

Wojciech Mitkowski
Janusz Kacprzyk
Krzysztof Oprzędkiewicz
Paweł Skruch *Editors*

Trends in Advanced Intelligent Control, Optimization and Automation

Proceedings of KKA 2017—The 19th
Polish Control Conference, Kraków,
Poland, June 18–21, 2017

Advances in Intelligent Systems and Computing

Volume 577

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series “Advances in Intelligent Systems and Computing” contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within “Advances in Intelligent Systems and Computing” are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Advisory Board

Chairman

Nikhil R. Pal, Indian Statistical Institute, Kolkata, India

e-mail: nikhil@isical.ac.in

Members

Rafael Bello Perez, Universidad Central “Marta Abreu” de Las Villas, Santa Clara, Cuba

e-mail: rbellop@uclv.edu.cu

Emilio S. Corchado, University of Salamanca, Salamanca, Spain

e-mail: escorchado@usal.es

Hani Hagrass, University of Essex, Colchester, UK

e-mail: hani@essex.ac.uk

László T. Kóczy, Széchenyi István University, Győr, Hungary

e-mail: koczy@sze.hu

Vladik Kreinovich, University of Texas at El Paso, El Paso, USA

e-mail: vladik@utep.edu

Chin-Teng Lin, National Chiao Tung University, Hsinchu, Taiwan

e-mail: ctlin@mail.nctu.edu.tw

Jie Lu, University of Technology, Sydney, Australia

e-mail: Jie.Lu@uts.edu.au

Patricia Melin, Tijuana Institute of Technology, Tijuana, Mexico

e-mail: epmelin@hafsamx.org

Nadia Nedjah, State University of Rio de Janeiro, Rio de Janeiro, Brazil

e-mail: nadia@eng.uerj.br

Ngoc Thanh Nguyen, Wrocław University of Technology, Wrocław, Poland

e-mail: Ngoc-Thanh.Nguyen@pwr.edu.pl

Jun Wang, The Chinese University of Hong Kong, Shatin, Hong Kong

e-mail: jwang@mae.cuhk.edu.hk

More information about this series at <http://www.springer.com/series/11156>

Wojciech Mitkowski · Janusz Kacprzyk
Krzysztof Oprzędkiewicz · Paweł Skruch
Editors

Trends in Advanced Intelligent Control, Optimization and Automation

Proceedings of KKA 2017—The 19th
Polish Control Conference, Kraków,
Poland, June 18–21, 2017

 Springer

Editors

Wojciech Mitkowski
Department of Automatics and Biomedical
Engineering, Faculty of Electrical
Engineering, Automatics, Computer
Science and Biomedical Engineering
AGH University of Science and Technology
Kraków
Poland

Krzysztof Oprzędkiewicz
Department of Automatics and Biomedical
Engineering, Faculty of Electrical
Engineering, Automatics, Computer
Science and Biomedical Engineering
AGH University of Science and Technology
Kraków
Poland

Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
Warszawa
Poland

Paweł Skruch
Department of Automatics and Biomedical
Engineering, Faculty of Electrical
Engineering, Automatics, Computer
Science and Biomedical Engineering
AGH University of Science and Technology
Kraków
Poland

ISSN 2194-5357

ISSN 2194-5365 (electronic)

Advances in Intelligent Systems and Computing

ISBN 978-3-319-60698-9

ISBN 978-3-319-60699-6 (eBook)

DOI 10.1007/978-3-319-60699-6

Library of Congress Control Number: 2017943246

© Springer International Publishing AG 2017

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature

The registered company is Springer International Publishing AG

The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The book constitutes the proceedings of KKA 2017—The 19th Polish Control Conference (Krajowa Konferencja Automatyki, in Polish) organized by the Department of Automatics and Biomedical Engineering, Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering, AGH University of Science and Technology in Kraków, Poland, on June 18–21, 2017, under the auspices of the Committee on Automatic Control and Robotics of the Polish Academy of Sciences, and the Commission for Engineering Sciences of the Polish Academy of Arts and Sciences.

The KKA is a triennial conference with a very long tradition spanning over a couple of decades which has always gathered the Polish control theory, automatic control, industrial automation, robotics, and other related communities. At all the KKA conferences, and this concerns KKA 2017 too, there have always participated many prominent control theorists and practitioners, not only from the neighboring Eastern and Central European countries but also from Western Europe, the USA, and many other countries from all over the world. The tradition of inviting well known foreign researchers and scholars has existed since the very beginning, and the present KKA 2017 is no exception to this important rule. We have also managed to gather the best people from the community who have submitted great contributions. An exceptionally wide participation of the young generation of control theoreticians and practitioners has been noteworthy.

The present volume includes a collection of selected papers which have been accepted after a careful peer review process to continue maintaining the high quality standards that have always been synonymous with the KKA conferences since the first edition. The volume is divided into 12 parts intended to cover the main topics of the Conference, both theoretical and practical. Such large number of parts is clearly a result of the fact that automatic control, control theory, automation, robotics and related topics are considered to be relevant to many areas of science and technology.

The introductory Part I deals with more general and foundational issue and describes a number of control methods and algorithms. It starts with the plenary talk of Irena Lasiecka. She presents how to eliminate flutter in flow structure

interactions. Other particular problems discussed here can be summarized as follows. Mateusz Pietrala, Marek Jaskuła, Piotr Leśniewski and Andrzej Bartoszewicz discuss the sliding mode control of discrete time dynamical systems with state constraints. Leszek Trybus and Zbigniew Świder propose a novel method for the root-locus design of a PID controller for an unstable plant. Paweł Dworak, Michał Brasel and Sandip Ghosh give a comparison of different dynamic decoupling methods for a nonlinear MIMO plant. Piotr Tatjewski proposes a new approach to the offset-free nonlinear model predictive control. Maciej Ciężkowski explains and clarifies some issues related to the problem of damping of the pendulum during a dynamic stabilization in arbitrary angle position. Piotr Bania gives a simple example of a dual control problem with almost analytical a solution. Jakub Mozaryn, Andrzej Jezierski and Damian Suski make a comparison of the LQR and MPC control algorithms of an inverted pendulum. And, finally, Agata Cellmer, Bartosz Banach and Robert Piotrowski present a new approach to the design of modified PID controllers for the 3D crane control.

Part II is intended to present recent results in optimization, estimation and prediction which are relevant from the control point of view. Notably, the following problems are covered. Józef Duda considers an approach based on the Lyapunov matrices for the parametric optimization of a time delay system with the PID controller. Adam Kowalewski, Zbigniew Emirsajłow, Jan Sokołowski and Anna Krakowiak present issues and solutions for the sensitivity analysis of optimal control parabolic systems with retardations. Wojciech Rafajłowicz elaborates upon the optimality conditions for control problems described by integral equations. Kamil Borawski discusses the state vector estimation in descriptor electrical circuits using the shuffle algorithm. Paweł Domański and Piotr Marusak present some new results on the estimation of control improvement benefit with the α -stable distribution and, finally, Andrzej Tutaj and Wojciech Grega consider problems of packet buffering, dead time identification and state prediction for the control quality improvement in a networked control system.

Part III deals with an issue that has attracted much interest in the recent time, namely the autonomous vehicles. It starts with the plenary talk of Paweł Skruch who discusses control systems in a semi and fully automated car. Then, Paweł Skruch, Marek Długosz and Wojciech Mitkowski discuss issues in the stability analysis of a series of cars driving in an adaptive cruise control mode. Paweł Skruch, Marek Długosz, Paweł Markiewicz and Michał Szulc propose a formal approach to the verification of control systems in autonomous driving applications. Krzysztof Kogut, Krzysztof Kołek, Maciej Rosół and Andrzej Turnau develop a new current based slip controller for the ABS. Zdzisław Kowalczyk and Sylwester Frączek present some relevant problem from the railroad engineering, namely a system for tracking multiple trains on a test railway track. Jerzy Kasprzyk, Piotr Krauze and Janusz Wyrwał discuss the clipped LQ control oriented on driving safety of a half-car model with magnetorheological dampers and, finally, Paweł Markiewicz, Marek Długosz and Paweł Skruch present a review of tracking and object detection systems for advanced driver assistance and autonomous driving applications with a focus on the sensing of vulnerable road users.

Part IV is concerned with a very wide spectrum of applications. It covers both hardware and software of digital control systems, embedded systems, image processing, industrial networks, just to mention a few. In particular, the following problems are discussed. Paweł Sokólski, Tomasz A. Rutkowski and Kazimierz Duzinkiewicz consider the QDMC model predictive controller for a steam turbine in a nuclear power plant. Krzysztof Łakomy and Maciej Michałek compare the feedforward control design methods for nonminimum-phase LTI SISO systems with an application to the double-drum coiling machine. Ewaryst Rafajłowicz and Wojciech Rafajłowicz discuss the use of cameras in the control loop, from the point of view of problems, methods and selected industrial applications. Przemysław Szewczyk describes an approach to the real-time control of an active stereo vision system. Paweł Majdzik and Ralf Stetter discuss a receding-horizon approach to the state estimation of a battery assembly system. Stanisław Wrona, Krzysztof Mazur and Marek Pawełczyk present a problem of defining the optimal number of actuators for active device noise reduction applications.

Part V presents various issues related to computer methods in control engineering. Dariusz Rzońca, Jan Sadolewski, Andrzej Stec, Zbigniew Świder, Bartosz Trybus and Leszek Trybus consider the use of the CPDev engineering environment for control programming. Patryk Chaber and Maciej Ławryńczuk discuss the automatic code generation of the MIMO model predictive control algorithms using Transcompiler. Paweł Rotter and Maciej Klemiato consider a prototype vision-based system for the supervision of a glass melting process. Andrzej Wojtulewicz considers the implementation of a dynamic matrix control algorithm using the field programmable gate array. Sebastian Plamowski discusses some problems concerning the implementation of the DMC algorithm in an embedded controller with an emphasis on resources, memory and numerical modifications. Dymitr Juszczyk, Jarosław Tarnawski, Tomasz Karla and Kazimierz Duzinkiewicz discuss some relevant problems in the real-time simulation of a nuclear reactor using the client-server network architecture with a Web browser as the user interface. Marcin Kowalczyk and Tomasz Kryjak present a new approach to object tracking with the use of a moving camera implemented in the heterogeneous Zynq System on Chip. Mieczysław Wodecki, Wojciech Bożejko and Mariusz Uchroński analyze the k -opt algorithm in the flexible job shop scheduling environment. Patryk Chaber and Maciej Ławryńczuk propose an implementation of an analytical generalized predictive controller for very fast applications using microcontrollers. Finally, Marcin Jastrzębski and Jacek Kabziński discuss the robustness of the adaptive motion control against a fuzzy approximation of the LuGre multi-source friction model.

Part VI is dedicated to the use of the fractional order calculus in the modeling and control of dynamic systems which has attracted much interest in the scientific community. Tadeusz Kaczorek explains the relationship between the reachability of positive standard and fractional discrete-time and continuous-time linear systems. Ewa Pawłuszewicz considers the descriptor fractional-order systems with the l -memory and their stability in the Lyapunov sense. Krzysztof Oprzędkiewicz and Edyta Gawin present an approach to the modeling of a heat transfer process with the

use of non-integer order, discrete, transfer function models. Artur Babiarz and Adrian Łęgowski consider the fractional dynamics of the human arm. Stefan Domek deals with the approximation and stability analysis of some kind of switched fractional linear systems. Jerzy Klamka explains the relationship between the controllability of standard and fractional linear systems, and—finally—Wojciech Mitkowski provides some deep remarks and views about the stability of fractional systems.

A little bit shorter but equally interesting is Part VII which is focused on concepts from the area of advanced robotics. Tomasz Gawron and Maciej Marcin Michałek discuss the problem of the G^3 -continuous paths planning for state-constrained mobile robots with a bounded curvature of motion. Maciej Hojda considers the problem of task allocation for multi-robot teams in dynamic environments. Cezary Zielinski, Tomasz Winiarski and Tomasz Kornuta present agent-based structures of robot systems. Piotr Bazydło, Janusz Kacprzyk and Krzysztof Lasota propose the use of a novel evolutionary algorithm for the global path planning of a specialized autonomous robot for intrusion detection in the wireless sensor networks (WSNs). Anna Witkowska, Roman Śmierzchalski and Przemysław Wilczyński discuss the problem of trajectory planning for a service ship in the STS operation by using an evolutionary algorithm.

Part VIII deals with problems of modeling and identification. Zygmunt Hasiewicz, Paweł Wachel, Grzegorz Mzyk and Bartłomiej Kozdraś discuss the multistage identification of a Wiener-Hammerstein system. Witold Byrski proposes a new method for the identification of multi-inertial systems using the Strejc model. Jan Maciej Kościelny, Anna Szyber and Michał Syfert present a graph description of a process and its applications. Wojciech Kreft discusses the dynamics of a straw combustion process in a biomass boiler. Marcin Drzewiecki presents problems of modelling, simulation and optimization of the wavemaker in a towing tank. Ewa Skubalska-Rafajłowicz proposes a new method for the modeling of dynamic systems using neural networks and random linear projections. Mateusz Jabłoński explains the design of a process model of steam superheating in a power boiler and the adaptive control system for controlling this process. Finally, Kamil Czerwiński and Maciej Ławryńczuk consider the identification of a discrete model of an active magnetic levitation system.

Part IX is concerned with crucial problems of security, fault detection and diagnostics of devices and industrial processes. Marek Amanowicz and Jacek Jarmakiewicz propose a new approach to the decision support for cyber security in industrial control systems. Damian Kowalów and Maciej Patan discuss the distributed design of a sensor network for the detection of an abnormal state in distributed parameter systems. Karol Kulkowski, Michał Grochowski, Anna Kobylarz and Kazimierz Duzinkiewicz consider the application of data driven methods in the diagnostics of selected process faults of a steam turbine in a nuclear power plant. Łukasz Kuczkowski and Roman Śmierzchalski present a new algorithm for path planning for the ship collision avoidance in an environment with a changing strategy of dynamic obstacles. Marcin Pazera and Marcin Witczak consider the robust sensor fault-tolerant control for a non-linear aero-dynamical MIMO

system. Krzysztof Jaroszewski presents the workspace of an industrial manipulator in the case of a fault of a drive. Emilian Piesik and Marcin Śliwiński discuss the determination and verification of the safety integrity level with security aspects. Finally, Anna Bryniarska proposes a data granulation model for discovering knowledge about diagnosed objects.

Part X deals with relations between the automatic control and broadly meant intelligent systems, mainly fuzzy logic, neural networks, data mining, computer networks and the Internet of Things. It starts with the plenary lecture of Dmitry A. Novikov on Cybernetics 2.0, and its related modern challenges and perspectives. Wojciech Bożejko, Łukasz Gniewkowski and Mieczysław Wodecki discuss blocks for the flow shop scheduling problem with uncertain parameters. Tomasz Żabiński, Tomasz Mączka and Jacek Kluska propose an industrial platform for rapid prototyping of intelligent diagnostic systems. Zdzisław Kowalczyk, Marek Tatar and Adam Bąk present a novel evolutionary music composition system with statistically modeled criteria. Błażej Cichy, Petr Augusta, Krzysztof Gałkowski and Eric Rogers explain issues in the iterative learning control for a class of spatially interconnected systems. Bartłomiej Sulikowski, Krzysztof Gałkowski and Eric Rogers consider the iterative learning control of a class of spatially interconnected systems modeled in the form of two-dimensional (2D) systems. Marcin Boski, Wojciech Paszke and Eric Rogers present the learning filter design for the intelligent learning control schemes using the FIR approximation over a finite frequency range. Krzysztof Wiktorowicz shows an example of the adaptive fuzzy control design with the use of frequency-domain methods. Piotr Kulczycki and Damian Kruszewski consider the detection of atypical elements with fuzzy and intuitionistic fuzzy evaluations. Tomasz Talaśka, Rafał Długosz and Paweł Skruch propose a new efficient transistor level implementation of some selected fuzzy logic operators used in control systems. Finally, Marcin Jastrzębski and Jacek Kabziński discuss the robustness of the adaptive motion control against a fuzzy approximation of the LuGre multi-source friction model.

Part XI presents some applications of methods and models stemming from the automatic control that can be effectively and efficiently applied in biomedical engineering. Helmut Maurer and Andrzej Świerniak propose a method for the optimization of a combined anticancer treatment using models with multiple control delays. Jerzy Baranowski, Piotr Bania, Waldemar Bauer, Jędrzej Chiliński and Paweł Piątek discuss a hybrid Newton observer in the analysis of a glucose regulation system for the intensive care unit (ICU) patients. Agnieszka Mikołajczyk, Arkadiusz Kwasigroch and Michał Grochowski are concerned with an intelligent system for supporting the diagnosis of the malignant melanoma. Konrad Ciecierski and Tomasz Mandat consider the application of decision support systems in the functional neurosurgery. Arkadiusz Kwasigroch, Agnieszka Mikołajczyk and Michał Grochowski deal with the application of deep convolutional neural networks as a decision support tool in medical problems concentrating on the case of the malignant melanoma.

Part XII, the final one, deals with very relevant issues related to engineering education and teaching in the area of broadly perceived automatic control and

robotics. Teresa Zielińska discusses in her plenary talk the experience in the education of foreign students in a robotic program. Paweł Skruch, Marek Długosz and Wojciech Mitkowski propose how to improve the success rate of student software projects through developing some novel effort estimation practices.

We wish to express our deep gratitude to all the authors for their excellent contributions. Special thanks are due to anonymous peer referees whose deep analyses, and constructive remarks and suggestions have greatly helped improve the contributions. Waldemar Bauer, M.Sc. and Marek Długosz, Ph.D., deserve our thanks for their editorial help. We wish to fully acknowledge a constant and multifaceted help and support of the Committee on Automatic Control and Robotics of the Polish Academy of Sciences, and the Commission for Engineering Sciences of the Polish Academy of Arts and Sciences.

And last but not least, we wish to thank Dr. Tom Ditzinger, Dr. Leontina di Cecco and Mr. Holger Schaepe from the Engineering Editorial, SpringerNature for their dedication and help to implement and finish this large publication project on time maintaining the highest publication standards.

