature still holds for such systems. Simulation results have shown a better filtering ability of command and output signals, and more robustness against additive perturbations, than in the integer order feedforward configuration case.

REFERENCES

Astrom, K. and Wittenmark, B. (1995). *Adaptive Control*. Addison-Wesley, USA.

Bar-Kana, I. (1986). Positive realness in discrete-time adaptive control systems. *International Journal of Systems Science*, 17(7):1001–1006.

Bar-Kana, I. (1987). Parallel feedforward and simplified adaptive control. *Int. J. Adaptive Control and Signal Processing*, 1(2):95–109.

Bar-Kana, I. (1989). On positive realness in multivariable stationary linear systems. In *Conference on Information Sciences and Systems*, Baltimore, Maryland, USA.

Bar-Kana, I. and Kaufman, H. (1985). Global stability and performance of a simplified adaptive algorithm. *Int. J. Control*, 42(6):1491–1505.

Charef, A., Sun, H., Tsao, Y., and Onaral, B. (1992). Fractal system as represented by singularity function. *IEEE Trans. On Automatic Control*, 37:1465–1470.

Desoer, C. and Vidyasagar, M. (1975). *Feedback Systems: Input-Output Properties*. Academic Press, N.Y. USA.

Duta, P. and Hom, P. (1981). Low frequency fluctuations in solids: 1/f noise. *Review of modern physics*, 53(3).

Hotzel, R. and Fliess, M. (1997). Systèmes linéaires fractionnaires avec et sans retard : Stabilité, commande, exemples. In *Actes d'AGIS'97*, pages 53–58, Angers.

Kwan, C., Dawson, D., and Lewis, F. (2001). Robust adaptive control of robots using neural network: Global stability. *Asian Journal of Control*, 3(2):111–121.

Ladaci, S. and Charef, A. (2006). On fractional adaptive control. *Nonlinear Dynamics*, 43:365–378.

Ladaci, S., Loiseau, J., and Charef, A. (2007). Fractional order adaptive high-gain controllers for a class of linear systems. *Communications in Nonlinear Science and Numerical Simulations. Elsevier, In Press*.

Landau, Y. (1979). *Adaptive Control : The model reference Approach*. Marcel Dekker, New York.

Naceri, F. and Abida, L. (2003). A novel robust adaptive control algorithm for ac drives. *Computers and Electrical Engineering*.

Oustaloup, A. (1983). *Systmes asservis d'ordre fractionnaire*. Masson, Paris.

Oustaloup, A. (1991). *La commande CRONE*. Hermès, Paris.

Podlubny, I. (1999). Fractional order systems and pi$^{\lambda}$d$^{\mu}$ controllers. *IEEE Transactions on Automatic Control*, 44(1):208–214.

Shaked, U. (1977). The zero properties of linear passive systems. *IEEE Transactions on Automatic Control*, 22(6):973–976.

Sobel, K. and Kaufman, H. (1986). Direct model reference adaptive control for a class of mimo systems. *in C. Leondes (ed.) Control and Dynamic Systems- Advances in Theory and Applications*, 24.

Sun, H. and Charef, A. (1990). Fractal system-a time domain approach. *Annals of Biomedical Ing.*

VanDerZiel, A. (1950). On the noise spectra of semiconductor noise and of flikker effects. *Physica*.

Vinagre, B., Petras, I., and Chen, Y. (2002). Using fractional order adjustment rules and fractional order reference models in model-reference adaptive control. *Nonlinear Dynamics*, 29:269–279.

CRONE OBSERVER
Definition and Design Methodology

Jocelyn Sabatier, Patrick Lanusse and Mathieu Merveillaut
LAPS - ENSEIRB - Université Bordeaux 1 - Equipe CRONE – UMR 5218 CNRS
351, Cours de la Libération, 33405 Talence cedex, France
{firstname.name}@laps.ims-bordeaux.fr

Keywords: Robust observer, CRONE observer, Fractional Order Controller, CRONE control.

Abstract: CRONE control, robust control methodology based on fractional differentiation, is applied to state observer design. State observation can indeed be viewed as a regulation problem given that the goal of a state observer is to cancel the observation errors in spite of measurement noises, disturbances and plant perturbations. This conclusion has been used recently to define a new class of state observers known in the literature as "dynamic observers" or "input-output observer". It is based on the observation error dynamic feedback. In this paper, this idea is used to define the CRONE observer design methodology. Performance robustness of the obtained observers versus plant perturbations is analysed. As for CRONE control, fractional differentiation in the definition of an equivalent open loop transfer function permits to reduce the number of parameters to be optimised.