Warszawa, Poland
Kraków, Poland
Kraków, Poland
Kraków, Poland
March 2017

Janusz Kacprzyk
Wojciech Mitkowski
Krzysztof Oprzędkiewicz
Paweł Skruch

Contents

Part I Control Algorithms and Methods

How to eliminate flutter in flow structure interactions	3
Irena Lasiecka	
Sliding Mode Control of Discrete Time Dynamical Systems with State Constraints	4
Mateusz Pietrala, Marek Jaskuła, Piotr Leśniewski and Andrzej Bartoszewicz	
Root-locus Design of PID Controller for an Unstable Plant	14
Leszek Trybus and Zbigniew Świder	
A comparison of different dynamic decoupling methods for a nonlinear MIMO Plant	21
Paweł Dworak, Michał Brasel and Sandip Ghosh	
Offset-Free Nonlinear Model Predictive Control	33
Piotr Tatjewski	
Damping of the pendulum during dynamic stabilization in arbitrary angle position	45
Maciej Ciężkowski	
Simple example of dual control problem with almost analytical solution	55
Piotr Bania	
A Comparison of LQR and MPC Control Algorithms of an Inverted Pendulum	65
Andrzej Jezierski, Jakub Mozaryn and Damian Suski	
Design of modified PID controllers for 3D crane control	77
Agata Cellmer, Bartosz Banach and Robert Piotrowski	

Part II Optimization, Estimation and Prediction

Lyapunov Matrices Approach to the Parametric Optimization of Time Delay System with PID Controller.	89
Józef Duda	
Sensitivity Analysis of Optimal Control Parabolic Systems with Retardations	98
Adam Kowalewski, Zbigniew Emirsajłow, Jan Sokołowski and Anna Krakowiak	
Optimality conditions for optimal control problems modeled by integral equations.	108
Wojciech Rafajłowicz	
State vector estimation in descriptor electrical circuits using the shuffle algorithm	118
Kamil Borawski	
Estimation of control improvement benefit with α-stable distribution	128
Paweł D. Domański and Piotr M. Marusak	
Packet buffering, dead time identification, and state prediction for control quality improvement in a networked control system	138
Andrzej Tutaj and Wojciech Grega	

Part III Control Systems in Semi and Fully Automated Cars

Control Systems in Semi and Fully Automated Cars.	155
Paweł Skruch	
Stability Analysis of a Series of Cars Driving in Adaptive Cruise Control Mode	168
Paweł Skruch, Marek Długosz and Wojciech Mitkowski	
A Formal Approach for the Verification of Control Systems in Autonomous Driving Applications.	178
Paweł Skruch, Marek Długosz and Paweł Markiewicz	
A new current based slip controller for ABS	190
Krzysztof Kogut, Krzysztof Kołek, Maciej Rosół and Andrzej Turnau	
System for tracking multiple trains on a test railway track.	200
Zdzisław Kowalczyk and Sylwester Frączek	
The clipped LQ control oriented on driving safety of a half-car model with magnetorheological dampers.	214
Jerzy Kasprzyk, Piotr Krauze and Janusz Wyrwał	

Review of tracking and object detection systems for advanced driver assistance and autonomous driving applications with focus on vulnerable road users sensing 224
 Paweł Markiewicz, Marek Długosz and Paweł Skruch

Part IV Applications

The QDMC Model Predictive Controller for the Nuclear Power Plant Steam Turbine Control 241
 Paweł Sokólski, Tomasz A. Rutkowski and Kazimierz Duzinkiewicz

Comparison of the feedforward control design methods for nonminimum-phase LTI SISO systems with application to the double-drum coiling machine 251
 Krzysztof Łakomy and Maciej Marcin Michałek

Camera in the control loop – methods and selected industrial applications 261
 Ewaryst Rafajłowicz and Wojciech Rafajłowicz

Real-Time Control of Active Stereo Vision System. 271
 Przemysław Szewczyk

A receding-horizon approach to state estimation of the battery assembly system 281
 Paweł Majdzik and Ralf Stetter

Defining the Optimal Number of Actuators for Active Device Noise Reduction Applications. 291
 Stanisław Wrona, Krzysztof Mazur and Marek Pawełczyk

Part V Computer Methods in Control Engineering

CPDev engineering environment for control programming 303
 Dariusz Rzońca, Jan Sadolewski, Andrzej Stec, Zbigniew Świder, Bartosz Trybus and Leszek Trybus

Automatic Code Generation of MIMO Model Predictive Control Algorithms using Transcompiler 315
 Patryk Chaber and Maciej Ławryńczuk

Implementation of Dynamic Matrix Control Algorithm Using Field Programmable Gate Array: Preliminary Results 325
 Andrzej Wojtulewicz

Implementation of DMC algorithm in embedded controller - resources, memory and numerical modifications. 335
 Sebastian Plamowski

Real-Time Basic Principles Nuclear Reactor Simulator Based on Client-Server Network Architecture with WebBrowser as User Interface	344
Dymitr Juszczuk, Jarosław Tarnawski, Tomasz Karła and Kazimierz Duzinkiewicz	
Object Tracking With the Use of a Moving Camera Implemented in Heterogeneous Zynq System on Chip	353
Marcin Kowalczyk and Tomasz Kryjak	
Prototype vision-based system for the supervision of the glass melting process: implementation for industrial environment	363
Paweł Rotter and Maciej Klemiato	
The k-opt algorithm analysis. The flexible job shop case	369
Wojciech Bożejko, Mariusz Uchroński and Mieczysław Wodecki	
Implementation of Analytical Generalized Predictive Controller for Very Fast Applications Using Microcontrollers: Preliminary Results	377
Ptryk Chaber and Maciej Ławryńczuk	
Robustness of Adaptive Motion Control Against Fuzzy Approximation of LuGre Multi-Source Friction Model	387
Marcin Jastrzębski and Jacek Kabziński	
Part VI Non-Integer Order Calculus	
Relationships between the reachability of positive standard and fractional discrete-time and continuous-time linear systems	401
Tadeusz Kaczorek	
Remarks on descriptor fractional-order systems with l-memory and its stability in Lyapunov sense	415
Ewa Pawłuszewicz	
Modeling of heat transfer process with the use of non integer order, discrete, transfer function models	425
Krzysztof Oprzędkiewicz and Edyta Gawin	
Human arm fractional dynamics	434
Artur Babiarez and Adrian Łęgowski	
Approximation and stability analysis of some kinds of switched fractional linear systems	442
Stefan Domek	
Relationship between controllability of standard and fractional linear systems	455
Jerzy Klamka	

Remarks about stability of fractional systems 460
 Wojciech Mitkowski

Part VII Advanced Robotics

Planning \mathbb{G}^3 -continuous paths for state-constrained mobile robots with bounded curvature of motion 473
 Tomasz Gawron and Maciej M. Michałek

Task allocation for multi-robot teams in dynamic environments 483
 Maciej Hojda

Agent-Based Structures of Robot Systems 493
 Cezary Zieliński, Tomasz Winiarski and Tomasz Kornuta

Global path planning for a specialized autonomous robot for intrusion detection in wireless sensor networks (WSNs) using a new evolutionary algorithm 503
 Piotr Bazydło, Janusz Kacprzyk and Krzysztof Lasota

Trajectory planning for Service Ship during emergency STS transfer operation 514
 Anna Witkowska, Roman Śmierczalski and Przemysław Wilczyński

Part VIII Modeling and Identification

Multistage identification of Wiener-Hammerstein system 527
 Zygmunt Hasiewicz, Paweł Wachel, Grzegorz Mzyk and Bartłomiej Kozdraś

A new method of multi-inertial systems identification by the Strejc model 536
 Witold Byrski

Graph description of the process and its applications 550
 Jan Maciej Kościelny, Anna Szyber and Michał Syfert

The dynamics of the straw combustion process in the batch-fired straw boiler 560
 Wojciech Kreft

Modelling, Simulation and Optimization of the Wavemaker in a Towing Tank 570
 Marcin Drzewiecki

Modeling dynamical systems using neural networks and random linear projections 580
 Ewa Skubalska-Rafajłowicz

Designing the process model of steam superheating in a power boiler and the adaptive control system which controls this process.	590
Mateusz Jabłoński	
Identification of Discrete-Time Model of Active Magnetic Levitation System	599
Kamil Czerwiński and Maciej Ławryńczuk	
Part IX Fault Detection, Security and Diagnostics	
Cyber Security Provision for Industrial Control Systems	611
Marek Amanowicz and Jacek Jarmakiewicz	
Distributed design of sensor network for abnormal state detection in distributed parameter systems.	621
Damian Kowalów and Maciej Patan	
Application of data driven methods in diagnostic of selected process faults of nuclear power plant steam turbine	631
Karol Kulkowski, Michał Grochowski, Anna Kobylarz and Kazimierz Duzinkiewicz	
Path planning algorithm for ship collisions avoidance in environment with changing strategy of dynamic obstacles	641
Łukasz Kuczowski and Roman Śmierzchalski	
Robust sensor fault-tolerant control for non-linear aero-dynamical MIMO system.	651
Marcin Pazera and Marcin Witczak	
The workspace of industrial manipulator in case of fault of the drive	661
Krzysztof Jaroszewski	
Determining and verifying the safety integrity level with security aspects.	669
Emilian Piesik and Marcin Śliwiński	
A data granulation model for searching knowledge about diagnosed objects	681
Anna Bryniarska	
Part X Intelligent systems	
Cybernetics 2.0: Modern Challenges and Perspectives.	693
Dmitry A. Novikov	

Blocks for the flow shop scheduling problem with uncertain parameters	703
Wojciech Bożejko, Łukasz Gniewkowski and Mieczysław Wodecki	
Industrial Platform for Rapid Prototyping of Intelligent Diagnostic Systems	712
Tomasz Żabiński, Tomasz Mączka and Jacek Kluska	
Evolutionary music composition system with statistically modeled criteria	722
Zdzisław Kowalczyk, Marek Tataro and Adam Bąk	
Iterative Learning Control for a class of spatially interconnected systems	734
Błażej Cichy, Petr Augusta, Krzysztof Gałkowski and Eric Rogers	
Iterative Learning Control for a discretized sub-class of spatially interconnected systems	744
Bartłomiej Sulikowski, Krzysztof Gałkowski and Eric Rogers	
Learning filter design for ILC schemes using FIR approximation over a finite frequency range	754
Marcin Boski, Wojciech Paszke and Eric Rogers	
An example of adaptive fuzzy control design with the use of frequency-domain methods	764
Krzysztof Wiktorowicz	
Detection of atypical elements with fuzzy and intuitionistic evaluations	774
Piotr Kulczycki and Damian Kruszewski	
Efficient transistor level implementation of selected fuzzy logic operators used in control systems	787
Tomasz Talaśka, Rafał Długosz and Paweł Skruch	
Part XI Biomedical Applications of Control Engineering	
Optimization of Combined Anticancer Treatment Using Models With Multiple Control Delays	799
Helmut Maurer and Andrzej Świerniak	
Hybrid Newton Observer in Analysis of Glucose Regulation System for ICU Patients	818
Jerzy Baranowski, Piotr Bania, Waldemar Bauer, Jędrzej Chiliński and Paweł Piątek	
Intelligent system supporting diagnosis of malignant melanoma	828
Agnieszka Mikołajczyk, Arkadiusz Kwasigroch and Michał Grochowski	

Applications of decision support systems in functional neurosurgery 838
Konrad A. Ciecierski and Tomasz Mandat

Deep convolutional neural networks as a decision support tool in medical problems – malignant melanoma case study. 848
Arkadiusz Kwasigroch, Agnieszka Mikołajczyk and Michał Grochowski

Part XII Engineering education and teaching

Sharing the Experience in International Students Education: Robotics Program 859
Teresa Zielińska

Improving the Success Rate of Student Software Projects through Developing Effort Estimation Practices 869
Paweł Skruch, Marek Długosz and Wojciech Mitkowski

Author Index. 881

Part I
Control Algorithms and Methods

How to eliminate flutter in flow structure interactions

Irena Lasiecka

University of Memphis
Email: lasiecka@memphis.edu.pl

An appearance of flutter in oscillating structures is an endemic phenomenon. Most common causes are vibrations induced by the moving flow of a gas which is interacting with the structure. Typical examples include: turbulent jets, vibrating bridges, oscillating facial palate in the onset of apnea. In the case of an aircraft it may compromise its safety. The intensity of the flutter depends heavily on the speed of the flow (subsonic, transonic or supersonic regimes). Thus, reduction or attenuation of flutter is one of the key problems in aeroelasticity with application to a variety of fields including aerospace engineering, structural engineering, medicine and life sciences. Mathematical models describing this phenomenon involve coupled systems of partial differential equations (Euler Equation and nonlinear plate equation) with interaction at the interface - which is the boundary surface of the structure. The aim of this talk is to present a theory describing: (1) qualitative properties of the resulting dynamical systems (existence, uniqueness and robustness of finite energy solutions), (2) asymptotic stability and associated long time behavior that includes the study of global attractors, (3) feedback control strategies aiming at the elimination or attenuation of the flutter. As a consequence one concludes that the flow alone (without any dissipation added to the elastic structure) provides some stabilizing effect on the plate by reducing asymptotically its dynamics to a finite dimensional structure. However, the resulting "dynamical system" may be exhibiting a chaotic behavior. In the subsonic case, one also shows that the flutter can be eliminated by adding structural damping to the plate.

Sliding Mode Control of Discrete Time Dynamical Systems with State Constraints

Mateusz Pietrala, Marek Jaskuła, Piotr Leśniewski, Andrzej Bartoszewicz

Institute of Automatic Control, Lodź University of Technology,
18/22 Stefanowskiego St., 90-924 Lodź, Poland

Abstract. In this paper, we study the problem of state and control signal constraints in discrete-time sliding mode control. We introduce a sufficient condition for finite time convergence of the representative point to the sliding hyperplane, while respecting imposed restrictions. We propose a new control strategy based on the reaching law approach.

1 Introduction

Sliding mode control is one of the most effective regulation methods used for a wide range of uncertain systems. The major benefits of this technique are robustness and computational efficiency. Therefore, it became a popular area of research. At first, the continuous-time systems have been considered by Utkin [21] and Emelyanov [12] and then the discrete-time systems have been introduced [18], [22], [14]. The main idea of the sliding mode control is to drive the representative point (state vector) to the sliding hyperplane and maintain its evolution on it. In the first place the hyperplane should be designed in order to ensure the desired dynamic behavior of the system. Afterwards, the control signal is computed and implemented into the system. It can be obtained using the reaching law technique, which is applied in this paper. This method was first used for continuous-time systems by Gao and Hung in [15]. Subsequently, Gao, Wang and Homaifa [16] presented similar results for discrete-time systems. In the following years, other forms of reaching law were presented [1]-[6], [10], [11], [13], [17], [19], [20], [23], [24], [25]. The basic reaching law strategy [16] may cause large values of state variables or control signal. To solve this problem, in this paper, we introduce the new reaching law. It is designed in order to satisfy state and control signal constraints [7], [8], [9].

This paper is organized as follows. Section 2 presents the design of the sliding mode controller based on a simple reaching law. Both state and control signal constraints are analyzed in Sect. 3. Moreover, in the same section the new reaching law is introduced. The main result, i.e. monotonic convergence of the representative point to the sliding hyperplane in finite time, while satisfying given constraints, is obtained in Sect. 4. Section 5 comprises a simulation example, and Sect. 6 presents the conclusions of this paper.

2 Sliding Mode Controller Design

Let us consider a discrete-time system described by the following equation

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{b}u(k), \quad (1)$$

where $\mathbf{x}(k) = [x_1(k), \dots, x_n(k)]^T$ is the state vector, \mathbf{A} is the state matrix ($\dim(\mathbf{A}) = n \times n$), \mathbf{b} is the input distribution vector ($\dim(\mathbf{b}) = n \times 1$) and $u(k)$ is a scalar input. We design the discrete-time sliding mode controller in order to obtain the following properties of system (1):

1. The system representative point starting from any initial position $\mathbf{x}(0)$ converges monotonically to the sliding hyperplane

$$s(k) = \mathbf{c}^T \mathbf{x}(k) = 0, \quad (2)$$

where $\mathbf{c}^T = [c_1, \dots, c_{n-1}, 1]$. We select vector \mathbf{c} so that $\mathbf{c}^T \mathbf{b} \neq 0$.

2. The system representative point reaches the sliding hyperplane in finite time.

In this paper we define the quasi-sliding mode similarly as in [1], [2], i.e. we do not require the representative point to cross the sliding hyperplane in each consecutive step. Nevertheless, we demand that the above point remains in a neighborhood of the sliding hyperplane. Let us consider the following reaching law

$$s(k+1) = s(k) - K \operatorname{sgn}(s(k)), \quad (3)$$

where K is a real number and the function $\operatorname{sgn}(x)$ is given as follows

$$\operatorname{sgn}(x) = \begin{cases} 1, & \text{when } x > 0 \\ 0, & \text{when } x = 0 \\ -1, & \text{when } x < 0 \end{cases}. \quad (4)$$

Reaching law (3) guarantees satisfying the two properties specified above. Furthermore, if the representative point arrives precisely on the sliding hyperplane at the time k_0 , i.e. $s(k_0) = 0$, then for every $k \geq k_0$ we obtain $s(k+1) = 0 - K \operatorname{sgn}(0)$. We can observe, that when the representative point arrives precisely on the sliding hyperplane, it remains on it. Using (1), (2) and (3) we obtain the following form of the control signal

$$u(k) = (\mathbf{c}^T \mathbf{b})^{-1} [-\mathbf{c}^T \mathbf{A}\mathbf{x}(k) + \mathbf{c}^T \mathbf{x}(k) - K \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k))]. \quad (5)$$

3 State and Control Signal Constraints

In this section state constraints will be considered first, and then we will focus our attention on the problem of input signal limitation.

3.1 State Constraints

Control signal (5) does not guarantee that the state constraints are satisfied. Therefore, we modify parameter K . Our goal is to limit each state variable $x_i(k)$, $i \in \{1, \dots, n\}$ for any $k \in \mathbb{N}$. We assume that the absolute value of state variable $x_i(0)$ is limited by r_i for any $i \in \{1, \dots, n\}$. We will find parameter K so that if the absolute value of state variable $x_i(k)$ is limited by r_i , then the absolute value of state variable $x_i(k+1)$ is also limited by r_i , i.e.

$$-r_i \leq x_i(k+1) \leq r_i. \quad (6)$$

We use (5) to rewrite (1) as follows

$$\begin{aligned} \mathbf{x}(k+1) = & \mathbf{A}\mathbf{x}(k) - \mathbf{b}(\mathbf{c}^T\mathbf{b})^{-1}\mathbf{c}^T\mathbf{A}\mathbf{x}(k) \\ & + \mathbf{b}(\mathbf{c}^T\mathbf{b})^{-1}\mathbf{c}^T\mathbf{x}(k) - \mathbf{b}(\mathbf{c}^T\mathbf{b})^{-1}K\text{sgn}(\mathbf{c}^T\mathbf{x}(k)). \end{aligned} \quad (7)$$

In order to simplify the notation let

$$\mathbf{G} = \mathbf{A} + \mathbf{b}(\mathbf{c}^T\mathbf{b})^{-1}(-\mathbf{c}^T\mathbf{A} + \mathbf{c}^T). \quad (8)$$

Then (7) is of the following form

$$\mathbf{x}(k+1) = \mathbf{G}\mathbf{x}(k) - K(\mathbf{c}^T\mathbf{b})^{-1}\mathbf{b}\text{sgn}(\mathbf{c}^T\mathbf{x}(k)). \quad (9)$$