1 INTRODUCTION

In many industrial applications of control, controlled variables cannot be directly measured by sensors. In such a situation, these variables can be reconstructed with a Luenberger type observer (Luenberger, 1971). However, it is really difficult to take into account modelling errors and disturbances in the synthesis the observer gains. We recently faced with this problem, for the speed control of a steel rolling mill, speed of the load being not measured due the high temperatures and maintenance costs (Sabatier et *al*., 2003). Moreover, some parameters of the system were not known with accuracy (such as sliding viscous coefficients). To solve this problem, a Luenberger observer was associated with a CRONE controller (Oustaloup, 1991). In this application of CRONE control, an overestimation of the plant uncertainties was required to take into account bias introduced by the observer due to differences between plant and observer model behaviours as the time of plant parameters variations. To reduce the resulting conservatism, a robust observer has to be designed, robustness of the observation error convergence to zero in spite of disturbances and plant perturbation being addressed.
A solution to obtain such an observer, consists in considering observation problem as a classic regulation problem and thus to construct a feedback loop with the available information (plant input and output), whose goal is to cancel the observation errors in spite of measurement noise, disturbances and plant perturbations. This new concept was recently published and applied on a real system (Marquez, 2003) (Marquez and Riaz, 2005). In this paper, a CRONE controller is introduced in the feedback loop in order to take into account the disturbances and the model perturbation. In comparison with the H_∞ approach used by Marquez, plant model perturbations are taken into account in a structured form with no overestimation, thus, without conservatism. Due to the introduction of fractional differentiation in the CRONE approach, an open loop transfer function with only three parameters (just like a PID controller) has to be optimised to simultaneously reduce the effects of disturbances and model perturbation on the observation error. Another contribution of the paper is the extension of the idea by Marquez to the problem of state observation with unknown input.
The paper is organised as follows. Section 2 presents the dynamic output feedback based observer concept developed in (Marquez, 2003) (Marquez and Riaz, 2005) and extends it to observation with unknown input. Section 3 gives some generalities on CRONE control. In section 4, application of CRONE control

to state observation problem is developed thus defining an observer that will be referred to as a CRONE observer in future developments.

2 DYNAMIC OUTPUT FEEDBACK BASED OBSERVER

2.1 Presentation

Dynamic output feedback based observer concept was introduced in (Marquez, 2003) and (Marquez and Riaz, 2005) in which the observation problem is solved using the feedback diagram of Fig. 1. The plant *P*, the model *M* and the dynamic controller *K* are supposed single input / single output systems represented by the state space descriptions:

$$P:\begin{cases}\dot{x}(t)=Ax(t)+Bu(t)\\ y(t)=Cx(t)\end{cases} \quad (1)$$

$$M:\begin{cases}\dot{\hat{x}}(t)=A\hat{x}(t)+Bv(t)=A\hat{x}(t)+B(u(t)+w(t))\\ \hat{y}(t)=C\hat{x}(t)\end{cases} \quad (2)$$

$$K:\begin{cases}\dot{x}_K(t)=A_k x_K(t)-B_K\varepsilon(t)\\ \qquad\quad =A_K x(t)+B_K C(x(t)-\hat{x}(t))\\ w(t)=C_K x_K(t)\end{cases} \quad (3)$$

State $x(t)$ is supposed not measurable and $\hat{x}(t)$ denotes the estimated state. All the elements of matrices and vectors in (1) to (3) are supposed element of $\mathbb{R}$.
Figure 1 clearly shows that the goal of the used feedback structure is to cancel the observation error $\chi(t)=x(t)-\hat{x}(t)$ by cancelling the error signal $\varepsilon=\hat{y}(t)-y(t)$. Time derivative of the observation error $\dot{\chi}(t)=\dot{x}(t)-\dot{\hat{x}}(t)$, is thus given by :

$$\begin{aligned}\dot{\chi}(t)&=Ax(t)+Bu(t)-A\hat{x}(t)-B(u(t)+w(t))\\ &=A\chi(t)-BC_K x_K(t)\end{aligned}. \quad (4)$$

Using controller state space description (3), a state space description for the system in Fig. 1 involving the observation error is thus:

$$\begin{bmatrix}\dot{\chi}(t)\\ \dot{x}_K(t)\end{bmatrix}=\begin{bmatrix}A & -BC_K\\ B_K C & A_K\end{bmatrix}\begin{bmatrix}\chi(t)\\ x_K(t)\end{bmatrix}=A_O\begin{bmatrix}\chi(t)\\ x_K(t)\end{bmatrix} \quad (5)$$

Matrix A_O in relation (5) is also the state matrix of the feedback system in Fig. 2. Such a remark permits to demonstrate the following theorem.

Theorem (Marquez, 2003)
State $\hat{x}(t)$ exponentially converge to the state $x(t)$ with the feedback structure of Fig. 1, if all matrix A_O eigenvalues of has a strictly negative part or if the system in Fig. 2 is internally stable.

2.2 Extension to State Observation with Unknown Input

The problem of state observation with unknown input is now addressed using the dynamic output feedback structure of Fig. 3.
It is supposed that the plant *P* and the model *M* are described by the following state space descriptions:

$$P:\begin{cases}\dot{x}(t)=Ax(t)+Bu(t)\\ y(t)=Cx(t)\end{cases} \quad (6)$$

$$M:\begin{cases}\dot{z}(t)=Nz(t)+Bv(t)\\ \hat{x}(t)=z(t)-Ey(t)\\ \hat{y}(t)=C\hat{x}(t)\end{cases} \quad (7)$$

Observation error dynamics is thus defined by:

$$\dot{\chi}(t)=\dot{x}(t)-\dot{\hat{x}}(t)=Ax(t)+Bu(t)-(\dot{z}(t)-E\dot{y}(t)) \quad (8)$$

or using relations (6) and (7):

$$\begin{aligned}\dot{\chi}(t)=&Ax(t)-NECx(t)+ECAx(t)\\ &-N\hat{x}(t)+Bu(t)+ECBu(t)-Bv(t)\end{aligned}. \quad (9)$$