Denote by \mathbf{e}_i ($\dim(\mathbf{e}_i) = 1 \times n$) versor of the i -th axis of a Cartesian coordinate system, i.e. the i -th element of vector \mathbf{e}_i is equal to one, while the remaining elements of this vector are equal to zero. We use (9) to rewrite (6) as follows

$$-r_i \leq \mathbf{e}_i \left[\mathbf{G}\mathbf{x}(k) - K(\mathbf{c}^T\mathbf{b})^{-1}\mathbf{b}\text{sgn}(\mathbf{c}^T\mathbf{x}(k)) \right] \leq r_i, \quad (10)$$

where $i \in \{1, \dots, n\}$. Our goal is to determine the time-varying K , so that (10) is satisfied. We select the largest K to obtain the fastest convergence to the sliding hyperplane, while respecting the state constraints. Transforming (10) in order to determine K we get

$$\mathbf{g}_i\mathbf{x}(k) - r_i \leq K(\mathbf{c}^T\mathbf{b})^{-1}\mathbf{e}_i\mathbf{b}\text{sgn}(\mathbf{c}^T\mathbf{x}(k)) \leq \mathbf{g}_i\mathbf{x}(k) + r_i, \quad (11)$$

where $\mathbf{g}_i = \mathbf{e}_i\mathbf{G}$. Note that if $\mathbf{e}_i\mathbf{b} = 0$ for some $i \in \{1, \dots, n\}$, then K has no influence on the x_i state variable evolution. In this situation if

$$-r_i \leq \mathbf{g}_i\mathbf{x}(k) \leq r_i, \quad (12)$$

then (11) is satisfied for any value of K . Otherwise, the constraint of the x_i variable cannot be satisfied. From now on, we assume that $\mathbf{e}_i\mathbf{b} \neq 0$. Note that if $(\mathbf{c}^T\mathbf{b})^{-1}\mathbf{e}_i\mathbf{b}\text{sgn}(\mathbf{c}^T\mathbf{x}(k)) > 0$, then (11) is of the following form

$$\begin{cases} K \geq \mathbf{c}^T\mathbf{b}(\mathbf{e}_i\mathbf{b})^{-1}\text{sgn}(\mathbf{c}^T\mathbf{x}(k))(\mathbf{g}_i\mathbf{x}(k) - r_i) \\ K \leq \mathbf{c}^T\mathbf{b}(\mathbf{e}_i\mathbf{b})^{-1}\text{sgn}(\mathbf{c}^T\mathbf{x}(k))(\mathbf{g}_i\mathbf{x}(k) + r_i) \end{cases}. \quad (13)$$

Otherwise, if $(\mathbf{c}^T \mathbf{b})^{-1} \mathbf{e}_i \mathbf{b} \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) < 0$, then

$$\begin{cases} K \leq \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) (\mathbf{g}_i \mathbf{x}(k) - r_i) \\ K \geq \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) (\mathbf{g}_i \mathbf{x}(k) + r_i) \end{cases}. \quad (14)$$

When $\operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) = 0$, the parameter K does not affect the dynamics of the system. For each $i \in \{1, \dots, n\}$ the parameter K must satisfy (13) or (14). Let us denote by K_i the largest K for which the i -th state variable constraint is satisfied. Then we obtain

$$\begin{aligned} K_i &= \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) \mathbf{g}_i \mathbf{x}(k) \\ &\quad + \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) \operatorname{sgn}\left(\mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k))\right) r_i. \end{aligned} \quad (15)$$

Let us observe that K_i is equal to the upper limit specified by either (13) or (14). We can rewrite (15) in the alternative form

$$K_i = \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) \mathbf{g}_i \mathbf{x}(k) + \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| r_i. \quad (16)$$

Theorem 1. *Assume that $K_i > 0$ is defined by (16) and $i \in \{1, \dots, n\}$. Then the state variable x_i satisfies its constraint for any $K \in (0; K_i]$.*

Proof. Let

$$K_\varepsilon = \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) \mathbf{g}_i \mathbf{x}(k) + \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| (r_i - \varepsilon). \quad (17)$$

In order to obtain $K_\varepsilon > 0$ we select $\varepsilon \in (0; r_i)$. Let us observe that $K_\varepsilon < K_i$. From (9) and (17) we have

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{G} \mathbf{x}(k) - (\mathbf{e}_i \mathbf{b})^{-1} \mathbf{b} \mathbf{g}_i \mathbf{x}(k) \\ &\quad - \operatorname{sgn}(\mathbf{c}^T \mathbf{b}) \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) \left| (\mathbf{e}_i \mathbf{b})^{-1} \right| \mathbf{b} (r_i - \varepsilon). \end{aligned} \quad (18)$$

Multiplying (18) by \mathbf{e}_i we get

$$x_i(k+1) = -\operatorname{sgn}(\mathbf{c}^T \mathbf{b}) \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) \operatorname{sgn}(\mathbf{e}_i \mathbf{b}) (r_i - \varepsilon). \quad (19)$$

Noting that $-\operatorname{sgn}(\mathbf{c}^T \mathbf{b}) \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) \operatorname{sgn}(\mathbf{e}_i \mathbf{b})$ can take a value of $-1, 0$ or 1 we obtain

$$-r_i + \varepsilon \leq x_i(k+1) \leq r_i - \varepsilon. \quad (20)$$

Hence, using K_ε , the constraint of the state variable x_i is also satisfied. This ends the proof. \square

We want to satisfy all of the state constraints and select the value of K as large as possible. Therefore,

$$K = \min\{K_1, \dots, K_n\}. \quad (21)$$

3.2 Control Signal Constraint

Since in practice large values of the control signal are undesirable, further in this section we will impose an additional constraint on the parameter K . We assume that the absolute value of control signal $u(k)$ is limited by $r_u \in \mathbb{R}_+$, i.e.

$$-r_u \leq u(k) \leq r_u. \quad (22)$$

We use (5) to rewrite (22) as follows

$$-r_u \leq (\mathbf{c}^T \mathbf{b})^{-1} [-\mathbf{c}^T \mathbf{A} \mathbf{x}(k) + \mathbf{c}^T \mathbf{x}(k) - K \text{sgn}(\mathbf{c}^T \mathbf{x}(k))] \leq r_u. \quad (23)$$

In order to simplify the notation we introduce symbol

$$\mathbf{g}_u = -(\mathbf{c}^T \mathbf{b})^{-1} \mathbf{c}^T \mathbf{A} + (\mathbf{c}^T \mathbf{b})^{-1} \mathbf{c}^T. \quad (24)$$

Then (23) is of the following form

$$-r_u \leq \mathbf{g}_u \mathbf{x}(k) - (\mathbf{c}^T \mathbf{b})^{-1} K \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) \leq r_u. \quad (25)$$

Transforming (25) in order to determine K we get

$$\mathbf{g}_u \mathbf{x}(k) - r_u \leq (\mathbf{c}^T \mathbf{b})^{-1} K \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) \leq \mathbf{g}_u \mathbf{x}(k) + r_u. \quad (26)$$

Note that if $(\mathbf{c}^T \mathbf{b})^{-1} \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) > 0$, then (26) is of the form

$$\begin{cases} K \geq \mathbf{c}^T \mathbf{b} \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) (\mathbf{g}_u \mathbf{x}(k) - r_u) \\ K \leq \mathbf{c}^T \mathbf{b} \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) (\mathbf{g}_u \mathbf{x}(k) + r_u) \end{cases}. \quad (27)$$

Otherwise, if $(\mathbf{c}^T \mathbf{b})^{-1} \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) < 0$ then

$$\begin{cases} K \leq \mathbf{c}^T \mathbf{b} \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) (\mathbf{g}_u \mathbf{x}(k) - r_u) \\ K \geq \mathbf{c}^T \mathbf{b} \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) (\mathbf{g}_u \mathbf{x}(k) + r_u) \end{cases}. \quad (28)$$

Similarly as in the previous section if $\text{sgn}(\mathbf{c}^T \mathbf{x}(k)) = 0$, then the parameter K does not affect the dynamics of the system. Let us denote by K_u the largest K for which the control signal constraint is satisfied. Then

$$K_u = \mathbf{c}^T \mathbf{b} \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) \mathbf{g}_u \mathbf{x}(k) + |\mathbf{c}^T \mathbf{b}| r_u. \quad (29)$$

Theorem 2. *Assume that $K_u > 0$ is defined by (29). Then the control signal satisfies its constraint for any $K \in (0; K_u]$.*

Proof. Let

$$K_\delta = \mathbf{c}^T \mathbf{b} \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) \mathbf{g}_u \mathbf{x}(k) + |\mathbf{c}^T \mathbf{b}| (r_u - \delta). \quad (30)$$

In order to ensure $K_\delta > 0$ we select $\delta \in (0; r_u)$. Let us observe that $K_\delta < K_u$ and use (24) and (30) to rewrite (5) as follows

$$u(k) = -\text{sgn}(\mathbf{c}^T \mathbf{b}) \text{sgn}(\mathbf{c}^T \mathbf{x}(k)) (r_u - \delta). \quad (31)$$

Noting that $\text{sgn}(\mathbf{c}^T \mathbf{b}) \text{sgn}(\mathbf{c}^T \mathbf{x}(k))$ can take a value of -1, 0 or 1 we have

$$-r_u + \delta \leq u(k) \leq r_u - \delta. \quad (32)$$

Hence, using parameter K_δ , the constraint of the control signal is also satisfied. This ends the proof. \square

Let us observe that if $\delta = 0$, then $K_\delta = K_u$. In this case using K_u given by (29) we obtain the value of control signal equal to $-r_u, 0$ or r_u .

3.3 State and Control Signal Constraints

In this subsection the above considerations are both taken into account. We select K in order to satisfy state and control signal constraints simultaneously. For this purpose, if K_i is described by (16) and K_u by (29), then

$$K = \min\{K_1, \dots, K_n, K_u\}. \quad (33)$$

Selecting K according to (33) in each consecutive step, results in satisfying both state and control signal constraints.

3.4 Modified Reaching Law

Reaching law (3) does not guarantee that the representative point arrives precisely on the sliding hyperplane. In this case the band width of the quasi-sliding mode is equal to $2K$. This value varies with time, so we cannot determine its specific value. To eliminate the above problem we introduce the following reaching law

$$s(k+1) = s(k) - \min\{K(k), |s(k)|\} \text{sgn}(s(k)), \quad (34)$$

where $K(k)$ is defined as K in (33) in the k -th step.

4 Sufficient condition

We begin this section with formulating and proving the theorem which specifies the sufficient condition for $K(k) > 0$ for any $k \in \mathbb{N} \cup \{0\}$.

Theorem 3. *Denote by g_{ij} the expression in the i -th row and j -th column of matrix \mathbf{G} and by g_{ui} the i -th element of vector \mathbf{g}_u . In order to obtain $K(k) > 0$ in each consecutive step it is sufficient that inequalities*

$$|g_{i1}|r_1 + \dots + |g_{in}|r_n < r_i \quad (35)$$

$$|g_{u1}|r_1 + \dots + |g_{un}|r_n < r_u \quad (36)$$

are satisfied for any $i \in \{1, \dots, n\}$.

Proof. From (16) we observe that if $|\mathbf{g}_i \mathbf{x}(k)| < r_i$, then $K_i > 0$. Hence, our goal is to satisfy inequalities

$$|\mathbf{g}_i \mathbf{x}(k)| < r_i, \quad (37)$$

for any $i \in \{1, \dots, n\}$. Let us estimate the greatest possible value of the left-hand side of (37)

$$\begin{aligned} \max |\mathbf{g}_i \mathbf{x}(k)| &= |g_{i1} \operatorname{sgn}(g_{i1}) r_1 + \dots + g_{in} \operatorname{sgn}(g_{in}) r_n| \\ &= |g_{i1}| r_1 + \dots + |g_{in}| r_n. \end{aligned} \quad (38)$$

Using (37) and (38) we obtain that if (35) is true, then $K_i > 0$. Thus, we want to satisfy (35) for any $i \in \{1, \dots, n\}$. Similarly, our goal is to find the sufficient condition for $K_u > 0$. We obtain that if (36) is satisfied, then $K_u > 0$. Hence, (35) and (36) are sufficient conditions for $K_i > 0$ and $K_u > 0$. This ends the proof. \square

Further in this section, we formulate and prove the theorem showing that one can find $\varepsilon > 0$ such that $K(k) \geq \varepsilon > 0$ in each consecutive step. That results in a convergence of the representative point to the sliding hyperplane in finite time.

Theorem 4. *Assume that (35) and (36) are satisfied. Then $K(k) \geq \varepsilon > 0$ in each consecutive step. Parameter*

$$\varepsilon = \min \left\{ \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_1 \mathbf{b})^{-1} \right| \delta_1, \dots, \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_n \mathbf{b})^{-1} \right| \delta_n, \left| \mathbf{c}^T \mathbf{b} \right| \delta_u \right\}, \quad (39)$$

where

$$\delta_i = r_i - (|g_{i1}| r_1 + \dots + |g_{in}| r_n) \quad (40)$$

for $i \in \{1, \dots, n\}$ and

$$\delta_u = r_u - (|g_{u1}| r_1 + \dots + |g_{un}| r_n). \quad (41)$$

Proof. Note that if $\mathbf{g}_i \mathbf{x}(k) = \operatorname{sgn}(\mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1}) \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) (r_i - \delta_i)$, where $\delta_i \in (0; r_i)$, then (16) has the following form

$$\begin{aligned} K_i &= \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| (r_i - \delta_i) + \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| r_i \\ &= \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| (2r_i - \delta_i). \end{aligned} \quad (42)$$

Otherwise, if $\mathbf{g}_i \mathbf{x}(k) = -\operatorname{sgn}(\mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1}) \operatorname{sgn}(\mathbf{c}^T \mathbf{x}(k)) (r_i - \delta_i)$, then

$$K_i = - \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| (r_i - \delta_i) + \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| r_i = \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| \delta_i. \quad (43)$$

We conclude that if $|\mathbf{g}_i \mathbf{x}(k)| \leq r_i - \delta_i$, then $K_i \geq \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| (2r_i - \delta_i)$ or $K_i \geq \left| \mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1} \right| \delta_i$. Hence, the representative point arrives on the sliding

hyperplane in finite time. Let us notice that if (35) is satisfied for x_i variable, then there exists $\delta_i > 0$ such that

$$|g_{i1}|r_1 + \dots + |g_{in}|r_n = r_i - \delta_i. \quad (44)$$

If (35) is satisfied for each $i \in \{1, \dots, n\}$, then in the whole subset of state-space determined by constraints the condition $K_i \geq |\mathbf{c}^T \mathbf{b} (\mathbf{e}_i \mathbf{b})^{-1}| \delta_i > 0$ is also satisfied. Similarly, if equality

$$|g_{u1}|r_1 + \dots + |g_{un}|r_n = r_u - \delta_u, \quad (45)$$

where $\delta_u \in (0; r_u)$, is true, then in the whole subset of the state-space determined by constraints the condition $K_u \geq |\mathbf{c}^T \mathbf{b}| \delta_u > 0$ is also satisfied. Hence, the parameter ε is of the form (39). This ends the proof. \square

5 Simulation Example

Let us consider the system described by (1), where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -0.99 & -2 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mathbf{c}^T = [1 \ 30]. \quad (46)$$

Our goal is to constraint both state variables in this system. After transformations, matrix \mathbf{G} is of the following form

$$\mathbf{G} = \begin{bmatrix} 0.9903 & 3.871 \\ 0.0003 & 0.871 \end{bmatrix}. \quad (47)$$

The maximum admissible absolute values of the first and the second state variable are $r_1 = 100$ and $r_2 = 0.25$, respectively. The initial state $\mathbf{x}(0) = [95 \ 0]^T$. Rewriting (35) for $i = 1$ and $i = 2$ we get

$$\begin{cases} |g_{11}|r_1 + |g_{12}|r_2 = 99.9978 < 100 = r_1 \\ |g_{21}|r_1 + |g_{22}|r_2 = 0.2478 < 0.25 = r_2 \end{cases}. \quad (48)$$

Inequalities (48) demonstrate that both state variable constraints are satisfied. Evolutions of the first and the second state variable are shown in Fig. 1 and Fig. 2. It can be seen from Fig. 1 that the first state variable is always smaller than 100. Figure 2 shows that the second state variable is equal to its minimum admissible value for a certain period of time. From Fig. 3 we conclude that the finite time convergence to the sliding line is obtained. Analyzing Fig. 4 we notice, that the representative point moves towards the sliding domain along the line representing the second state constraint.

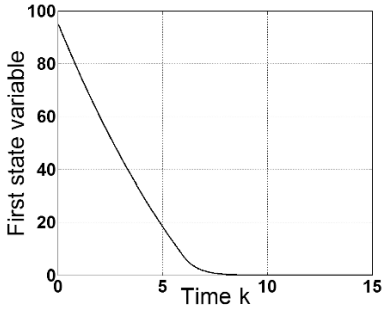


Fig. 1. First state variable

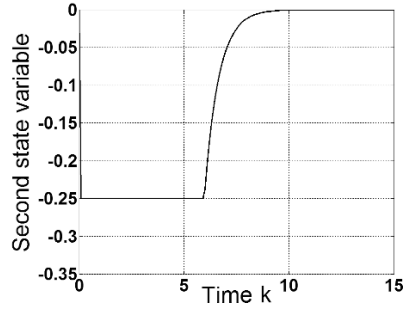


Fig. 2. Second state variable

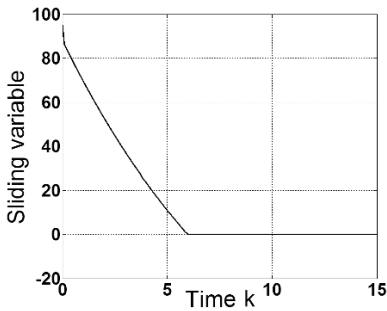


Fig. 3. Sliding variable

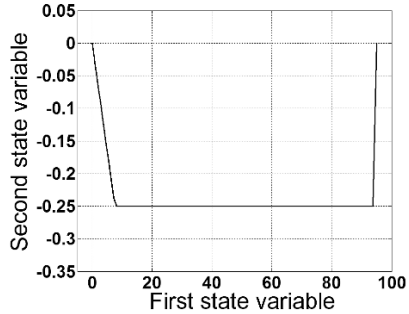


Fig. 4. State trajectory

6 Conclusions

In this paper, the issue of state and input constraints in discrete-time sliding mode control has been considered. In order to obtain the fastest convergence of the representative point to the sliding hyperplane, without violating the restrictions, the new reaching law was designed. Sufficient condition for finite time convergence in the presence of constraints was stated and formally proved. Computer simulations verified theoretical considerations. Our future work will focus on weakening the sufficient condition introduced in this paper.