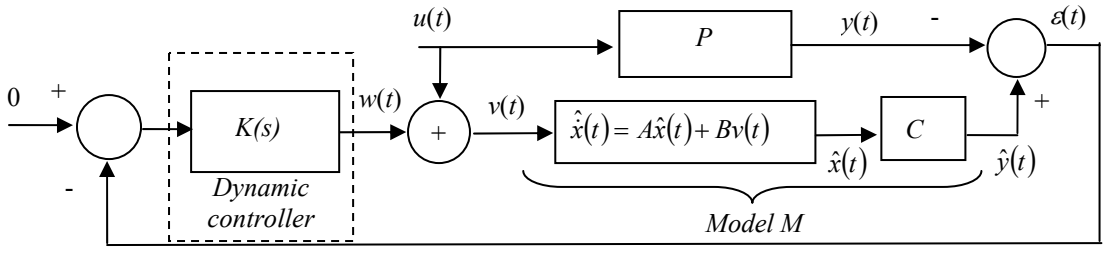


Figure 1: Dynamic output feedback based observer.

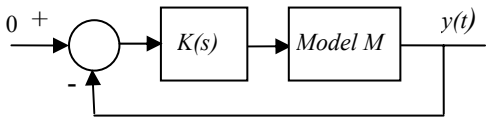


Figure 2: Feedback system with state matrix A_O

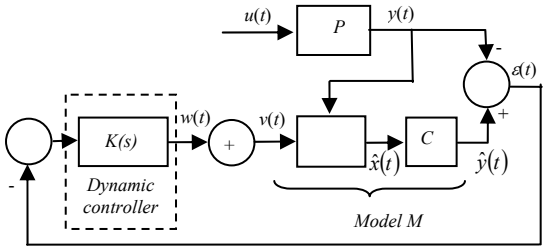


Figure 3: Dynamic output feedback based observer with unknown input.

Suppose now that matrix E is such that

$$B + ECB = 0 \qquad \text{or} \qquad E = -B(CB)^{*} \qquad (10)$$

in which $(CB)^{*}$ denotes the generalised inverse of CB if it exits. Equation (9) thus becomes:

$$\dot{\chi}(t) = Ax(t) - NECx(t) + ECAx(t) - N\hat{x}(t) - Bv(t), \quad (11)$$

or using the state space description of the controller K of relation (3):

$$\begin{cases} \dot{\chi}(t) = Ax(t) - NECx(t) + ECAx(t) \\ \qquad\qquad - N\hat{x}(t) - BC_K x_K(t). \\ \dot{x}_K(t) = A_K x_K(t) + B_K C\chi \end{cases} \quad (12)$$

Let now

$$P = I + EC \qquad \text{and thus} \qquad EC = P - I \qquad (13)$$

then

$$\begin{aligned} A - NEC + ECA &= A - N(P - I) + (P - I)A \\ &= -NP + N + PA \end{aligned}. \quad (14)$$

If it is now imposed now that

$$-NP + PA = 0 \quad \text{and thus} \quad N = PAP^{-1}, \quad (15)$$

equation (12) becomes:

$$\begin{bmatrix} \dot{\chi}(t) \\ \dot{x}_K(t) \end{bmatrix} = \begin{bmatrix} N & -BC_K \\ B_K C & A_K \end{bmatrix} \begin{bmatrix} \chi(t) \\ x_K(t) \end{bmatrix}. \quad (16)$$

Relation (16) is similar to relation (5) and thus highlights, given the analysis following relation (5), that the observation error converges exponentially to zero if controller K internally stabilise the feedback system in Fig. 2, model M being defined by:

$$M : \begin{cases} \dot{\chi}(t) = N\chi(t) + Bv(t) \\ y(t) = C\chi(t) \end{cases}. \quad (17)$$

3 CRONE CSD PRINCIPLES

3.1 Introduction to Fractional Integro-Differentiation

The first definitions of fractional order differentiation (or integration) were given by Leibniz and Euler at the end of the 17th and during the 18th century. In the 19th century many mathematicians generalized these definitions: Laplace, Lacroix, Fourier, Liouville, Abel, Hargreave, Riemann etc. In 1869 Sonin extended the Cauchy integral to fractional integration orders and the Riemann-Liouville definition was finally proposed.

Operational calculus can also be used. Let $y(t)$ be the order n derivative of the causal signal $x(t)$:

$$y(t) = x^{\{n\}}(t) = D^n x(t) \qquad (18)$$

with $n \in \mathbb{C}$ and where D is the differentiation operator. If the real part of n is negative, then $y(t)$ is in fact the order $-n$ integral of $x(t)$.

The transfer function of the linear operator D^n is defined by the Laplace transform:

$$D(s) = L\{y(t)\} / L\{x(t)\} = s^n. \qquad (19)$$

Its impulse response is given by:

$$d(t) = L^{-1}\{s^n\} = \frac{t^{-n-1}}{\Gamma(-n)} \mathrm{H}(t) \text{ if } \mathrm{Re}[n] \in \mathbb{R}^{-} \text{ and } \mathrm{Re}[n] \neq 0,$$

$$\text{or } \forall t \neq 0 \text{ if } \mathrm{Re}[n] \in \mathbb{R}^{+} - \mathbb{N} \text{ and } \mathrm{Re}[n] = 0 \qquad (20)$$

where $\Gamma(.)$ and $H(.)$ denote the gamma and Heaviside functions.

Convoluting $d(t)$ and $x(t)$, $y(t)$ can be computed using the following integrals:

$$y(t) = \int_0^t \frac{\theta^{-n-1}}{\Gamma(-n)} x(t-\theta) d\theta \qquad (21)$$

if $\mathrm{Re}[n] \in \mathbb{R}^{-}$ and $\mathrm{Re}[n] \neq 0$ which is the Riemann-Liouville definition, and

$$y(t) = \left(\frac{d}{dt}\right)^m \int_0^t \frac{\theta^{m-n-1}}{\Gamma(-n)} x(t-\theta) d\theta \qquad (22)$$

if $\mathrm{Re}[n] \in \mathbb{R}^+ - \mathbb{N}$ and $\mathrm{Re}[n]=0$, where m is defined by the integer part of the real part of n. It is obvious that a specificity of this fractional differentiation, is that it takes into account all the past of signal $x(t)$. A fractional-order system can be considered as an infinite order rational system. Thus, fractional systems are often used to model distributed parameter systems. As fractional operators can replace high order transfer functions in system-identification or control-system design, they are also used to determine models or controllers with few tuning parameters.

Since the sixties, some electrical circuits have been proposed for synthesizing half order differentiators (Suezaki and Takahashi,1966), (Dutta Roy, 1970), (Biorci and Ridella, 1970), (Ichise *et al.*, 1971), (Oldham, 1973). From 1975 on, Oustaloup et al. proposed methodologies for synthesizing band-limited differentiators whose orders are fractional (Oustaloup, 1975). Since 1990, they have extended this to complex fractional order differentiators (Oustaloup et al., 1990), (Oustaloup et al., 2000) and have applied it to robust control design. Fractional or non-integer order systems are also termed Warburg impedance or Constant Phase Element (CPE), and are associated to long-time memory behaviours.

3.2 Introduction to the CRONE Methodology

The CRONE control-system design is based on the common unity-feedback configuration (Fig. 4). The robust controller or the open-loop transfer function is defined using fractional order integro-differentiation. The required robustness is that of both stability margins and performance, and particularly peak value M_r (called resonant peak) of the common complementary sensitivity function $T(s)$.

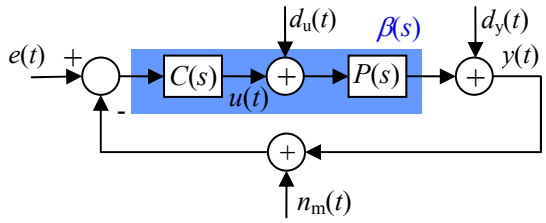


Figure 4: Common CRONE control-system diagram.

Three CRONE control design methods have been developed, successively extending the application field.

The third CRONE control generation must be used when the plant frequency uncertainty domains are of various types (not only gain-like). It is based on the definition of a generalized template described as a straight line in the Nichols chart of any direction (complex fractional order integration), or by a multi-template (or curvilinear template) defined by a set of generalized templates.

An optimization allows the determination of the independent parameters of the open loop transfer function. This optimization is based on the minimization of the stability degree variations, while respecting other specifications taken into account by constraints on sensitivity function magnitude. The complex fractional order permits parameterization of the open-loop transfer function with a small number of high-level parameters. The optimization of the control is thus reduced to only the search for the optimal values of these parameters. As the form of uncertainties taken into account is structured, this optimization is necessarily nonlinear. It is thus very important to limit the number of parameters to be optimized. After this optimization, the corresponding CRONE controller is synthesized as a rational fraction only for the optimal open-loop transfer function.

The third generation CRONE CSD methodology, the most powerful one, is able to design controllers for plants with positive real part zeros or poles, time delay, and/or with lightly damped modes (Oustaloup *et al.*,1995). Associated with the w-bilinear variable change, it also permits the design of digital controllers. The CRONE control has also been extended to linear time variant systems and nonlinear systems whose nonlinear behaviors are taken into account by sets of linear equivalent behaviors (Pommier *et al.*, 2002). For MIMO (multivariable) plants, two methods have been developed (Lanusse et *al.*, 2000). The choice of the method is made through an analysis of the coupling rate of the plant. When this rate is reasonable, one can opt for the simplicity of the multi SISO approach.

3.3 Third Generation CRONE Methodology

Within a frequency range $[\omega_A, \omega_B]$ around open-loop gain-crossover frequency ω_{cg}, the Nichols locus of a third generation CRONE open-loop is defined by an any-angle straight line segment, called a generalized template (Fig. 5).

The generalized template can be defined by an integrator of complex fractional order n whose real part determines its phase location at frequency ω_{cg}, that is $-\mathrm{Re}_{/i}(n)\pi/2$, and whose imaginary part then determines its angle to the vertical (Fig. 5).