References

1. Bartolini G., Ferrara A., Utkin V. I.: Adaptive sliding mode control in discrete-time systems. *Automatica*, 31, 769-773 (1995)
2. Bartoszewicz A.: Discrete-time quasi-sliding-mode control strategies. *IEEE Trans. Ind. Electron.*, 45, 633-637 (1998)
3. Bartoszewicz A., Latosiński P.: Discrete time sliding mode control with reduced switching a new reaching law approach. *Intern. Jour. Robust Nonlinear Control*, 26, 47-68 (2016)
4. Bartoszewicz, A., Leśniewski, P.: Reaching law approach to the sliding mode control of periodic review inventory systems. *IEEE Trans. Autom. Sci. Eng.*, 11, 810-817 (2014)

5. Bartoszewicz A., Leśniewski P.: Reaching law based sliding mode congestion control for communication networks. *IET Control Theory Appl.*, 8, 1914-1920 (2014)
6. Bartoszewicz A., Leśniewski P.: New switching and nonswitching type reaching laws for SMC of discrete time systems. *IEEE Trans. Control Syst. Technol.*, 24, 670-677 (2016)
7. Bartoszewicz A., Nowacka A.: Switching plane design for the sliding mode control of systems with elastic input constraints. *Proceedings of institution of mechanical engineers, part I. Jour. Syst. Control Eng.*, 219, 393-403 (2005)
8. Bartoszewicz A., Nowacka A.: Reaching phase elimination in variable structure control of the third order system with state constraints. *Kybernetika*, 42, 111-126 (2006)
9. Bartoszewicz A., Nowacka-Leverton A.: ITAE optimal sliding modes for third order systems with input signal and state constraints. *IEEE Trans. Autom. Control*, 55, 1928-1932 (2010)
10. Chakrabarty S., Bandyopadhyay B.: A generalized reaching law for discrete sliding mode control. *Automatica*, 52, 83-86 (2015)
11. Chakrabarty S., Bandyopadhyay B.: A generalized reaching law with different convergence rates. *Automatica*, 63, 34-37 (2016)
12. Emelyanov S. V.: Variable structure control system. Nauka, Moscow (1967)
13. Fallaha C.J., Saad M., Kanaan H.Y., Al-Haddad K.: Sliding-mode robot control with exponential reaching law. *IEEE Trans. Ind. Electron.*, 58, 600-610 (2011)
14. Furuta K.: Sliding mode control of a discrete system. *Syst. Control Lett.*, 14, 145-152 (1990)
15. Gao W., Hung J. C.: Variable structure control of nonlinear systems: A new approach. *IEEE Trans. Ind. Electron.*, 40, 45-55 (1993)
16. Gao W., Wang Y., Homaifa A.: Discrete-time variable structure control systems. *IEEE Trans. Ind. Electron.*, 42, 117-122 (1995)
17. Golo G., Milosavljević Č.: Robust discrete-time chattering free sliding mode control. *Syst. Control Lett.*, 41, 19-28 (2000)
18. Milosavljević Č.: General conditions for the existence of a quasisliding mode on the switching hyperplane in discrete variable structure systems. *Autom. Remote Control*, 46, 307-314 (1985)
19. Niu Y., Ho D. W. C., Wang Z.: Improved sliding mode control for discrete-time systems via reaching law. *IET Control Theory and Appl.*, 4, 2245-2251 (2010)
20. Qu S., Xia X., Zhang J.: Dynamics of discrete-time sliding-mode-control uncertain systems with a disturbance compensator. *IEEE Trans. Ind. Electron.*, 61, 3502-3510 (2014)
21. Utkin V. I.: Variable structure systems with sliding modes. *IEEE Trans. Autom. Control*, 22, 212-222 (1977)
22. Utkin V. I., Drakunow S. V.: On discrete-time sliding mode control. *IFAC Conf. Nonlinear Control*, 484-489 (1989)
23. Veselić B., Peruničić-Draženić B., Milosavljević Č.: Improved discrete-time sliding-mode position control using Euler velocity estimation. *IEEE Trans. Ind. Electron.*, 57, 3840-3847 (2010)
24. Wang A., Jia X., Dong S.: A new exponential reaching law of sliding mode control to improve performance of permanent magnet synchronous motor. *IEEE Trans. Magn.*, 49, 2409-2412 (2013)
25. Zhang X., Sun L., Zhao K., Sun L.: Nonlinear speed control for PMSM system using sliding-mode control and disturbance compensation techniques. *IEEE Trans. Power Electron.*, 28, 1358-1365 (2013)

Root-locus Design of PID Controller for an Unstable Plant

Leszek Trybus and Zbigniew Świder

Rzeszów University of Technology, Rzeszów, Poland

{ltrybus, swiderzb}@prz.edu.pl

Abstract. Unstable plant considered in the paper consists of an integrator and 1st order component with positive pole. PID controller is used, so the feedback system becomes of 3rd order. Root-locus design method is applied which for such system gives analytic expressions for controller settings. If weak control action is required, the design provides better responses than conventional one. Unstable ships, particularly oil tankers, require weak control.

Keywords: PID controller, root-locus method, design of PID settings, unstable plant.

1 Introduction

Bounded-input bounded-output (BIBO) unstable plant under consideration here is described by the following integrating transfer function

$$\frac{K}{s(Ts+1)}, \quad \text{with } K < 0, \quad T < 0, \quad (1)$$

so one of the poles $0, -1/T$ lies strictly in right-half plane. Unstable ship, where rudder angle is the input and course angle the output, may be modeled by (1) [1, 2]. Such ship for small rudder angle turns in opposite direction than expected. Only for medium and large rudder angles it turns as stable ship, i.e. with positive K and T . The instability is caused by nonlinear characteristic of the rudder, usually described by a 3rd order polynomial.

Unstable ships are rare, with oil tankers as typical examples. Autopilots of such ships involve two basic control modes, course-keeping and turning. In the first mode the ship is kept on constant course despite wind and sea current by applying small rudder motions to prevent displacements of liquid shipload. Then the model (1) is appropriate. However, this requires the controller to be "weak", i.e. with low value of the overall gain. Turning mode handles major changes of the course, so nonlinear characteristic of the rudder must be taken into account.

Typical course-keeping autopilot involves PID controller. Conventional tuning rules for the settings K_p , T_i , T_d are given in [1], covering both stable and unstable ships. Closed-loop natural frequency is basic design parameter. However, T_i in those rules is selected by a "rule-of-thumb" what may rise some questions. Therefore here we propose a rigorous, root-locus based method [3] of PID controller tuning for unstable ship. Responses for weak control action are shown in examples.

Among relevant literature, analytic expressions for PID settings obtained from ITAE criterion are also given in [4]. PD controller taking into account rudder nonlinearity is considered in [5]. Optimal tracking control of an ESSO tanker is described in [6]. Earlier papers on nonlinear ships applied adaptive control, as e.g. [7]. Recent literature focuses on applications of fuzzy, neural, genetic, and ant colony algorithms.

2 Conventional Design

Conventional design of PID controller for course-keeping involves two steps. First a PD controller $K_p(1+T_d s)$ is designed, so as the denominator $s^2 + 2\xi\omega_n s + \omega_n^2$ of the closed-loop transfer function would have some natural frequency ω_n and damping ratio $\xi \in [0.8, 1]$. This yields

$$K_p = \frac{T}{K} \omega_n, \quad T_d = \frac{2\xi T \omega_n - 1}{T \omega_n^2}, \quad T_i = \frac{10}{\omega_n} \quad (2)$$

Then T_i of PID controller $K_p(1+1/(T_i s)+T_d s)$ is selected by "rule-of-thumb" at $10/\omega_n$, as shown above.

In case of unstable ship, the rules may be written as

$$K_p = \frac{|T|}{|K|} \omega_n^2, \quad T_d = \frac{2\xi |T| \omega_n + 1}{|T| \omega_n^2}, \quad T_i = \frac{10}{\omega_n} \quad (3)$$

Controller properties are determined by the ratio

$$\frac{T_i}{T_d} = \frac{10|T|\omega_n}{2\xi|T|\omega_n + 1} \quad (4)$$

Suppose $\xi = 1$. If $|T|\omega_n = 2$, then $T_i/T_d = 4$ what means that PID controller has double zero at $-1/(2T_d)$ (or at $-2/T_i$). For $|T|\omega_n < 2$ the zeroes are complex, and for $|T|\omega_n > 2$, real distinct. So selection of ω_n affects relative locations of the closed-loop poles.

3 Root-locus Design

To keep relative locations of poles independent of ω_n the ratio T_i/T_d should be constant. Recall that Ziegler-Nichols tuning rules where

$$\frac{T_i}{T_d} = 4 \quad (5)$$

are well-established in process control. Then the PID transfer function has the double zero, so can be written as

$$\text{PID: } K_p T_d \frac{(s + \frac{1}{2T_d})^2}{s} \quad (6)$$

Introduce normalized Laplace operator $s' = |T|s$. For the plant (1) and controller (6), the open-loop transfer function becomes

$$G_{open}(s') = k \frac{(s' + z)^2}{s'^2 (s' - 1)}, \quad s' = |T|s \quad (7a)$$

where

$$k = K_p |K| T_d, \quad z = \frac{|T|}{2T_d} \quad (7b)$$

z may be called a normalized zero. Note that unlike in the conventional design, here we deal with 3rd order system.

Root-locus plot of $G_{open}(s')$ with respect to k is shown in Fig.1.

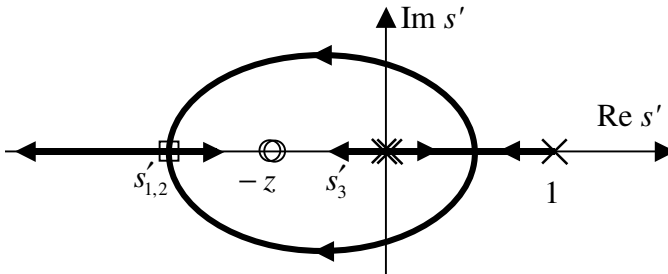


Fig. 1. Root-locus plot of $G_{open}(s')$

As k increases, two roots move from the right-half plane to the left and meet at the breakpoint $s'_{1,2}$. The third root s'_3 comes out of the origin and tends towards $-z$ along real axis.

Assume that we want to place the two roots at the breakpoint $s'_{1,2}$. It can be found from the condition $dG_{open}(s')/ds = 0$ what yields

$$s'_{1,2} = -\frac{3z + \sqrt{9z^2 + 8z}}{2} \quad (8a)$$

Gain k corresponding to $s'_{1,2}$ is calculated as

$$k = -\frac{s'^2(s'-1)}{(s'+z)^2} \Big|_{s'=s'_{1,2}} \quad (8b)$$

From (7b) we get

$$K_p = \frac{k}{|K|T_d} \quad (8c)$$

The plot of the function $k(z)$ given by (8a,b) is shown in Fig.2a for $z \in [0.01, 1.0]$. Such range corresponds to $|T|\omega_n$ from 0.16 up to 4.45, so from very weak controller up to reasonably strong. Naturally, small rudder changes for unstable ship are provided by weak controller.

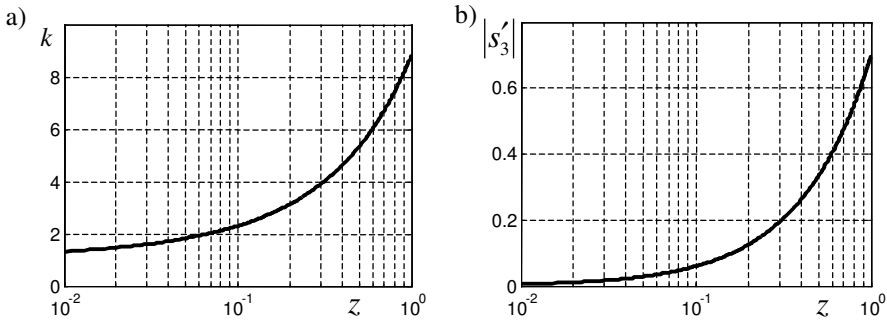


Fig. 2. Design supporting functions: a) gain $k(z)$, b) dominant root $|s'_3(z)|$

Assume that settling time is a basic design parameter. It is determined by the dominant third root s'_3 in Fig.1, so

$$t_{settle} \cong 4 \frac{|T|}{|s'_3|} \quad (9)$$

Note that besides Matlab, s'_3 can be obtained manually by dividing the polynomials

$$\frac{s'^2(s'-1) + k(s'+z)^2}{(s'-s'_{1,2})^2} = s' - s'_3 \quad (10a)$$

(Bézout rule) what yields

$$s'_3(z) = 1 - k(z) - 2s'_{1,2}(z) \quad (10b)$$

with $k(z)$ and $s'_{1,2}(z)$ from (8a,b). Explicit expression for $s'_3(z)$, as fairly long, is not given here. The plot of $|s'_3(z)|$ is shown in Fig.2b.

For design purposes we assume that the functions $k(z)$ and $|s'_3(z)|$ are available, practically in the form of look-up tables.

4 Comparison of the Designs

In [1] (p.261) a test example of conventional design is presented for $K = -0.1$ and $T = -10$. For natural frequency $\omega_n = 0.05$ and damping ratio $\xi = 0.8$ the rules (2) yield $K_p = 0.25$, $T_d = 72$ and $T_i = 200$. Controller zeroes are complex, $-0.0069 \pm j0.0046$. Two of the closed-loop poles are also complex, $-0.037 \pm j0.0262$, and the third one s_3 , the dominant, lies at -0.0061 . So the settling time becomes $t_{settle} = 4/|s_3| = 656$. Note that it is 65 times larger than the time parameter $|T|$ of the plant, so it represents a weak controller. Unit-step load-disturbance output and control responses are shown in Fig.3a,b (disturbance suppression is the task of course-keeping autopilot).

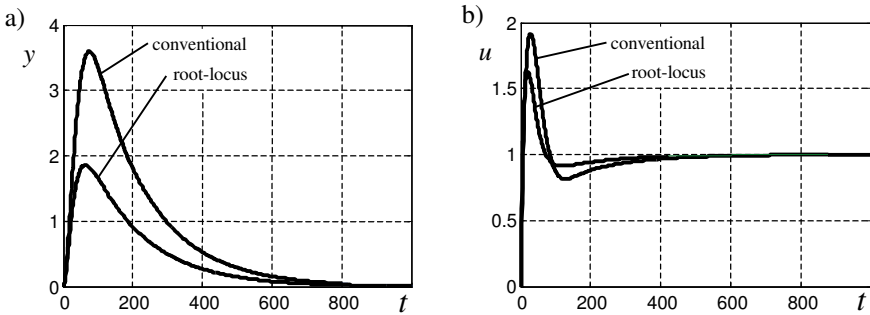


Fig. 3. Load-disturbance responses for test data: a) output y , b) control u

Now suppose we want to get the same t_{settle} for root-locus design, so the same dominant pole s_3 . Since $|T| = 10$, the normalized pole $s'_3 = |T|s_3$ is -0.061 . From $|s'_3(z)|$ characteristic (Fig.2b) for $|s'_3| = 0.061$ one gets the normalized zero

$z = 0.102$. Now from (7b) we obtain $T_d = |T|/(2z) = 49$ and $T_i = 4T_d = 196$. $k(z)$ characteristic (Fig.2a) yields $k = 2.32$ for $z = 0.102$. So finally $K_p = 4/(k|T_d) = 0.469$. Load-disturbance output and control responses for such settings are also in Fig.3a,b. Close-loop poles are at $-0.0619 \pm j0.0105$, -0.0061 (small imaginary part caused by roundings).

As a second example we take data of an oil tanker [1] (p.174), whose more accurate description has the form

$$\frac{K(T_3s+1)}{s(T_1s+1)(T_2s+1)}, \quad \text{with } k = -0.019, \quad T_1 = -124.1, \quad T_2 = 16.3, \quad T_3 = 46 \quad (11)$$

Such transfer function, but without the integral, is called 2nd order Nomoto model, whereas (1) represents 1st order Nomoto. The model (11) will be used for simulation. 1st order Nomoto model needed for PID design has the same K and T calculated as $T_1 + T_2 - T_3$. So $T = -153.7$ here.

This time we will proceed the other way, i.e. assuming certain t_{settle} the root-locus tunings will be calculated as above. They will give some dominant pole $s_3 = |T|s'_3$ of the closed-loop system. Then by trial-and-error such natural frequency ω_n will be found, so as the conventional tunings would provide the same dominant pole s_3 .

Let $t_{settle} = 7200$ seconds (2 hours) to get a controller with weak control action. Following the root locus design we calculate $|s'_3| = 4|T|/t_{settle} = 0.0854$, $z = 0.14$ from Fig.2a for such $|s'_3|$, $T_d = |T|/(2z) = 549$, $T_i = 4T_d = 2196$, $k = 2.64$ from Fig.2a for $z = 0.14$, $K_p = k/(|K|T_d) = 0.469$.

After a few trials one can find that for $\omega_n = 0.00485$ conventional tunings $K_p = 0.19$, $T_i = 2062$, $T_d = 606$ provide closed-loop dominant pole at $s_3 = |T|s'_3 = -0.00055$. Note that $|T|\omega_n = 0.745$ here, so $t_{settle} = 7200$ is 47 times larger than $|T| = 153.7$. Load-disturbance responses for the two designs are shown in Fig.4a,b.

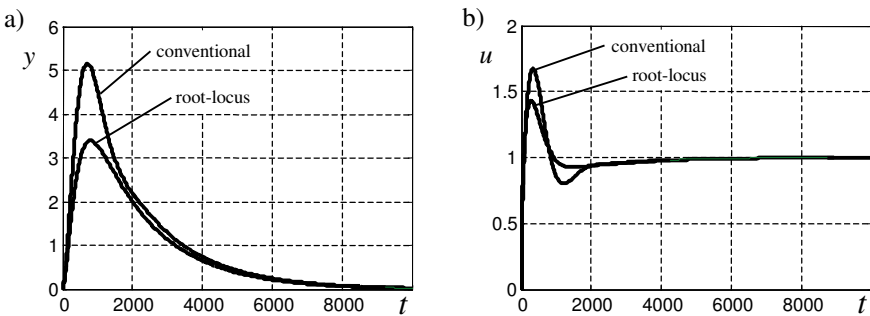


Fig. 4. Load-disturbance responses for the oil tanker: a) output y , b) control u

The two examples indicate that root-locus design may be beneficial under the condition of weak control action. The benefit is seen for $|T|\omega_n \leq 1.0$. For larger $|T|\omega_n$ responses resulting from conventional design begin to look better. For instance, if $|T|\omega_n = 2$ the maximum of load-disturbance output response for conventional design is lower by 13% than that for root-locus, and if $|T|\omega_n = 4$, by over 30%.

5 Conclusions

Root-locus method has been applied to develop PID controller tuning rules for integrating transfer function with additional positive pole. Assuming that the controller has double zero, as in the Ziegler-Nichols tunings, we have been able to find analytic expressions for the rules. If weak control action is required, the root-locus design exhibits some advantages over conventional approach. Autopilots of unstable ships, particularly oil tankers, provide weak control.

References

1. Fossen, T.I.: *Guidance and Control of Ocean Vehicles*, 4th edn. Wiley, Chichester, 1999
2. Lisowski, J.: *Ship as Automatic Control Plant*. Wyd. Morskie, Gdańsk, 1981 (in Polish)
3. Dorf, R.C., and Bishop, R.M.: *Modern Control Systems* (11th ed.). Prentice Hall, Upper Saddle River, NY, 2008
4. Morawski L., Pomirski J.: Design of the robust PID course-keeping control system for ship. *PMR*, 6 p., 2002
5. Morawski L., Pomirski J.: Identification and control of a direction unstable ship. *Problems of Nonlinear Analysis in Engineering Systems*, v. 7, 1(13), 2001 (Kazan)
6. Cimen T., Banks S.P.: Nonlinear optimal tracking control with application to super-tankers for autopilot design. *Automatica*, v.40, p.1845-1863, 2004
7. Åström K.J.: Why use adaptive techniques for steering large tankers? *Int. J. Control*, v. 32, no. 4, p.689-708, 1980

A comparison of different dynamic decoupling methods for a nonlinear MIMO Plant

Paweł Dworak¹, Michał Brasel¹, Sandip Ghosh²

¹ West Pomeranian University of Technology, Szczecin,
ul. 26 Kwietnia 10, 71-126 Szczecin, Poland
pawel.dworak@zut.edu.pl, michal.brasel@zut.edu.pl

² Indian Institute of Technology (Banaras Hindu University),
Varanasi 221005 (UP) INDIA
sghosh.eee@iitbhu.ac.in

Abstract. In the paper two different control systems for nonlinear multi-input multi-output MIMO plants are compared and discussed. Their main objective is to reduce the influence of the plant inputs and outputs. A multi-controller control structure, which contains a set of a dynamic decoupling controllers is compared with a synthesized online nonlinear model predictive controller. Pros and cons of both methods are discussed and presented in series of simulations of control of a selected nonlinear MIMO plant. The paper ends with some final remarks on a practical implementation of decoupling methods for a nonlinear MIMO plants.

Keywords: dynamic decoupling, nonlinear plants, MIMO, MPC, TS fuzzy controllers

1 Introduction

Control of multi-input multi-output MIMO dynamic systems enjoys a continuous attention. New methods are proposed e.g. for plants identification, autotuning of PID controllers, its robust optimization or adaptive control of MIMO T-S Fuzzy system. The same applies to the dynamic decoupling problem. However despite it has been investigated for many years and has been solved for LTI plants it is still an object of interest [5, 15, 24, 27] particularly for nonlinear ones where it may be treated like an open question.

A full dynamic decoupling of a nonlinear plant needs a global linearization [20, 21]. [19] gives necessary and sufficient conditions for the strong input-output decoupling problem with the use of static measurement feedback for a some type of nonlinear square control plants. But such methods has its limits and e.g. the results obtained may be not robust to the plant perturbations.

As the full decoupling of the nonlinear plant is extremely difficult to obtain practical realizations in such cases boil down to minimization of the coupling interactions only instead of full decoupling. Authors of [22, 23] consider problems of robust decoupling of linear systems with nonlinear uncertain structure with the use of output and state feedback. Robust decoupling controllers for

uncertain MIMO systems are also proposed in [18] where a parametric uncertainty model is used to describe the system behavior and the controller design is expressed as a min-max non-convex optimization problem with taking into account the desired performance and uncertainties. In [31, 32] for a discrete-time nonlinear MIMO system, a multiple models fuzzy and neural network decoupling controllers were proposed. In these proposals the system is expanded into a linear and nonlinear term at each equilibrium point. Then at each instant, the best model is chosen as the system model according to the switching index. To design the controller accordingly, the nonlinear term and the interactions of the best model are viewed as measurable disturbances and eliminated by the use of the feedforward strategy. Switching multivariable controllers are also used in [12–14] where the nonlinear plant is linearized at each working point and a switching, fuzzy and neural decoupling controllers are constructed. A survey on a decoupling control based on multiple models is presented in [25].

At the same time many researchers tried to find methods of decoupling which do not require synthesis of the complicated multivariable controller. One of the natural way seems to be a predictive control [1, 16, 17, 26, 29, 30, 34, 35]. However as we see in these works the MIMO predictive controller does not automatically enhance decoupling. Additionally these methods may often be either unsolvable or computationally unrealizable when used on-line. Thus most of them are constructed and tested for specific 2 by 2 (TITO) nonlinear plants [1, 26, 30, 34, 35].

[26] proposes a multivariable fuzzy predictive functional control where the control law is given in an analytical form. In works [1, 30, 34, 35] predictive control is supported by some additional techniques to obtain dynamic decoupling effects. They are: deceleration of the reference signals change in order to make the control slower and error weighting factors in the cost function changing. Of course, the adoption of the weighting factors has to be synchronized to the reference signal change. In this paper we are going to show that all of the above methods are necessary to obtain the dynamic decoupling effect and that nonlinear predictive control does not decouple plants automatically and its implementation and its implementation and on-line using may be problematic.

In the paper we compare two different control systems whose main goal is a dynamic decoupling of a nonlinear multi-input multi-output MIMO plant. A multi-controller control structure, which contains a set of a dynamic decoupling controllers is presented and compared with a synthesized online nonlinear model predictive controller. To do that the paper is organized as follows. In Section II a dynamic decoupling problem is defined. Then we present shortly two compared control methods. In Section III a multivariable switching controller and adopted method of synthesis of the dynamic decoupling controller for a LTI MIMO plants, in Section IV a model predictive control idea. Pros and cons of both methods are discussed and presented in series of simulations of control of a selected nonlinear MIMO plant in Section 5. The paper ends with some final remarks on a practical implementation of decoupling methods for a nonlinear MIMO plants.

2 Problem statement

Let us assume the plant described by the nonlinear state space and output equations

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) &= \mathbf{g}(\mathbf{x}(t))\end{aligned}\quad (1)$$

where $\mathbf{x}(t) \in R^n$, $\mathbf{u}(t) \in R^m$ and $\mathbf{y}(t) \in R^l$ are the state, input and output vectors respectively. The goal of dynamic (block or diagonal) decoupling is to separate the control system into $i = 1, 2, \dots, k$ independent control loops with its outputs and reference signals grouping according to the partitions

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{y}_1(t) \\ \vdots \\ \mathbf{y}_i(t) \\ \vdots \\ \mathbf{y}_k(t) \end{bmatrix}, \mathbf{y}_0(t) = \begin{bmatrix} \mathbf{y}_{01}(t) \\ \vdots \\ \mathbf{y}_{0i}(t) \\ \vdots \\ \mathbf{y}_{0k}(t) \end{bmatrix}, i = 1, 2, \dots, k \quad (2)$$

where $\mathbf{y}_i(t) \in R^{l_i}$, $\mathbf{y}_{0i}(t) \in R^{l_i}$, $\sum_{i=1}^k l_i = l$. Then each part (loop) $i = 1, 2, \dots, k$ of a system is defined by pairs of signals $\mathbf{y}_{0i}(s)$, $\mathbf{y}_i(s)$ which could be controlled independently of other parts $j \neq i$.

We also assume that each part of the system should be designed with individually supposed dynamic properties.

3 Concept of the decoupling multivariable switching controllers

3.1 Multivariable switching controller

As we see e.g. in [2, 11–13] the control system for the nonlinear MIMO plant may consist of an adaptive controller designed on the basis of set of modal or dynamic decoupling controllers. These linear controllers are synthesized for possibly all operating points of the plant - all controllable and observable LTI MIMO models defined by the state and output equations

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}[\mathbf{x}(t) - \mathbf{x}_0] + \mathbf{B}[\mathbf{u}(t) - \mathbf{u}_0] \\ \mathbf{y}(t) - \mathbf{y}_0 &= \mathbf{C}[\mathbf{x}(t) - \mathbf{x}_0]\end{aligned}\quad (3)$$

obtained by linearization of the model (1) at all working points $(\mathbf{x}_0, \mathbf{u}_0)$. Such a control system can be realized with a single adaptive controller with stepwise tuned parameter values [2, 11–13] (Fig. 1).

This adaptive controller may be realized by a fuzzy controller [11, 13] where a group of linear controllers, appropriate to a given operation conditions is chosen and used to calculate, by employing Takagi-Sugeno (T-S) fuzzy rules, control

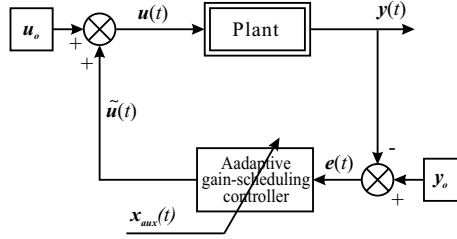


Fig. 1. General scheme of the switching control system.

signals. Such an adaptive controller (stepwise) varies its parameters depending on the current plant operating point.

An example given in [13] shows that such constructed T-S controller allows one to obtain a significant reduction of interactions between plants inputs and outputs. It was also pointed out that the proposed solution allows one to soften stability conditions during synthesis of linear controllers used in a T-S fuzzy controller. Local controllers do not have to be stable by themselves, which considerably softens synthesis constraints. It is also easy to check system stability and very simple fuzzy rules adopted make the control system easy to implement in any programmable controller.

3.2 Dynamic decoupling controller

To synthesize a linear decoupling controller used in a multivariable switching controller described in the previous subsection one can use a decoupling algorithm presented in detail in [8, 10]. This decoupling concept uses the state vector feedback together with a feedforward compensator and apart of dynamic decoupling simultaneously ensures an arbitrary closed-loop dynamics - independent for each decoupled part, zero steady-state regulations errors and reconstruction of the plants state vector. The algorithm may be used for plants described by rectangular proper rational full rank transfer matrix, plants which could be unstable, non-minimum phase or both. The control system contains a feedforward compensator together with a feedback matrix \mathbf{F} and a controller. Structure of this decoupled closed-loop system for a plant with accessible state vector $\mathbf{x}(t)$ is presented in Fig. 2.

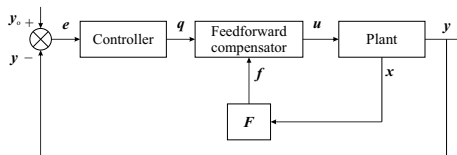


Fig. 2. General scheme of the dynamically decoupled control system.

Results obtained with the use of this algorithm for the linear plant (3) are: a state feedback matrix \mathbf{F} , a dynamic feedforward compensator and a diagonal controller. Describing the feedforward compensator by the state and output equations

$$\begin{aligned}\dot{\mathbf{x}}_w(t) &= \mathbf{A}_w \mathbf{x}_w(t) + \mathbf{B}_w \mathbf{u}_w(t) \\ \mathbf{u}(t) &= \mathbf{C}_w \mathbf{x}_w(t) + \mathbf{D}_w \mathbf{u}_w(t)\end{aligned}\quad (4)$$

where $\mathbf{u}_w(t) = \begin{bmatrix} \mathbf{q}(t) \\ \mathbf{F}\mathbf{x}(t) \end{bmatrix}$ and $\mathbf{B}_w = [\mathbf{B}_{wp} \quad \mathbf{B}_{wm}]$, $\mathbf{D}_w = [\mathbf{D}_{wp} \quad \mathbf{D}_{wm}]$ with $\mathbf{B}_{wp} \in R^{n_w \times l}$, $\mathbf{B}_{wm} \in R^{n_w \times m}$, $\mathbf{D}_{wp} \in R^{m \times l}$, $\mathbf{D}_{wm} \in R^{m \times m}$ and the controller in the form

$$\begin{aligned}\dot{\mathbf{x}}_r(t) &= \mathbf{A}_r \mathbf{x}_r(t) + \mathbf{B}_r e(t) \\ \mathbf{q}(t) &= \mathbf{C}_r \mathbf{x}_r(t)\end{aligned}\quad (5)$$

with $\mathbf{e} = \mathbf{y}_o(t) - \mathbf{y}(t)$ we obtain a linear decoupling controller

$$\begin{aligned}\dot{\mathbf{x}}_{rw}(t) &= \mathbf{A}_{rw} \mathbf{x}_{rw}(t) + \mathbf{B}_{rw} \mathbf{u}_z(t) = \begin{bmatrix} \mathbf{A}_r & \mathbf{0} \\ \mathbf{B}_{wp} \mathbf{C}_r & \mathbf{A}_w \end{bmatrix} \mathbf{x}_{rw}(t) + \begin{bmatrix} \mathbf{B}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{wm} \mathbf{F} \end{bmatrix} \mathbf{u}_z(t) \\ \mathbf{y}(t) &= \mathbf{C}_{rw} \mathbf{x}_{rw}(t) + \mathbf{D}_{rw} \mathbf{u}_z(t) = [\mathbf{C}_w \quad \mathbf{D}_{wp} \mathbf{C}_r] \mathbf{x}_{rw}(t) + [\mathbf{0} \quad \mathbf{D}_{wm} \mathbf{F}] \mathbf{u}_z(t)\end{aligned}\quad (6)$$

where $\mathbf{u}_z(t) = [\mathbf{e}^T(t) \quad \mathbf{x}^T(t)]^T$ and $\mathbf{x}_{rw}(t) = [\mathbf{x}_r^T(t) \quad \mathbf{x}_w^T(t)]^T$.

Such synthesized linear controllers dynamically decouple the nonlinear plant model at its operating points. Thus an adaptive controller may vary its parameters depending on the current plant operating point.

4 Dynamic decoupling with the model predictive control

Another method which allows to solve the posed problem may be the use of predictive controller which optimizes the relevant quality control criterion. Such a criterion is usually formulated in the form of the sum of the square norms of the difference in predicted output signals and corresponding reference signals and predicted control signals increments specified in the relevant time horizons: prediction N_p and control N_u . The cost function may be written in the form

$$\mathbf{J}(t) = [\mathbf{y}_o^p(t) - \mathbf{y}^p(t)]^T \mathbf{M} [\mathbf{y}_o^p(t) - \mathbf{y}^p(t)] + [\Delta \mathbf{u}^p(t)]^T \mathbf{\Lambda} [\Delta \mathbf{u}^p(t)] \quad (7)$$

or in a norm form

$$\mathbf{J}(t) = \|\mathbf{y}_o^p(t) - \mathbf{y}^p(t)\|_{\mathbf{M}}^2 + \|\Delta \mathbf{u}^p(t)\|_{\mathbf{\Lambda}}^2 \quad (8)$$

with the weighting matrices

$$\mathbf{M} = \text{diag}[\mu_i], \quad i = 1, \dots, p$$

$$\mathbf{\Lambda} = \text{diag}[\lambda_j], \quad j = 1, \dots, m$$

where

\mathbf{y}_o^p describes a vector of the predicted reference signals,

\mathbf{y}^p is a vector of the predicted output,

Δu^p predicted control signal increments,
 μ_i control error weighting factor for the i -th output,
 λ_j control increments weighting factor for the j -th output.

As typically for the predictive controllers the actual control signals are used only and the computation is repeated in the next control step. Solving the criterion (7) needs to find a constrained minimum of a function of several variables which is the main cause of problems with the implementation and use of these controllers on-line.

5 Example

5.1 The plant model

The comparison of the discussed control methods will be exemplified by a positioning control system for the MIMO nonlinear dynamic model of the drillship “Wimpey Sealab” [33]. This 3DOF nonlinear model of ship’s slow-varying motions has been used in our previous works [3, 9, 11, 12, 14], thus may be used as a reference model for current considerations. It is described in the form of nonlinear state space and output equations

$$\begin{aligned}
 \dot{x}_1 &= x_4 \cos x_3 - x_5 \sin x_3 + V_c \cos \Psi_c, \\
 \dot{x}_2 &= x_4 \sin x_3 + x_5 \cos x_3 + V_c \sin \Psi_c, \\
 \dot{x}_3 &= x_6, \\
 \dot{x}_4 &= 0.088x_5^2 - 0.132x_4V_s + 0.958x_5x_6 + 0.958u_1, \\
 \dot{x}_5 &= -1.4x_5V_s - 0.978x_5^3/V_s - 0.543x_4x_6 + 0.037x_6|x_6| + 0.544u_2, \\
 \dot{x}_6 &= (0.258x_5V_s - 0.764x_4x_5 - 0.162x_6|x_6| + u_3)/a, \\
 y_1 &= x_1, y_2 = x_2, y_3 = x_3,
 \end{aligned} \tag{9}$$

where state variables x_1, \dots, x_3 represent ship position and course angle over the drilling point and x_4, \dots, x_6 her longitudinal, transversal and angular velocities, $V_s = \sqrt{x_4^2(t) + x_5^2(t)}$ is the ship’s velocity measured with respect to water. Coefficient $a = k_{zz}^2 + 0.0431$ describes the ship’s inertia moment, k_{zz}^2 is the square of the relative inertia radius referenced to the ship’s length L_{pp} . V_c and Ψ_c denote the sea current velocity and direction, respectively (in Fig. 3). All the signals appearing in the equations (9) are dimensionless, i.e. related to the ship’s dimensions and displacement together with the dimensionless time $t = t_r / \sqrt{L_{pp}/g} \approx 0.32t_r$.

5.2 Multivariable and predictive controllers in a positioning control system of a ship

To compare the described above control systems we have carried out a series of simulations in Matlab/Simulink environment. The control goal was to keep the drilling setpoint (course angle y_{3o} and the both ship’s position coordinates $y_2(t)$ and $y_3(t)$) which in order to show the dynamic decoupling abilities of both controllers has been changed several times.

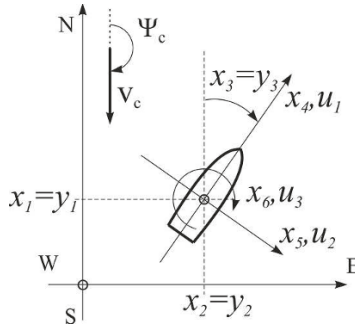


Fig. 3. Ship's co-ordinate systems.

The multivariable controller presented in Section III has been constructed exactly as in [13]. It is a Takagi-Sugeno fuzzy controller whose parameter values are changed on the basis of auxiliary variables measured. These are the ship's current transitional velocity $V_s(t)$ measured with respect to water and the systematically calculated difference between the sea current angle and the ship's course angle $\Psi_c - x_3(t)$. It allows us to synthesize the control system with a group of linear controllers calculated for velocities $V_s \in [-4.9 \div 4.9]$ knots with the resolution of 0.2 knot and round angle $\Psi_c - x_{30} \in [0 \div 360^0]$ with the resolution of 5^0 (about 0.0873 rad), which results in a set of 3650 decoupling controllers. Results of simulation for this controller are presented in figure (4). It may be seen that the controller significantly reduces interactions between plants inputs and outputs. All this with the acceptable control signal values and way of changing.

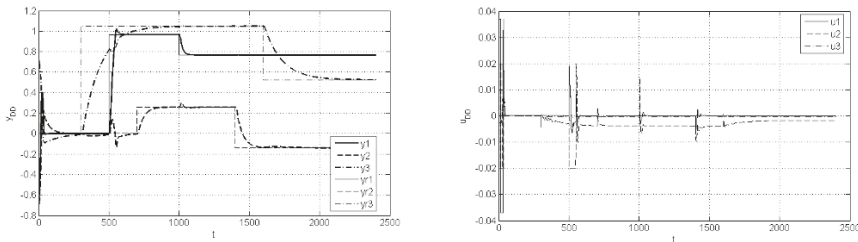


Fig. 4. Ships position and yaw angle and control signals for a T-S fuzzy dynamic decoupling controller.

Figure (5) presents results of simulation of a model predictive control system with solving quality criterion (7) at each time instant. For the algorithm pur-

poses the continuous time ship's model (9) has been discretized with the sampling time $T_p = 0.5s$. A Matlab "fmincon" function with default Sequential Quadratic Programming (SQP) method and a prediction $N_p = 20$ and control $N_u = 5$ horizon has been utilised. Values of the weighting matrices $\mathbf{M} = \text{diag}[10; 0.5; 0.01]$ and $\mathbf{\Lambda} = \text{diag}[0.01; 0.01; 1]$ has been chosen to obtain a system dynamics similar to the one assumed for the multivariable controller as in Fig. (4).

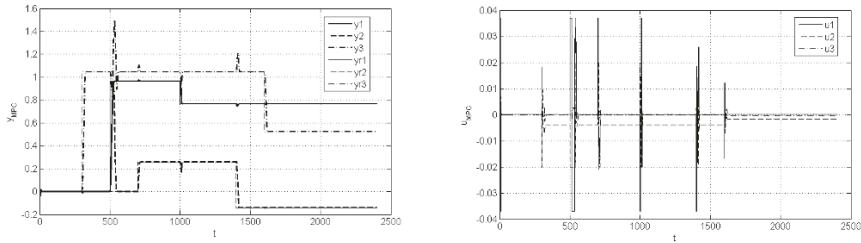


Fig. 5. Ships position and yaw angle control signals for MPC controller.

Analysis of the results presented in Fig. (5) shows a series of drawbacks of this control method (pure MPC algorithm). First of all the system is not decoupled. Secondly it is quite difficult to shape the dynamics for the particular control loop. E.g. we are not able to slow down changing the ship course. We also observe a bigger control signal values. All this means that the pure MPC controller does not satisfy our control goals and the control scheme and algorithm have to be modified.