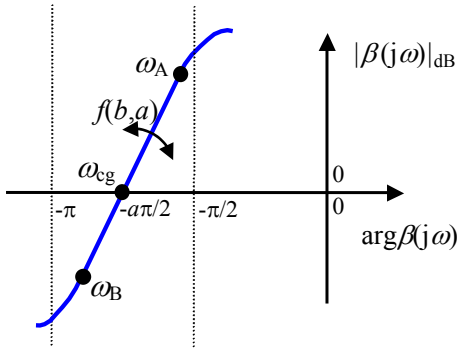


Figure 5: Generalized template in the Nichols plane.

The transfer function including complex fractional order integration is:

$$\beta(s)=\left(\cosh\left(b\frac{\pi}{2}\right)\right)^{\text{sign}(b)}\left(\frac{\omega_{cg}}{s}\right)^{a}\left(\text{Re}_{/i}\left(\left(\frac{\omega_{cg}}{s}\right)^{ib}\right)\right)^{-\text{sign}(b)} \quad (23)$$

with $n = a + ib \in \mathbb{C}_i$ and $\omega \in \mathbb{C}_j$, and where $\mathbb{C}_i$ and $\mathbb{C}_j$ are respectively time-domain and frequency-domain complex planes. In (Hartley and Lorenzo, 2005) a physical interpretation of such a complex order operator is proposed.

The definition of the open-loop transfer function including the nominal plant must take into account:

- accuracy specifications at low frequencies;
- the generalized template around frequency ω_{cg};
- plant behaviour at high frequencies while respecting the control effort specifications at these frequencies.

Thus, the open-loop transfer function is defined by a transfer function using band-limited complex fractional order integration:

$$\beta(s)=\beta_l(s)\bar{\beta}(s)\beta_h(s), \quad (24)$$

with:

$$\bar{\beta}(s)=C^{\text{sign}(b)}\left(\alpha\frac{1+s/\omega_h}{1+s/\omega_l}\right)^{a}\left(\Re e_{/i}\left\{\left(\alpha\frac{1+s/\omega_h}{1+s/\omega_l}\right)^{ib}\right\}\right)^{-q\,\text{sign}(b)} \quad (25)$$

$$\alpha_0=\left(1+\left(\frac{\omega_r}{\omega_0}\right)^2\Big/1+\left(\frac{\omega_r}{\omega_1}\right)^2\right)^{1/2} \quad (26)$$

where $\beta_l(s)$ is an integer order n_l proportional integrator:

$$\beta_l(s)=C_l\left(\frac{\omega_l}{s}+1\right)^{n_l} \quad (27)$$

- where $\beta_h(s)$ is a low-pass filter of integer order n_h:

$$\beta_h(s)=\frac{C_h}{\left(\frac{s}{\omega_h}+1\right)^{n_h}} \quad (28)$$

with

$$C_l=\left(\frac{\omega_{cg}{}^2}{\omega_l{}^2+\omega_{cg}{}^2}\right)^{n_l/2} \text{ and } C_h=\left(\frac{\omega_{cg}{}^2}{\omega_h{}^2}+1\right)^{n_h/2} \quad (29)$$

The optimal open loop transfer function is obtained by the minimization of the robustness cost function

$$J=\sup_{\omega,P}\left|T(j\omega)\right|-M_{r0}\ , \quad (30)$$

where M_{r0} is the resonant peak set for the nominal parametric state of the plant, while respecting the following set of inequality constraints for all plants (or parametric states of the plant) and for $\omega \in \mathbb{R}^+$:

$$\inf_P\left|T(j\omega)\right|\geq T_l(\omega) \text{ and } \sup_P\left|T(j\omega)\right|\leq T_u(\omega)\ , \quad (31)$$

$$\sup_P\left|S(j\omega)\right|\leq S_u(\omega)\ ,\ \sup_P\left|CS(j\omega)\right|\leq CS_u(\omega)$$

and

$$\sup_P\left|PS(j\omega)\right|\leq PS_u(\omega)\ , \quad (32)$$

with

$$\begin{cases} T(s)=\dfrac{C(s)P(s)}{1+C(s)P(s)} & S(s)=\dfrac{1}{1+C(s)P(s)} \\ CS(s)=\dfrac{C(s)}{1+C(s)P(s)} & PS(s)=\dfrac{G(s)}{1+C(s)P(s)} \end{cases} \quad (33)$$

As the uncertainties are taken into account by the least conservative method, a non-linear optimization method must be used to find the optimal values of three independent parameters. The parameterization of the open-loop transfer function by complex fractional orders, then simplifies the optimization considerably. During optimization a complex order has the same function as a whole set of parameters found in common rational controllers.

When the optimal nominal open-loop transfer is determined, the fractional controller $K_F(s)$ is defined by its frequency response:

$$K_F(j\omega)=\frac{\beta(j\omega)}{P_0(j\omega)}, \quad (34)$$

where $P_0(j\omega)$ is the nominal frequency response of the plant.

The parameters of a rational transfer function $K_R(s)$ with a predefined low-order structure are tuned to fit the ideal frequency response $K_F(j\omega)$. The rational integer model on which the parametric estimation is based, is given by:

$$K_{\mathrm{R}}(s)=\frac{B(s)}{A(s)}\ , \tag{35}$$

where $B(s)$ and $A(s)$ are polynomials of specified integer degrees n_B and n_A. Any frequency-domain system-identification technique can be used. An advantage of this design method is that whatever the complexity of the control problem, satisfactorily low values of n_B and n_A, usually around 6, can be used without performance reduction.