5.3 Reference signal change

In typical control systems a reference values are changed stepwise. However, in the case of predictive control we have to shape the reference signals, which allows us to change the outputs more smoothly, shape the dynamics in the different control loops independently. Thus we have simulated the above control systems with a decelerated reference signal change. Results of these experiments are presented in Fig. (6) and (7). All steps of the reference signal were filtered by a first-order filter with time constants $T_{1,2,3} = 5s$. As we can see the couplings have been significantly reduced, in both cases. This means that a modification necessary for MPC controller improves also the quality control for the multivariable decoupling controller. The comparison still shows that interactions and signal values for the MPC controller are bigger than for its multivariable counterpart.

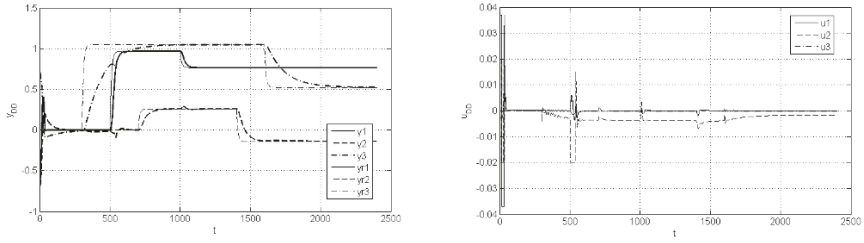


Fig. 6. Ships position and yaw angle control signals for a T-S fuzzy dynamic decoupling controller with filtered reference signals.

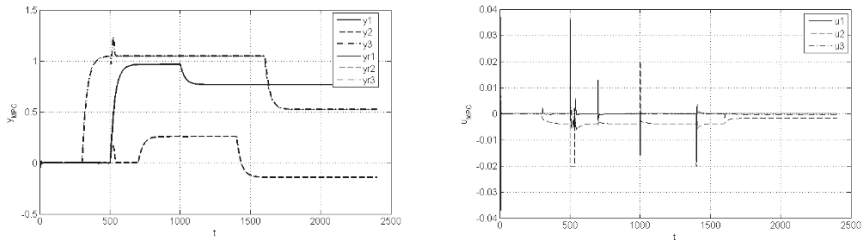


Fig. 7. Ships position and yaw angle control signals for MPC controller with filtered reference signals.

5.4 Adjusting weighting factors

Analysing the presented above results we see we can not divide the system with a MPC into several separated SISO control loops. It is possible to separate some specific outputs which e.g. means that a significant reduction of the coupling effects is here possible after change of the one output only. We can do that by changing the weighting factors values. The simplest and in our opinion the best way of changing these factors is its manual change at the moments of changing of the reference signals. As we see in Fig.(8) changing weighting factors μ_i reduces effects of coupling. It is done with the change of matrix \mathbf{M} to e.g. $\mathbf{M}_1 = \text{diag}[10; 50; 1]$, $\mathbf{M}_2 = \text{diag}[1000; 0.5; 1]$ and $\mathbf{M}_3 = \text{diag}[1000; 50; 0.01]$ after change of the reference of the first, second and third output, respectively - which was obtained after multiplication of appropriate weighting factors of the used before weighting matrix $\mathbf{M} = \text{diag}[10; 0.5; 0.01]$ by 100.

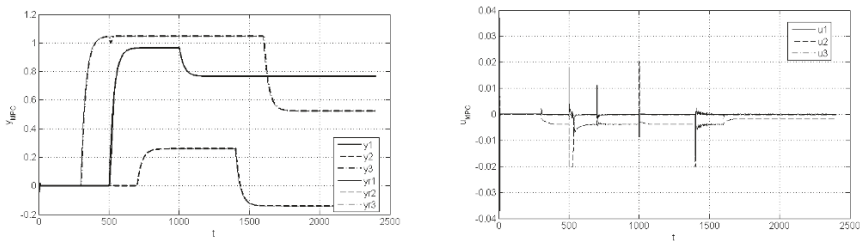


Fig. 8. Ships position and yaw angle control signals for MPC controller with filtered reference signals and adjusted weighting factors.

Similar technique has been applied in [4] where values of changed weighting factors decreased exponentially. We must be aware that such method is possible only if we know in advance the schedule of the reference signal changing.

In [30] and then [34] the weighting factor was dependent on the control error as in the formula

$$\mu_i = \frac{\mu_{i,\max}}{1 + |e_i(k)| \mu_{i,\text{damp}}} \quad (10)$$

where the maximal $\mu_{i,\max}$ and damping values of the factors were assumed after some simulations in a trial and error method. However, such algorithm may not work properly if the reference signals is changing stepwise - with reduced speed of the reference signal change as in previous subsection.

6 Summary

A pure predictive controller, without any modification, do not guarantee decoupling. It is due to the sense of the criterion which is to minimize the criterion

itself, not any specific by effects. Thus, the whole decoupling is simply not possible. Reducing of the input-output interactions is possible after modification of the reference signals and weighing factors in the quality control criterion. However, the presented in the paper examples shows that the same techniques may also improve results obtained by the use of another type of controllers. And there is still problem of the controller implementation and its use in an on-line mode. The multivariable controller is synthesized off-line while the MPC is calculated at each time instant with unspecified calculation time for SQP method. In our research, for the assumed prediction and control horizon the MPC simulation took too long to be used on-line.

References

1. Arousi F., Predictive control algorithms for linear and nonlinear processes, *PhD. Thesis*, Budapest, Hungary, (2009)
2. Bańka S., Dworak P., Jaroszewski K., Linear adaptive structure for control of a nonlinear MIMO dynamic plant, *International Journal of Applied Mathematics and Computer Science*, Vol. 23, No. 1., pp. 47–63 (2013)
3. Bańka S., Dworak P., Jaroszewski K., Design of a multivariable neural controller for control of a nonlinear MIMO plant, *International Journal of Applied Mathematics and Computer Science*, Vol. 24, No. 2., pp. 357–369 (2014)
4. Bego O., Peric N., Petrovic I., Decoupling multivariable GPC with reference observation. *10th Mediterranean Electromechanical Conference*, Vol. II, pp. 819–822 (2000)
5. Chiu Ch-S., A dynamic decoupling approach to robust T-S fuzzy model-based control, *IEEE Transactions on Fuzzy Systems*, Vol. 22, No. 5, pp. 1088–1100 (2014)
6. Dworak, P., Bańka S.: Efficient algorithm for synthesis of multipurpose control systems with dynamic decoupling. *MMAR05*, Miedzyzdroje, 345–350 (2005)
7. Dworak P., Pietruszewicz K., Domek S., Improving stability and regulation quality of nonlinear MIMO processes, *Methods and Models in Automatic and Robotics*, (2009)
8. Dworak, P.: Dynamic decoupling of left-invertible MIMO LTI plants. *Archives of Control Science* 21(4), 443–459 (2011)
9. Dworak P., Brasel M., Improving quality of regulation of a nonlinear MIMO dynamic plant, *Electronics and Electrical Engineering*, Vol. 19, No. 7, pp. 3–6 (2013)
10. Dworak, P.: Squaring down plant model and I/O grouping strategies for a dynamic decoupling of left-invertible MIMO plants. *Bulletin of the Polish Academy of Sciences*,
11. Dworak P., A Type of Fuzzy T-S Controller for a Nonlinear MIMO Dynamic Plant, *Elektronika ir elektrotechnika*, Vol. 20, No. 5, pp. 8–14 (2014)
12. Dworak P., Jaroszewski K., About Dynamic Decoupling of a Nonlinear MIMO Dynamic Plant, *Methods and Models in Automatic and Robotics*, Miedzyzdroje, Poland, pp. 106–111 (2014)
13. Dworak P., Dynamic Decoupling of a Nonlinear MIMO Plant, in *Aktualne Problemy Automatyki i Robotyki*, Eds. Jzefczyk J., Malinowski K., witek J., EXIT, pp. 158–167 (2014)
14. Dworak P., Jaroszewski K., Neural Networks for a Dynamic Decoupling of a Nonlinear MIMO Dynamic Plant, *Methods and Models in Automatic and Robotics*, Miedzyzdroje, Poland, pp. 788–793 (2015)

15. Galindo R., Input/Output Decoupling of Square Linear Systems by Dynamic Two-Parameter Stabilizing Control, *Asian Journal of Control*, Vol. 18, No. 6, pp. 2310–2316 (2016)
16. Grune L., Pannek J., Nonlinear model predictive control, Springer Verlag, London (2007)
17. Haber R., Bars R., Schmitz U., *Predictive control in process engineering*, Weinheim, WILEY-VCH (2011)
18. Ben Hariz M., Bouani F., Synthesis and Implementation of a Robust Fixed Low-Order Controller for Uncertain Systems, *Arabian Journal for Science and Engineering*, Vol. 41, No. 9, pp. 3645–3654 (2016)
19. Huijberts H.J.C., Moog C.H., Pothin R., Input-output decoupling of nonlinear systems by static measurement feedback, *Systems & Control Letters*, Vol. 39, pp. 109–114 (2000)
20. Isidori A., *Nonlinear control systems*, New York, Springer-Verlag (1995)
21. Khalil H.K., *Nonlinear systems*, Prentice Hall (2001)
22. Koumboulis F.N., Skarpetis M.G., Input-output decoupling for linear systems with nonlinear uncertain structure, *Journal of the Franklin Institute*, Vol. 333(B), No. 4, pp. 593–624 (1996)
23. Koumboulis F.N., Skarpetis M.G., Output feedback decoupling of linear systems with nonlinear uncertain structure, *Journal of the Franklin Institute*, Vol. 333(B), No. 4, pp. 625–629 (1996)
24. Kueera V., Optimal decoupling controllers revisited, *Control and Cybernetics*, Vol. 42, No. 1, pp. 139–154, (2013)
25. Liu G., Wang Z., Mei C., Ding Y., A review of decoupling control based on multiple models, *24th Chinese Control and Decision Conference*, pp. 1077–1081 (2012)
26. Oblak S., Skrjanc I., Multivariable fuzzy predictive functional control of a MIMO nonlinear system, *Proc. of the 2005 IEEE International Symposium on Intelligent Control*, Limassol, Cyprus, pp.1029–1034 (2005)
27. Park K., H2 design of decoupling controllers based on directional interpolations, *Joint 48th IEEE CDC and 28th Chinese Control Conference*, pp. 5333–5338 (2009)
28. Pereira R.D.O., Veronesi M., Visioli A., Normey-Rico J.E., Torrico B.C., Implementation and test of a new autotuning method for PID controllers of TITO processes, *Control Engineering Practice*, Vol. 58, pp. 171–185 (2017)
29. Saniye A., Suleyman K., Decoupling constrained model predictive control of multi-component packed distillation column, *World Applied Science Journal*, Vol. 13, No. 3, pp. 517–530 (2011)
30. Schmitz U., Haber R., Arousi F., et al., Decoupling predictive control by error dependent tuning of the weighting factors, *Process Control Conference*, Sterbskie Pleso, pp. 131–140 (2007)
31. Wang X., Li S., Wang Z., Yue H., Multiple Models Neural Network Decoupling Controller for a Nonlinear System, *LNCS*, Vol. 3174, pp. 175–180 (2004)
32. Wang X., Yang H., Wang B., Multiple models fuzzy decoupling controller for a nonlinear systems, *LNAI*, Vol. 4223, pp. 860–863 (2006)
33. Wise D.A., English J.W., Tank and wind tunnel test for a drillship with dynamic position control, *Offshore Tehnology Conference*, TX Dallas (1975)
34. Zabert K., Haber R., Improvement of the decoupling effect of the predictive controller GPC and PFC by parameter adaptation, *18th International Conference on Process Control*, Tatranska Lomnica, Slovakia, pp. 419–426 (2011)
35. Zermani M.A., Feki E., Mami A., Self tuning weighting factor to decoupling control for incubator system, *International Journal of Information Technology, Control and Automation*, Vol. 2, No. 3., pp. 67–83 (2012)

Offset-Free Nonlinear Model Predictive Control

Piotr Tatjewski

Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warszawa, Poland
<P.Tatjewski@ia.pw.edu.pl>

Abstract. Offset-free model predictive control (MPC) for nonlinear state-space process models, with modeling errors and under asymptotically constant external disturbances, is the subject of the paper. A brief introduction of the MPC formulation used is first given, followed by a brief remainder of the case with measured state vector. The case with process outputs measured only and thus the necessity of state estimation is further considered. The main result of the paper is the presentation of a novel technique with process state estimation only, despite the presence of deterministic disturbances. The core of the technique is the state disturbance model used for the state prediction. It was introduced originally for linear state-space models and is generalized to the nonlinear case in the paper. This leads to a simpler design without the need for decisions of disturbance structure and placement in the model and to simpler (lower dimensional) control structure with process state observer only. Results of theoretical analysis of the proposed algorithm are provided, under applicability conditions which are weaker than in the conventional approach of extended process-and-disturbance state estimation. The presented theory is illustrated by simulation results of a nonlinear process.

1 Introduction

Among the advanced control techniques, the model predictive control (MPC) is now a well established general technology, see, e.g., [2], [18], [3], resulting in a variety of successful control techniques applied in practice, see, e.g., [4], [8], [15], [17], [2], [18], [25], [16], [19], [20], [3], [23]. The nonparametric models (step or impulse response models) and transfer function models lead to well established MPC structures, as DMC and GPC, respectively. On the other hand, the state-space modeling results in a variety of possibilities. There are different approaches to state-space modeling, as minimal and non-minimal models, extended velocity form models – leading to a different handling of deterministic disturbances. It should be also realized that points and ways these disturbances influence the process are also important for a controller design. The mentioned problem has attracted a rather limited attention in the literature, until the last decade [12], [14], [18], [13], [5], [10], [11], [21], [7]. However, there is still a certain lack of a clear understanding how the disturbances should be most effectively treated in the MPC algorithms with state-space models. We concentrate on this problem

in the paper, for the defined class of asymptotically constant deterministic disturbances. More general, continuously varying disturbances, like sinusoids, will be not considered, see [10] for offset-free MPC reference tracking under varying disturbances.

The main subject of the paper is to present possible techniques of state estimation for offset-free model predictive control (MPC) for nonlinear processes, under (asymptotically) constant set-point values and (asymptotically) constant disturbances entering the process at any point, possibly with white noises added. The considered class of disturbances, important e.g. in process control, includes modeling errors and external step or piecewise-constant disturbances changing rarely with respect to the controlled process dynamics. The conventional technique of extended state estimation leading to the process-and-disturbance state model is first briefly recalled [5], [11]. Main contribution of the paper is to present a new technique with estimation of the process state only, which generalizes the results obtained earlier by the author for linear state-space process models [18], [21], to the case of nonlinear models. The proposed approach leads to a simpler and more general MPC control structure than the mentioned conventional approach, with weaker applicability conditions.

The structure of the paper is as follows. In Section 2 the MPC is briefly reviewed, to introduce formulations needed for further considerations. In Section 3 process models further used are presented. The case with measured state is also briefly recalled, referring to the approach originally presented in [22], resulting in a MPC formulation where the observer/estimator of the considered deterministic disturbances is not needed at all. In Section 4 the most general case with state estimation is treated and the main result of the paper is presented. This results in simpler nonlinear MPC control structure and simpler design as well, under significantly weaker applicability conditions. In Section 5, the proposed MPC approach is illustrated by simulations of the control structure with a nonlinear example process model, taken from the literature. Finally, conclusions are formulated.

2 Predictive Control Briefly Recalled

The principle of MPC is now well known and different descriptions and algorithms can be found in many papers and books, including those with discrete-time state-space process models we are interested in. In the books, see [8], [17], [18], [25], [16], mainly linear process models, of different types, are considered. Nonlinear process models used in the MPC algorithms are in the form of standard nonlinear state-space models or nonlinear difference equations of higher orders, including neural network models, see, e.g., [24], [6].

We shall now briefly recall the MPC formulation which will be needed for further presentation. The principle of the MPC is to evaluate the current control signal by minimizing, at each sampling instant k , a performance function (cost function) over a future prediction horizon of N samples. The following performance function is one of the most widely used in process control:

$$J(k) = \sum_{p=1}^N \|[y^{sp}(k+p|k) - y(k+p|k)]\|_{\Psi}^2 + \sum_{p=0}^{N_u-1} \|\Delta u(k+p|k)\|_{\Lambda}^2, \quad (1)$$

where $\|x\|_{\mathbf{R}}^2 = x^T \mathbf{R} x$, $\Psi \succeq \mathbf{0}$ and $\Lambda \succ \mathbf{0}$ are square diagonal scaling matrices of dimensions n_y and n_u of the process controlled output and control input vectors, respectively (a simpler formulation of (1) is often used in theoretical considerations, with one scaling scalar λ only, i.e., $\Psi = \mathbf{I}$ and $\Lambda = \lambda \mathbf{I}$). In the above, $N_u \leq N$ denotes the length of the control horizon, $y^{sp}(k+p|k)$ and $y(k+p|k)$ are set-point (reference) and output vectors predicted for a future sample $k+p$, but calculated at the current sample k , $p = 1, \dots, N$. The decision variables are control input increments on the control horizon, $\Delta u(k+p|k) = u(k+p|k) - u(k+p-1|k)$, $p = 0, \dots, N_u - 1$. The performance function (1) yields a unique solution if $n_u = n_y$, which will be assumed in the paper (without loss of generality).

We assume that the optimization of $J(k)$ is subject to simple constraints:

$$-\Delta u_{max} \leq \Delta u(k+p|k) \leq \Delta u_{max}, \quad p = 0, \dots, N_u - 1, \quad (2)$$

$$u_{min} \leq u(k+p|k) \leq u_{max}, \quad p = 0, \dots, N_u - 1, \quad (3)$$

$$y_{min} \leq y(k+p|k) \leq y_{max}, \quad p = 1, \dots, N. \quad (4)$$

More general form of the constraints, including any linear functions of all variables used, is possible, but avoided here for simplicity.

Denote the vector of the decision variables by $\Delta U(k)$,

$$\Delta U(k) = [\Delta u(k|k)^T \quad \Delta u(k+1|k)^T \quad \dots \quad \Delta u(k+N_u-|k)^T]^T, \quad (5)$$

and denote composite vectors of set-points and predicted outputs on the prediction horizon by $Y^{sp}(k)$ and $Y^{pr}(k)$, respectively,

$$Y^{sp}(k) = [y^{sp}(k+1|k)^T \quad \dots \quad y^{sp}(k+N|k)^T]^T, \quad (6)$$

$$Y^{pr}(k) = [y(k+1|k)^T \quad \dots \quad y(k+N|k)^T]^T. \quad (7)$$

We can then formulate, in a compact form, *the MPC optimization problem* which calculates the optimal control trajectory:

$$\begin{aligned} \min_{\Delta U(k)} \{ & J(k) = \|Y^{sp}(k) - Y^{pr}(k)\|_{\Psi}^2 + \|\Delta U(k)\|_{\Lambda}^2 \} \\ & \text{subject to (2), (3) and (4),} \end{aligned} \quad (8)$$

where

$$\Psi = \text{diag} \left\{ \overbrace{\Psi, \dots, \Psi}^{N \text{ times}} \right\}, \quad \Lambda = \text{diag} \left\{ \overbrace{\Lambda, \dots, \Lambda}^{N_u \text{ times}} \right\}, \quad (9)$$

and the predicted output trajectory $Y^{pr}(k)$ is calculated using the process model. When this model is nonlinear, then the optimization problem (8) is also nonlinear and nonlinear optimization procedures must be applied.

When using linear process models, the superposition principle can be applied. The predicted trajectory of the outputs $Y^{pr}(k)$ can be then split into a sum of

a "forced trajectory" $Y^+(k) = \mathbf{M} \Delta U(k)$, being a linear mapping of the vector of decision variables only (consisting of elements $y^+(k+p|k), p = 1, \dots, N$) and a "free trajectory" $Y^0(k)$ (consisting of elements $y^0(k+p|k), p = 1, \dots, N$), depending on current and past data only and obtained with the control input frozen over the prediction horizon on the last applied value $u(k-1)$. With a linear process model, the MPC optimization problem (8) is a strictly convex quadratic programming (QP) problem, thus with a well defined, unique solution – provided the set defined by the constraints assures feasibility (is non-empty). For a detailed description of the MPC algorithms with linear process models, see, e.g., [18].

Once the MPC optimization problem has been solved, the first element $\Delta u(k|k) = \Delta u(k)$ of the control trajectory is used only and the input $u(k) = u(k-1) + \Delta u(k)$ is applied to the process. After the next measurement (at the next sampling instant) the whole procedure is repeated (receding horizon strategy).

3 Process description and modeling

The following nonlinear process description will be used:

$$x(k+1) = f_p(x(k), u(k), d_p(k)), \quad (10a)$$

$$y(k) = g(x(k)), \quad (10b)$$

where x denotes the process state vector, $\dim x = n_x$, y the controlled output vector, $\dim y = n_y$, u the control (control input) vector, $\dim u = n_u$ and d_p represents unknown, unmeasured disturbances (including modelling errors), $\dim d_p = n_{d_p}$. Measured disturbances will not be explicitly considered in the paper, for the sake of simplicity.

The following process model is assumed to be known:

$$x(k+1) = f(x(k), u(k)), \quad (11a)$$

$$y(k) = g(x(k)). \quad (11b)$$

For state prediction in the MPC algorithm, the model (11) will be augmented to the form

$$x(k+1) = f(x(k), u(k)) + v(k), \quad (12a)$$

$$y(k) = g(x(k)), \quad (12b)$$

where $v(k)$ represents influence of unmeasured disturbances on the state vector, $\dim v = n_x$ [18, 22].

The key factor of the presented MPC approach is the way $v(k)$ is modeled for the use in (12a). The *constant state disturbance prediction*, originally proposed in [18] for the MPC with linear state-space process models (see also [21, 22]), is generalized to the nonlinear case. In is defined as follows

$$v(k) = x(k) - f(x(k-1), u(k-1)), \quad (13)$$

$$v(k|k) = v(k+1|k) = v(k+2|k) = \dots = v(k+N-1|k) = v(k), \quad (14)$$

where $x(k)$ denotes the *process state* at time k , which can be measured or estimated.

Having the state measured and using the process model (12) with the disturbance model (13)-(14), the state and thus output predictions are calculated. Then the nonlinear MPC algorithm solves at each sampling instant k the nonlinear MPC optimization problem (8), by a nonlinear optimization procedure with the vector of decision variables $\Delta U(k)$ defined by (5). The optimization procedure iterates the decision variables, starting from an initial vector $\Delta U^{(0)}(k)$ and calculating next (improved) values $\Delta U^{(i)}(k)$, $i=1,2,3,\dots$ etc., until the optimality criterion is fulfilled. For each new value $\Delta U^{(i)}(k)$ calculation of the output predictions $y^{(i)}(k+p|k)$ over the control horizon (for $p = 1, \dots, N$) is needed.

The briefly described nonlinear MPC algorithm with measured process state, called further the *Algorithm NMPC1*, has been proposed and analysed in [22]. The essence and novelty of the algorithm is that the controller assures the offset-free control without the necessity to use an observer/estimator of the deterministic disturbances. The key element of this algorithm is *the technique of state disturbance modeling*.

Notice that $g(x(k))$ is assumed to be a known nonlinear function of the state (in most cases it is linear). In practice it is almost always true. If it would not be the case, additional measure must be taken to assure an offset-free control, see [22].

4 Offset-free nonlinear MPC with state estimation

In cases when the process state needs estimation, a conventional technique for offset-free control in the case with linear model used is the addition of a disturbance state model and an extended state (process-and-disturbance state) estimation, see [12], [14], [5], [9], [21]. The analysis is here usually made for deterministic state observers, as it is simpler and all results can be easily generalized to the stochastic case with Kalman filtering. An extension of this approach to the nonlinear case can be found, e.g., in [11]. The MPC controller designed according to this more conventional technique does not use the disturbance model (13)-(14), instead the modeled disturbance estimate $\hat{d}(k)$ is appropriately used in state and output predictions. The disadvantage of this technique, both in the linear and, especially, in the nonlinear case, is that the designer must define the quantity (dimensionality) n_d and placement of the disturbances in the model, under the restriction $n_d \leq n_y$.

The technique proposed in this section is the main result of the paper. On one hand, it is a further development of the technique briefly recalled in Section 3 for the case with measured process state, on the other hand it is a generalisation of the approach presented in [21], to the nonlinear case.

When the process state is not measured, application of a state observer is necessary. For the process (10) modeled by (11), a simple and straightforward deterministic approach to the observation of the process state is to use the

extended Luenberger observer (ELO):

$$\hat{x}(k) = f(\hat{x}(k-1), u(k-1)) + \mathbf{L}[y(k-1) - g(\hat{x}(k-1))], \quad (15)$$

where $\hat{x}(k)$ is the state estimate and \mathbf{L} is the observer gain matrix. Design and analysis of the extended Luenberger observer for nonlinear processes is not so easy as for linear systems, even in the case without a process-model mismatch, see, e.g., [1]. The main requirement is the local observability of the model in the region of operation of the feedback control system, and the simplest method to design the gain matrix \mathbf{L} is to do it as for a linear Luenberger observer, for a linearization of the model (11). The observer (15) is in the predictive form, more practical is the current observer form:

$$\hat{x}(k) = f(\hat{x}(k-1), u(k-1)) + \mathbf{L}_c[y(k) - g(f(\hat{x}(k-1), u(k-1)))], \quad (16)$$

The analysis of nonlinear observers is out of scope of this paper, we shall further assume that the observer can be successfully designed (in the next section, design details will be given for an example process). Any kind of a more elaborate observer, providing asymptotically stable estimation error in a working region of interest, can be also applied. An extended Kalman filter (EKF) may be a choice, especially when the process state and outputs are under influence of noises.

The main new feature of the proposed nonlinear MPC algorithm is that, despite modeling errors and external disturbances, the process state only is estimated by the observer (the process state is not augmented). The algorithm will be denoted *Algorithm NMPC2*. Like the *Algorithm NMPC1*, it solves at each sampling instant k the nonlinear MPC optimization problem (8).

Algorithm NMPC2

1. The process outputs $y(k)$ are measured. The process state estimate $\hat{x}(k)$ is calculated by the nonlinear observer (15).
2. The state disturbance prediction $v(k)$ is calculated:

$$v(k) = \hat{x}(k) - f(\hat{x}(k-1), u(k-1)). \quad (17)$$

3. The optimization problem (8) is solved, iteratively by a nonlinear optimization procedure, which calculates at its every (i -th) internal iteration the output predictions in the following way:

3a. Using the model (12), state predictions $x^{(i)}(k+p|k)$ are recursively calculated over the control horizon (for $p = 1, \dots, N$):

$$x^{(i)}(k+p+1|k) = f(x^{(i)}(k+p|k), u^{(i)}(k+p|k)) + v(k), \quad (18)$$

where $x^{(i)}(k|k) = \hat{x}(k)$ is the estimated state,

$$u^{(i)}(k+p|k) = u(k-1) + \sum_{j=0}^{i-1} \Delta u^{(i)}(k+p|k) \quad \text{for } p < N_c, \quad (19)$$

$$u^{(i)}(k+p|k) = u^{(i)}(k+N_c-1|k) \quad \text{for } p \geq N_c. \quad (20)$$

3b. The output predictions are calculated:

$$y^{(i)}(k+p|k) = g(x^{(i)}(k+p|k)) + [y(k) - g(\hat{x}(k))], \quad p = 1, \dots, N. \quad (21)$$

4. After the optimal trajectory $\Delta \hat{U}(k)$ is found by the optimization procedure, its first element $\Delta \hat{u}(k|k)$ defines the current control input signal $u(k) = u(k-1) + \Delta \hat{u}(k|k)$, which is sent to the process actuators.
5. The algorithm waits for the next sampling, then repeats from **1**.

It should be noted that state predictions (18) incorporate *the state disturbance prediction* $v(k)$ and the output predictions are calculated, in step (3b), *by a non-standard formula* (21) which includes the newly introduced *correction term* $y(k) - g(\hat{x}(k))$.

The following Theorem 1, which formulates conditions of offset-free property of the feedback control system with the Algorithm NMPC2, can be proved (the proof is omitted due to limited space).

Theorem 1. *Assume that:*

1. *Disturbances affecting the process (10a)-(10b) are asymptotically constant, stabilizing at a certain value d_{ss} , the set-point (reference) values are asymptotically constant, stabilizing at a value y_{ss}^{sp} .*
2. *Set-point values y_{ss}^{sp} are feasible and steady-state controllable for the disturbance values d_{ss} , i.e., y_{ss}^{sp} satisfies the output constraints (4) and there is a feasible control signal u (i.e., satisfying the constraints (3)) resulting in the process output $y(u) = y_{ss}^{sp}$ in steady-state, under the disturbance values d_{ss} .*
3. *The MPC optimization problem (8) with output predictions as defined by (18)-(21) in the NMPC2 algorithm, is feasible for every k (i.e., the set defined by the constraints (2)-(4) is not empty) and the feedback control system consisting of the process (10a)-(10b) and the Algorithm NMPC2 (including the process state observer (15)) is asymptotically stable.*

Then the feedback control system defined in assumption (3) provides an offset-free control, i.e., the process outputs stabilize at the set-point values y_{ss}^{sp} , despite the influence of the disturbances as defined in assumption (1).

The key difference between the algorithm with measured state (Algorithm NMPC1) and the Algorithm NMPC2 is the introduction of the correction term $y_{ss} - g(\hat{x}_{ss})$ in the final prediction equation (21). Without this term, the process outputs would stabilize at the value $y_{ss} = g(\hat{x}_{ss})$, in general not equal to the value y_{ss}^{sp} , as in general $\hat{x}_{ss} \neq x_{xx}$. This is due to the influence of deterministic disturbances and is true also in the linear case. The reason is that at a steady-state the process equations $x_{ss} = f_p(x_{ss}, u_{ss}, d_{ss})$ and $y_{ss} = g(x_{ss})$ must be fulfilled, with results in the values (x_{ss}, u_{ss}) . On the other hand, the steady-state estimate value \hat{x}_{ss} is a fixed point of the observer equation (for the obtained values x_{ss} and u_{ss})

$$\hat{x}_{ss} = f(\hat{x}_{ss}, u_{ss}) + \mathbf{K}(y_{ss} - g(\hat{x}_{ss})) \quad (22)$$

which is not equal to x_{ss} , in general, $x_{ss} \neq \hat{x}_{ss}$. This occurs also in the conventional approach with the extended process-and-disturbance state estimation, due to differences in numbers and locations of disturbances in the process equations and estimated disturbances placed by the designer in the model equations (e.g., see results of simulations presented in [11]). If $f_p(x_{ss}, u_{ss}, d_{ss}) \neq f(\hat{x}_{ss}, u_{ss})$, then we have in steady-state a non-zero correction term in the observer equation, e.g., for the observer (15), $y_{ss} - g(\hat{x}_{ss}) \neq 0$.

Comparing the main differences between the proposed technique of Algorithm NMPC2 and the more conventional approach with the extended process-and-disturbance state estimation [11], the advantage of the proposed one is an easier design, without the necessity of a careful choice and placement of estimated deterministic disturbances in the model (under the restriction $n_d \leq n_y$!). Another advantage is a simpler controller structure, with the observer of the process state only (not augmented). On the other hand, the Algorithm NMPC2 applies to output control only, i.e., with the main part of the MPC performance function being the sum of squared output control errors (see (1)), not squared differences between current and desired (referenced) states of the process, as in [11]. This is due to the fact that only the outputs are measured, thus the correction term can be for the outputs only.

5 Simulation Example

The example given below is taken from [11], to enable certain comparison with the technique of extended process-and-disturbance state modeling and estimation applied there.

A nonlinear (bilinear) SISO process with the following state-space description is considered:

$$x_{p1}(k+1) = 0.95x_{p1}(k) - 0.25x_{p1}(k)x_{p2}(k) + x_{p2}(k) + d_1(k), \quad (23a)$$

$$x_{p2}(k+1) = 0.7x_{p2}(k) + 0.1x_{p2}(k)d_2(k) + u(k), \quad (23b)$$

$$y_p(k) = x_{p1}(k) + d_3(k), \quad (23c)$$

where $x_p(k) = [x_{p1}(k) \ x_{p2}(k)]^T$ is the system state, $d_1(k)$, $d_2(k)$ are unknown (unmeasured) disturbances entering the state equations and $d_3(k)$ is an unknown (unmeasured) pure output disturbance. The presented example process is taken from [11], but it is with reacher structure of disturbances which can be mutually independent, whereas in [11] $d_1(k) = d_2(k) = d_3(k) = d(k)$ had to be assumed (i.e., one disturbing signal), due to the restriction $n_d \leq n_y$.

The following nonlinear model of the process is used for the controller design:

$$x_1(k+1) = 0.9x_1(k) - 0.3x_1(k)x_2(k) + x_2(k), \quad (24a)$$

$$x_2(k+1) = 0.8x_2(k) + u(k), \quad (24b)$$

$$y(k) = x_1(k), \quad (24c)$$

where $x(k) = [x_1(k) \ x_2(k)]^T$, $y(k)$ are the model state and output, respectively.

The MPC controller was designed implementing the algorithm NMPC2. The appropriate prediction horizon was found to be $N = 8$, and the control horizon was set to $N_u = 4$. Only one scalar weighting coefficient $\lambda = 0.5$ was assumed in the performance function, i.e., $\Psi = \mathbf{I}$ and $\Lambda = \lambda\mathbf{I}$ in (1).

The simulation scenario contained step changes of the set-point and all unmeasured disturbances. The simulation starts at the zero equilibrium point. Then at $k = 1$ the set-point changes to -1 , at $k = 20$ jumps from -1 to $+1$ and at $k = 40$ returns to 0 . At $k = 60$ all disturbances jump from 0 to 0.3 , 0.4 and 0.5 , respectively; at $k = 80$ $d_1(k)$ returns to 0 , at $k = 100$ $d_2(k)$ returns to 0 and at $k = 120$ $d_3(k)$ jumps from 0.5 to 0.3 – the disturbance behaviour is shown in the lowest plot in Fig. 1.

The ELO observer in the current form, with constant gains, was found to be satisfactory for the example system. It was designed for the linear state-space model being linearization of the nonlinear model (24) at the origin. The gain matrix of the current observer \mathbf{L}_c was designed in a standard way, first designing the gain matrix \mathbf{L} for the predictive observer, for a predefined set of eigenvalues for the closed loop, and then recalculated for the current observer using the formula $\mathbf{L}_c = \mathbf{A}^{-1}\mathbf{L}$ (where \mathbf{A} is the state matrix of the linearized model).

A representative result, chosen after several simulations with the Algorithm NMPC2, is shown in Fig. 1. The presented results show very good performance of the proposed algorithm for the considered example. Both the modeling inaccuracies (compare coefficients and structures of (23a)-(23c) and (24a)-(24c)) and step changes of many different unknown (unmeasured) external disturbances are well attenuated, providing offset-free control.

For a comparison, the MPC nonlinear controller designed according to conventional technique of extended process-and-disturbance state estimation was also tested. Since the example system is a SISO one, this design has to be under the restriction $\dim d(k) = n_d = 1$. First, $d(k)$ was chosen as an output disturbance, i.e., for the state prediction the following process model was used:

$$x_1(k+1) = 0.9x_1(k) - 0.3x_1(k)x_2(k) + x_2(k), \quad (25a)$$

$$x_2(k+1) = 0.8x_2(k) + u(k), \quad (25b)$$

$$y(k) = x_1(k) + d(k), \quad (25c)$$

and the extended state vector was $[x(k) \ d(k)]^T$. This trial choice was a complete failure for the considered example, the extended state observer was not able to estimate the extended state in a way sufficient for a proper controller action. On the other hand, the design occurred to be most successful for the considered example with $d(k)$ assumed to be the process input disturbance, i.e., for the

following model used for the state prediction:

$$x_1(k+1) = 0.9x_1(k) - 0.3x_1(k)x_2(k) + x_2(k), \quad (26a)$$

$$x_2(k+1) = 0.8x_2(k) + u(k) + d(k), \quad (26b)$$

$$y(k) = x_1(k). \quad (26c)$$

The simulation results obtained in this case were similar to these with the NMPC2 algorithm (and not better). Since also in this case there is significant difference between disturbance structure in the process (23) and its model (26), trajectories of the state estimates significantly differed from the true state trajectories, similarly as it was for the NMPC2 algorithm (see Fig. 1).

For a representative result of the simulations with measured state (Algorithm NMPC1) the reader is referred to [22].

6 Conclusions

New approach to offset-free model predictive control with nonlinear state-space process models has been presented in the paper, for deterministic asymptotically constant unmeasured external and internal disturbances (modeling errors) and asymptotically constant set-points (reference values). The presented control structure is derived for the output control, i.e. for MPC formulations with squared output control errors being the main part of the performance function. The presented approach was originally proposed by the author for linear state-space process models. In the paper, the formulation for nonlinear models is derived. The basic element is an appropriate formulation of state disturbances in the model used for the state and output prediction (a constant state disturbance model). This leads to a simpler control structure, with the process state estimation only, as opposed to the conventional approach with the process state-and-disturbance vector estimation. Moreover, the design is simpler – due to the prescribed disturbance model the necessity of a choice of a limited number of disturbances and their placement in the equations of the process model is avoided. Results of theoretical analysis of the proposed algorithm are reported, under the applicability conditions that are weaker than in the conventional approach. However, it should be pointed out that the presented approach is based on the classical and most practical formulation of the MPC performance function based on predicted control errors and process input increments, whereas in the approach discussed in [11], process state and input deviations from appropriate steady-state targets are used. Theoretical results presented in the paper are illustrated by simulations with a nonlinear process, taken from the literature, to enable a comparison with the conventional approach. Comparing the presented theoretical and simulation results with those for the technique with extended process-and-disturbance state estimation, we can conclude that the proposed algorithm is competitive, broadening the range of possible solutions available for the design engineer.