4 CRONE OBSERVER

Robustness considerations versus plant perturbation are also addressed in (Marquez, 2003) in an H_∞ framework for the synthesis of an dynamic output feedback based observer. In this paper, robustness to plant perturbation is taken into account with CRONE Control, thus leading to a new formulation of in the CRONE control-system design methodology.

4.1 Plant Perturbations and Disturbance Rejection Effects

It is now supposed that the plant whose state is estimated is submitted to perturbations. Effects of these perturbations but also effects of output disturbances $d_y(t)$ and measurement noises $n(t)$ on the estimation error are now studied. Control diagram of Fig. 6 is considered.
Using the notations previously introduced for the plant P, the model M and the controller K, the following state space description are now manipulated:

$$P:\begin{cases}\dot{x}(t)=(A+\Delta_A)x(t)+(B+\Delta_B)(u(t)+d_u(t))\\ y(t)=Cx(t)+d_y(t)+n(t)\end{cases} \tag{36}$$

$$M:\begin{cases}\dot{\hat{x}}(t)=A\hat{x}(t)+Bv(t)=A\hat{x}(t)+B(u(t)+w(t))\\ \hat{y}(t)=C\hat{x}(t)\end{cases} \tag{37}$$

$$K:\begin{cases}\dot{x}_K(t)=A_k x_K(t)-B_K\varepsilon(t)\\ \quad\ =A_K x(t)+B_K\left(Cx(t)+d_y(t)+n(t)\right)-B_K C\hat{x}(t)\\ w(t)=C_K x_K(t)\end{cases} \tag{38}$$

Δ_A and Δ_B are real matrices of appropriate dimensions that models plant perturbations.
At time t = 0, it is supposed that $x_K(0)=0$, $x(0)=x_0$, $\hat{x}(0)=0$ and thus $\chi(0)=x_0=\chi_0$. Laplace transform applied to relations (36) to (38) thus lead to:

$$P:\begin{cases}[sI-(A+\Delta_A)]x(s)=\chi_0+(B+\Delta_B)u(s)\\ y(s)=Cx(s)+d_y(s)+n(s)\end{cases} \tag{39}$$

$$M:\begin{cases}[sI-A]\hat{x}(s)=B(u(s)+w(s))\\ \hat{y}(s)=C\hat{x}(s)\end{cases} \tag{40}$$

and

$$K:\begin{cases}x_K(s)=[sI-A_K]^{-1}\begin{pmatrix}B_K\left(Cx(s)+d_y(s)+n(s)\right)\\ -B_K C\hat{x}(s)\end{pmatrix}\\ w(s)=C_K x_K(s)\end{cases}. \tag{41}$$

At time t = 0, it is supposed that $x_K(0)=0$, $x(0)=x_0$, $\hat{x}(0)=0$ and thus $\chi(0)=x_0=\chi_0$. Laplace transform applied to relations (36) to (38) thus lead to:

$$P:\begin{cases}[sI-(A+\Delta_A)]x(s)=\chi_0+(B+\Delta_B)u(s)\\ y(s)=Cx(s)+d_y(s)+n(s)\end{cases} \tag{39}$$

$$M:\begin{cases}[sI-A]\hat{x}(s)=B(u(s)+w(s))\\ \hat{y}(s)=C\hat{x}(s)\end{cases} \tag{40}$$

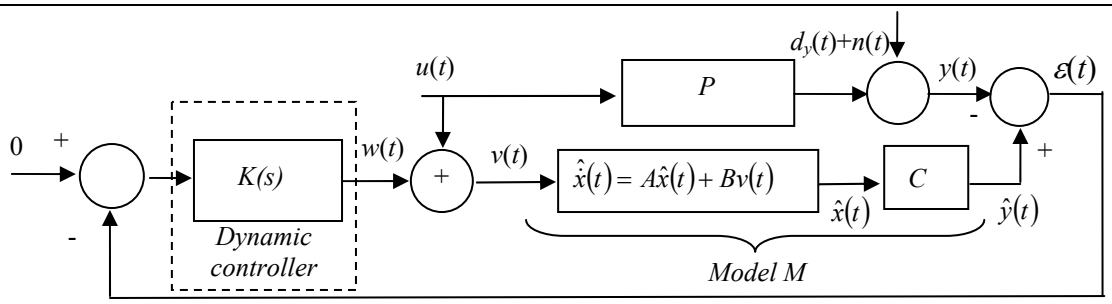


Figure 6: Dynamic output feedback based observer.

$$K:\begin{cases} x_K(s) = [sI - A_K]^{-1}\begin{pmatrix} B_K\left(Cx(s) + d_y(s) + n(s)\right) \\ -B_K C\hat{x}(s) \end{pmatrix} \\ w(s) = C_K x_K(s) \end{cases}. \quad (41)$$

Difference of state equations of representations (39) and (40) gives:

$$\begin{aligned} &[sI - (A + \Delta_A)]x(s) - [sI - A]\hat{x}(s) \\ &\quad = \chi_0 + (B + \Delta_B)u(s) - B(u(s) + w(s)) \end{aligned} \quad (42)$$

and thus using output equation of representation (41):

$$\begin{aligned} &[sI - (A + \Delta_A)]x(s) - [sI - A]\hat{x}(s) \\ &\quad = \chi_0 + (B + \Delta_B)u(s) - B(u(s) + C_K x_K(s)) \end{aligned} \quad (43)$$

Let $K(s)$ denotes the transfer function of the controller K, with:

$$K(s) = C_K [sI - A_K]^{-1} B_K, \quad (44)$$

Then relation (43) becomes:

$$\begin{aligned} &[sI - (A + \Delta_A)]x(s) - [sI - A]\hat{x}(s) \\ &\quad = \chi_0 + (B + \Delta_B)u(s) - Bu(s) \\ &\qquad - BC_K [sI - A_K]^{-1} B_K C(x(s) - \hat{x}(s)) \\ &\qquad - BC_K [sI - A_K]^{-1} B_K \left(d_y(s) + n(s)\right) \end{aligned} \quad (45)$$

and thus

$$\begin{aligned} &[sI - (A + \Delta_A)]x(s) - [sI - A]\hat{x}(s) \\ &\quad = \chi_0 + (B + \Delta_B)u(s) - Bu(s) \\ &\qquad - BK(s)C(x(s) - \hat{x}(s)) - BK(s)\left(d_y(s) + n(s)\right) \end{aligned} \quad .(46)$$

Laplace transform of the observation error is thus given by:

$$\begin{aligned} \chi(s) = &[sI - A + BK(s)C]^{-1}\chi_0 \\ &+ [sI - A + BK(s)C]^{-1}\Delta_A x(s) \\ &+ [sI - A + BK(s)C]^{-1}\Delta_B u(s) \\ &- [sI - A + BK(s)C]^{-1} BK(s)\left(d_y(s) + n(s)\right) \end{aligned} \quad (47)$$

4.2 Crone Observer Synthesis

Relation (47) demonstrates that without disturbances and plant perturbations ($\Delta_A = 0$, $\Delta_B = 0$, $d_y(s) + n(s) = 0$) observation error converges exponentially to 0 if the roots of the determinant of transfer matrix $[sI - A + BK(s)C]^{-1}$ lie in the left half complex plane, or equivalently given comments before theorem 1, if the closed loop in Fig. 2 is internally stable.

Moreover, relation (47) demonstrates that with disturbances and plant perturbations, observation errors can be reduced by finding a controller $K(s)$ that minimizes the modulus of the elements of the transfer matrix $[sI - A + BK(s)C]^{-1}\Delta_A$ and vectors $[sI - A + BK(s)C]^{-1}\Delta_B$ and $[sI - A + BK(s)C]^{-1} BK(s)$. Also notes that final value theorem can be applied on the elements on the previous matrix and vectors, to analyse the effects of plant perturbation and disturbances on observation error.

CRONE observer synthesis thus consist in finding an optimal open loop behaviour defined by transmittance (25) that minimises the maximal gain of matrix $[j\omega I - A + BK(j\omega)C]^{-1}\Delta_A$ and vectors $[j\omega I - A + BK(j\omega)C]^{-1}\Delta_B$ and $[j\omega I - A + BK(j\omega)C]^{-1} BK(j\omega)$ as ω varies within the frequency range $]0,..,\infty[$.

An algorithm for the CRONE observer synthesis can thus be summarized as follows:

- choice of an open-loop gain-crossover frequency ω_{cg} that ensures a satisfactory observation error cancellation dynamics;
- choice of orders n_l and n_h in order to ensure that the gain of the elements of matrix $[j\omega I - A + BK(j\omega)C]^{-1}\Delta_A$ and vectors $[j\omega I - A + BK(j\omega)C]^{-1}\Delta_B$ and $[j\omega I - A + BK(j\omega)C]^{-1} BK(j\omega)$ tends towards 0 as ω tends towards 0 and infinity to ensure a cancellation of observation error in steady stage and an immunity of this error to measurement noise;
- optimisation of parameters of open loop transmittance (25) through the minimisation of the criterion

$$J = \left\| \begin{bmatrix} F_1(j\omega) & F_2(j\omega) & F_3(j\omega) \end{bmatrix} \right\|_\infty, \quad (48)$$

with

$$F_1(j\omega) = W_A(\omega)[j\omega I - A + BK(j\omega)C]^{-1}\Delta_A$$

$$F_2(j\omega) = W_B(\omega)[j\omega I - A + BK(j\omega)C]^{-1}\Delta_B$$

$$F_3(j\omega) = W_C(\omega)[j\omega I - A + BK(j\omega)C]^{-1} BK(j\omega),$$

where $W_A(\omega)$, $W_B(\omega)$ and $W_C(\omega)$ denotes weighting matrices;

- synthesis of the controller $K(s)$ using the procedure described at the end of section 3.3 (relations (34) and (35)).

5 CONCLUSION

The main contribution of this paper is the development of a dynamic output feedback based observer that will be referred to as a CRONE observer in future developments. This name results in the introduction of CRONE controller in a feedback loop whose goal is to cancel the error between a model state and the unmeasured state of a plant that must be estimated. State observation with a dynamic output feedback based observer is concept that was developed in two papers (Marquez, 2003) and (Marquez and Riaz, 2005). Such an approach of state observation permits:

- a generalisation of the Luenberger form (Luenberger, 1971) that thus allows more freedom and flexibility in the design,
- a formulation allowing a more transparent view of the observer properties in term of feedback elements
- to poses the disturbances rejection problem and the observation robustness problem in the context of robust control theory.

The main differences between this paper and (Marquez, 2003) and (Marquez and Riaz, 2005) are :

- the extension of the dynamic output feedback based observer idea to the observation problem with unknown input,
- the uses of a CRONE controller to solve the disturbances rejection problem and the observation robustness (robustness of the observation error convergence to zero).

With the CRONE controller, plant model perturbations are taken into account in a structured form with no overestimation (but unmodelled dynamics can also be taken into account). Thus, without conservatism introduced in the plant uncertainties modelling, and in spite of a global optimization proof lack of the non convex optimisation problem defined in CRONE control, it turn out that in practice a CRONE controller permits to obtain better performance than an H_∞ one on the same plants (see for instance (Landau, et al, 1995) for a comparison on a benchmark based on robust digital control of a flexible transmission system). Due to the introduction of fractional differentiation, a parameterization of the open loop transfer function with a small number of parameters (three just like a PID controller) is obtained. The optimisation of the control law is thus reduced to the search for the optimal values of these parameters.

REFERENCES

Biorci G. and S. Ridella (1970). Ladder RC network with constant RC product - *IEEE Trans. Circuit Theory*, 17.

Dutta Roy S. C. (1970). On the realization of a constant-argument impedance of fractional operation - *IEEE Trans. Circuit Theory*, 17.

Hartley T. and C. Lorenzo (2005). Conjugated-order Differ Integrals, ASME Conference, Long Beach, California, 2005

Ichise M., Y. Nagayanagi and T. Kojima (1971). An analog simulation of non integer order transfer functions for analysis of electrode processes - *J. Electroanal. Chem. Interfacial Electrochem.*, 33, 253.

Landau I.D., Rey D., Karimi A., Voda A. and Franco A., (1995). *A Flexible Transmission System as a Benchmark for Robust Digital Control*, European Journal of Control, Vol. 1, pp. 77-96.

Lanusse P., A. Oustaloup and B. Mathieu (2000). Robust control of LTI square MIMO plants using two CRONE control design approaches - *IFAC Symposium on Robust Control Design "ROCOND 2000"*, Prague, Czech Republic, June 21-23.

Luenberger D. G. (1971). An introduction to observers. IEEE transactions on Automatic control, AC-16, 596-602.

Marquez H. J. (2003). A frequency domain approach to state estimation, Journal of the Franklin Institute, vol. 340, pp 147-157.

Marquez H. J., M. Riaz (2005). Robust state observer design with application to an industrial boiler system, Control Engineering Practice, n° 13, pp 713-728.

Miller K. S. and B. Ross (1993). An introduction to the fractional calculus and fractional differential equations - *John Wiley & Sons Inc.*, New York.

Oldham K. B. (1973). Semiintegral electroanalysis: analog implementation , *Anal. Chem.*, Vol. 45, p 39.

Oldham K. B. and J. Spanier (1974). The fractional calculus, *Academic Press*, New York.

Oustaloup A. (1975). Etude et réalisation d'un système d'asservissement d'ordre 3/2 de la fréquence d'un laser à colorant continu - *PhD Thesis*, Bordeaux I University, France.

Oustaloup A., A. Ballouk, P. Melchior, P. Lanusse and A. Elyagoubi (1990). Un nouveau régulateur CRONE fondé sur la dérivation non entiere complexe - *GR Automatique CNRS Meeting*, Bordeaux, France, March 29-30.

Oustaloup A. (1991). The CRONE control, *ECC'91* - Grenoble, France, July 2-5.

Oustaloup A., B. Mathieu and P. Lanusse (1995). The CRONE control of resonant plants: application to a flexible transmission - *European Journal of Control*, Vol. 1 n°2.

Oustaloup A., F. Levron, F. Nanot and B. Mathieu (2000). Frequency-band complex non integer differentiator: characterization and synthesis, *IEEE Transactions on Circuits and Systems,* Vol 47, n° 1, pp 25-40.

Pommier V., J. Sabatier, P. Lanusse, A. Oustaloup (2002). CRONE control of a nonlinear Hydraulic Actuator -

Control Engineering Practice, Vol. 10, n°4, pp. 391-402

Sabatier J., Poullain S., Latteux P., Thomas J. L., Oustaloup A. (2004), *Robust speed control of a low damped electromechanical system: application to a four mass experimental test bench*, International Journal of Nonlinear Dynamics and Chaos in Engineering Systems, Vol. 38, n° 1-4, pp 383-400.

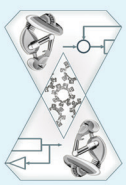
Samko S. G., A. A. Kilbas and O. I. Marichev (1993). Fractional integrals and derivatives - *Gordon and Breach Science Publishers*.

Suezaki and Takahashi (1966). Phase splitter of symmetric lattice network type and the termination with the impedance $1/\sqrt{S}$ - *Paper of the technical group on network and system theory, IECE*, Japan, 1966.

AUTHOR INDEX

AUTHOR INDEX (CONT.)

AUTHOR INDEX (CONT.)



Proceedings of ICINCO 2007
Fourth International Conference on
Informatics in Control, Automation and Robotics
ISBN: 978-972-8865-84-9
http://www.icinco.org