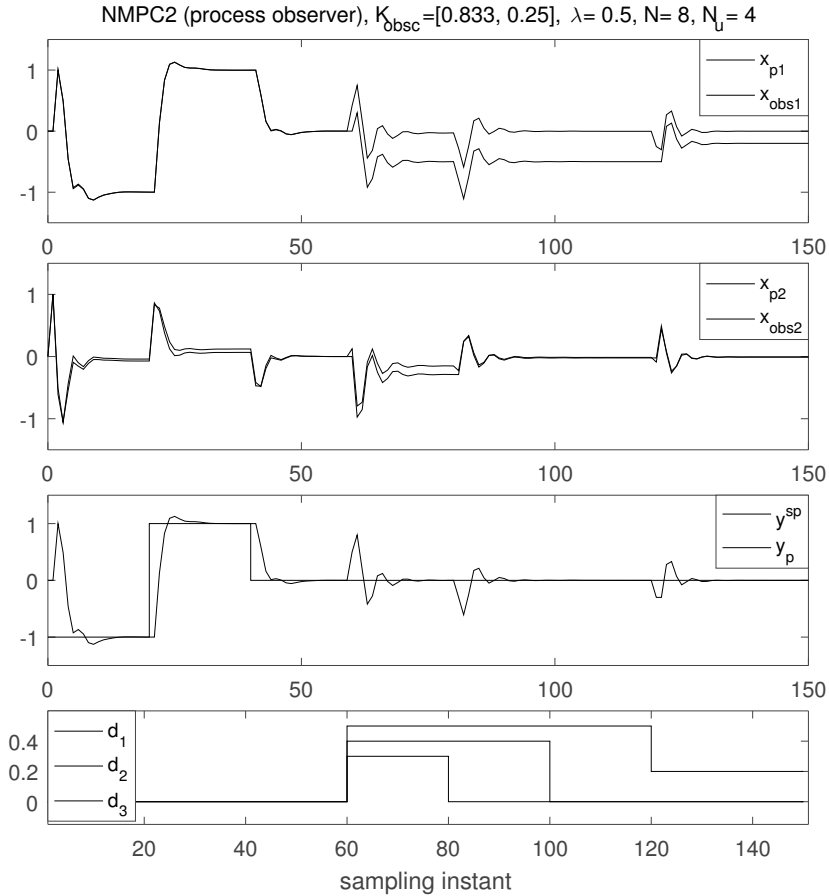


Fig. 1. Algorithm NMPC2: trajectories of the process states and state estimates, set-point and measured output, unmeasured disturbances.

References

1. Birk, J., Zeitz, M.: Extended Luenberger observer for non-linear multivariable systems. *International Journal of Control* 47(6), 1823–1835 (1988)
2. Blevins, T.L., McMillan, G.K., Wojsznis, W.K., Brown, M.W.: *Advanced Control Unleashed*. The ISA Society, Research Triangle Park, NC (2003)
3. Blevins, T.L., Wojsznis, W.K., Nixon, M.: *Advanced Control Foundation*. The ISA Society, Research Triangle Park, NC (2013)
4. Camacho, E., Bordons, C.: *Model Predictive Control*. Springer Verlag, London (1999)
5. Gonzalez, A.H., Adam, E.J., Marchetti, J.L.: Conditions for offset elimination in state space receding horizon controllers: A tutorial analysis. *Chemical Engineering and Processing* 47, 2184–2194 (2008)

6. Lawryńczuk, M.: *Computationally Efficient Model Predictive Control Algorithms: A Neural Network Approach*, Studies in Systems, Decision and Control, Vol. 3. Springer Verlag, Heidelberg (2014)
7. Lawryńczuk, M.: Nonlinear state-space predictive control with on-line linearisation and state estimation. *International Journal of Applied Mathematics and Computer Science* 25(4), 833–847 (2015)
8. Maciejowski, J.: *Predictive Control*. Prentice Hall, Harlow, England (2002)
9. Maeder, U., Borelli, F., Morari, M.: Linear offset-free model predictive control. *Automatica* 45, 2214 – 2222 (2009)
10. Maeder, U., Morari, M.: Offset-free reference tracking with model predictive control. *Automatica* 46, 1469–1476 (2010)
11. Morari, M., Maeder, U.: Nonlinear offset-free model predictive control. *Automatica* 48, 2059–2067 (2012)
12. Muske, K., Badgwell, T.: Disturbance modeling for offset-free linear model predictive control. *Journal of Process Control* 12, 617–632 (2002)
13. Pannocchia, G., Bemporad, A.: Combined design of disturbance model and observer for offset-free model predictive control. *IEEE Transactions on Automatic Control* 52(6), 1048–1053 (2007)
14. Pannocchia, G., Rawlings, J.: Disturbance models for offset-free model predictive control. *AIChE Journal* 49(2), 426–437 (2003)
15. Qin, S., Badgwell, T.: A survey of industrial model predictive control technology. *Control Engineering Practice* 11, 733–764 (2003)
16. Rawlings, J.B., Mayne, D.Q.: *Model Predictive Control: Theory and Design*. Nob Hill Publishing, Madison (2009)
17. Rossiter, J.: *Model-Based Predictive Control*. CRC Press, Boca Raton - London - New York - Washington, D.C. (2003)
18. Tatjewski, P.: *Advanced Control of Industrial Processes*. Springer Verlag, London (2007)
19. Tatjewski, P.: Advanced control and on-line process optimization in multilayer structures. *Annual Reviews in Control* 32, 71–85 (2008)
20. Tatjewski, P.: Supervisory predictive control and on-line set-point optimization. *International Journal of Applied Mathematics and Computer Science* 20(3), 483–496 (2010)
21. Tatjewski, P.: Disturbance modeling and state estimation for offset-free predictive control with state-spaced process models. *International Journal of Applied Mathematics and Computer Science* 24(2), 313–323 (2014)
22. Tatjewski, P.: Offset-free nonlinear predictive control with measured state and unknown asymptotically constant disturbances. In: Malinowski, K., Józefczyk, J., Świątek, J. (eds.) *Aktualne problemy automatyki i robotyki (Actual problems in automation and robotics)*, pp. 288–299. Akademicka Oficyna Wydawnicza Exit, Warszawa (2014)
23. Tatjewski, P.: *Sterowanie zaawansowane procesów przemysłowych (Advanced Control of Industrial Processes)*, Second edition, revised (e-book, in Polish). EXIT Academic Publishers, Warszawa (2016)
24. Tatjewski, P., Lawryńczuk, M.: Soft computing in model-based predictive control. *International Journal of Applied Mathematics and Computer Science* 16(1), 7–26 (2006)
25. Wang, L.: *Model Predictive Control System Design and Implementation using MATLAB*. Springer Verlag, London (2009)

Damping of the pendulum during dynamic stabilization in arbitrary angle position

Maciej Ciężkowski

Białystok University of Technology
Białystok, Poland
m.ciezkowski@pb.edu.pl

Abstract. The paper presents an approach to dumping of the pendulum during dynamic stabilization in an arbitrary angle. The rapid oscillations of the pendulum's suspension point cause that created effective potential has a local minimum which guarantees the stability of the pendulum. The external disturbances bring an additional energy into the system and the pendulum increases the amplitude of oscillations around its equilibrium position. The aim of this paper is to describe the damping method of this excess oscillations during dynamic stabilization of the pendulum.

Keywords: dynamic stabilization, pendulum, damping of the pendulum, LQR control

1 Introduction

Over a hundred years ago Andrew Stephenson discovered that the rapid oscillations of the inverted pendulum's suspension point in the vertical direction stabilized the pendulum in vertical position [1, 2]. In 1932, Lowenstern [3] determined the equation of motion for pendulum exposed to fast oscillations but did not explain why such a system is stable. About twenty years later, Peter Kapitza [4] gave an explanation of this phenomenon using concept of an effective potential: fast oscillations of the pendulum's suspension point in vertical direction will lead to the formation of the effective potential, and the minimum of this potential is for the pendulum's vertical position. Method discovered by Peter Kapitza is also used to describe some of the processes as well as in atomic and quantum physics [5, 6], as in the control theory [7–12].

The most common cases of dynamic stabilization of the pendulum described in the literature, concern the stabilization of the inverted pendulum by vertical oscillations of the pendulum's suspension point. It turns out that it is possible to generalize the problem and demonstrate the possibility of stabilization of the pendulum in an arbitrary position by oscillating the pendulum's suspension point at the appropriate angle [13, 14]. In [15] is shown an approach to the dynamic stabilization of the pendulum when the desired angle of the pendulum's position changed in time. This is achieved by controlling the angle of oscillations of the pendulum's suspension point, and so it becomes possible to carry out a pendulum from one stable position to another.

Due to the fact that the force driving the pendulum's suspension point during dynamic stabilization is a conservative force, the equilibrium position of the pendulum is Lyapunov stable but not asymptotically stable position, see for example [16]. The pendulum deflected from its equilibrium position starts to oscillate around this position because of a lack of energy dissipation. In practice, it is usually important to achieve asymptotically stable equilibrium state, therefore some control technique to dampen these oscillations should be used. The goal of the presented studies is to demonstrate a dumping control method of the pendulum in the case of its dynamic stabilization.

2 Mathematical model

As the model of the pendulum, the perfectly rigid rod with mass m and length l has been taken. The one end of the pendulum is the point of suspension. The system is placed in a gravitational field g . The pendulum is allowed to move only in the xy plane. The pendulum inclination angle with respect to the y -axis is denoted as θ , see Fig. 1.

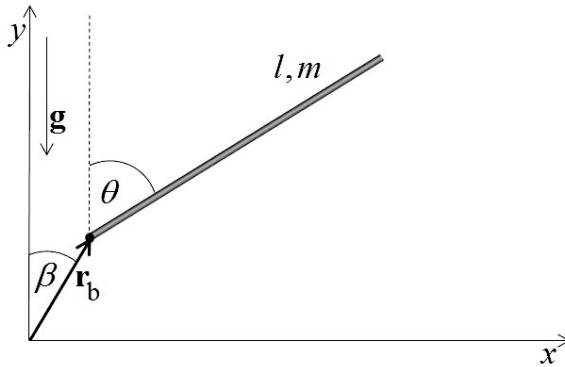


Fig. 1. Physical model of pendulum.

The position of the pendulum's suspension point describes vector:

$$\mathbf{r}_b = (A \cos(\Omega t) \sin(\beta), A \cos(\Omega t) \cos(\beta)) \quad (1)$$

where A is the amplitude of vibrations of the suspension point, Ω - the frequency of vibrations of the suspension point, β - the direction angle of vibrations of the suspension point. Oscillating change of the position of the pendulum's suspension point realizes the pendulum control. Using the Lagrange formalism, the equation of motion of the pendulum was obtained [14]:

$$\ddot{\theta} = \frac{3(A\Omega^2 \sin(\beta - \theta) \cos(t\Omega) + g \sin(\theta))}{2l} \quad (2)$$

Figure 2 shows the numerical results for the time evolution of the pendulum's angle. This is a solution of the equation (2) for values: $l = 0.5$ m, $A = 0.05$ m, $\Omega = 80$ rad/s, $g = 9.81$ m/s², $\beta = 0$ rad, $\theta(0) = 0.3$ rad, $\dot{\theta}(0) = 0$ rad/s

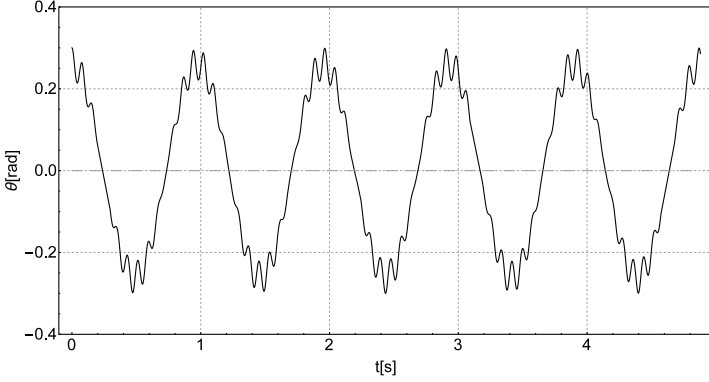


Fig. 2. Simulation results for the time evolution of the pendulum's angle.

3 Stability analysis

According to results presented in Fig. 2, the swing of the pendulum is composed of two vibrations:

$$\theta(t) = \phi(t) + \xi(t) \quad (3)$$

Function $\phi(t)$ describes high amplitude and low frequency oscillations, $\xi(t)$ is small oscillations with high frequency. Thus a function $\phi(t)$ describes the "smooth" movement of the pendulum, averaged due to the rapid oscillations. According to the procedure presented in [14], that is obtained by substituting (3) into (2) and expanding the result in the first-order Taylor series with respect to the $\xi(t)$ (small oscillations). Then, the equation describing the "smooth" motion of the pendulum is obtained as following:

$$\ddot{\phi} = \frac{9A^2\Omega^2 \sin(2(\beta - \phi))}{16l^2} + \frac{3g \sin(\phi)}{2l} \quad (4)$$

Due to the fact that the force driving the pendulum's suspension point is a conservative force, one can write relationship between force and effective potential energy as:

$$F = -\nabla U_{ef} \quad (5)$$

where U_{ef} is the effective potential energy. In the case of the considered pendulum, the equation (5) can be written as:

$$\frac{1}{3}ml^2\ddot{\phi} = -\frac{dU_{ef}}{d\phi} \quad \Rightarrow \quad U_{ef} = -\frac{1}{3}ml^2 \int \ddot{\phi} d\phi.$$

Then taking into account the equation (4), the effective potential energy of the pendulum takes the form:

$$U_{ef} = \frac{1}{2}glm \cos(\phi) - \frac{3}{32}A^2m\Omega^2 \cos(2(\beta - \phi)) \quad (6)$$

The Fig. 3 illustrates the effective potential of the pendulum for different values of angle β and fixed values of parameters: $l = 0.5$ m, $A = 0.05$ m, $\Omega = 80$ rad/s, $g = 9.81$ m/s², $m = 0.1$ kg. As shown in Fig. 3, each graph of U_{ef} has a

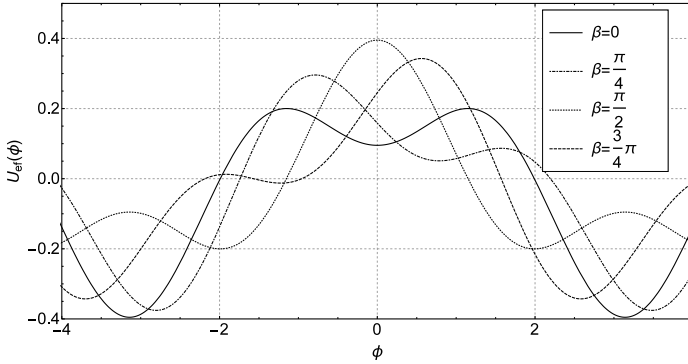


Fig. 3. Effective potential of the pendulum for various values of β .

local minimum of the potential and thus it satisfies the condition of the stability. The condition for a minimum of the effective potential energy takes the form [14]:

$$\begin{cases} \beta = \beta_{ext} = \frac{1}{2}(2\phi - \arcsin(\frac{2\sin(\phi)}{\lambda})) \\ \lambda > \sqrt{4\sin^2(\phi) + \cos^2(\phi)} & \text{for } \phi \in \langle 0, \frac{\pi}{2} \rangle \\ \lambda \geq 2\sin(\phi) & \text{for } \phi \in (\frac{\pi}{2}, \pi) \end{cases} \quad (7)$$

where $\lambda = \frac{3A^2\Omega^2}{4gl}$. The parameter λ , which determines the stability of the pendulum is a function of variables describing the controlled object (the pendulum's length l) and variables controlling the pendulum (A, Ω). The equation (7) shows that for any angle ϕ in $\langle 0, \pi \rangle$, the stability condition of the pendulum can be written as $\lambda > 2$. This condition for λ is fulfilled for the example values: $l = 0.5$ m, $A = 0.05$ m, $\Omega = 80$ rad/s, $g = 9.81$ m/s², for which the parameter $\lambda = 2.44648$. The above-mentioned example values and the pendulum mass $m = 0.1$ kg will be used in further analysis of the system.

If the parameter λ is constant in the experiment (and of course satisfies the stability conditions), the only problem to solve is to define the angle ϕ at which we want to stabilize the pendulum and then, according to (7), determine the angle β_{ext} which is the direction of vibration of the pendulum's suspension point.

For example, if the pendulum has to be stable at position $\phi = \pi/4$, direction of vibrations of the pendulum's suspension point according to (7) should be equal:

$$\beta = \beta_{ext} = \frac{1}{2} \left(2 \pi/4 - \arcsin \left(\frac{2 \sin(\pi/4)}{\lambda} \right) \right) = 0.477 \text{ rad.}$$

Figure 4 and 5 show the simulation results of the pendulum motion for $\lambda=2.44648$, $\theta(0) = \pi/4+0.3$, $\dot{\theta}(0) = 0 \text{ rad/s}$ and $\beta = 0.477 \text{ rad}$. As can be seen, the pendulum inclined to a certain angle starts to oscillate around the fixed point, in this case equal to $\pi/4$. The pendulum oscillates around this point because for $\theta(0) = \pi/4$ function (6) has a minimum, and the pendulum moves in a potential well around the equilibrium state. As can be seen in Fig. 4 and Fig. 2, the

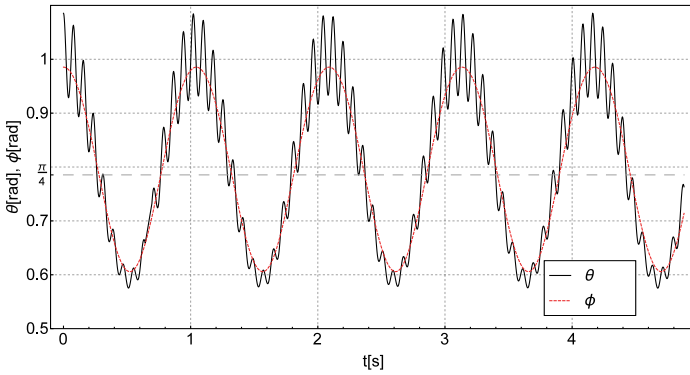


Fig. 4. Numerical simulation result of equation (2) and (4)

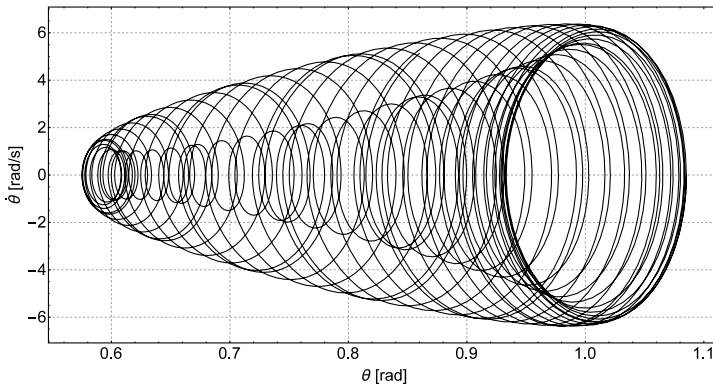


Fig. 5. Numerical simulation result of equation (2) - phase portrait

pendulum deflected from its equilibrium position starts to oscillate around this

position. The next chapter describe the control method for suppressing these oscillations.

4 Damping of the pendulum

As a control variable, the angle of oscillations of the pendulum's suspension point β has been used. This control consists in changing the angle of oscillations of the pendulum's suspension point. Mathematical formula for such a control system in the case of pendulum describes by equation (2) takes the form :

$$\begin{cases} \ddot{\theta} = \frac{3(A\Omega^2 \sin(\beta-\theta) \cos(t\Omega) + g \sin(\theta))}{2l} \\ \dot{\beta} = u \end{cases} \quad (8)$$

where u is a control input. Equation (2) is a nonlinear and worse, depends on time explicitly. To determine the control law, therefore the equation (4) describing the "smooth" motion should be used. In this case the control system is as follow:

$$\begin{cases} \ddot{\phi} = \frac{9A^2\Omega^2 \sin(2(\beta-\phi))}{16l^2} + \frac{3g \sin(\phi)}{2l} \\ \dot{\beta} = u \end{cases} \quad (9)$$

The equation (9) can be linearized around a desired operating point $\phi = \phi_{SET}, \dot{\phi} = 0, \beta = \beta_{SET} = \beta_{ext}(\phi_{SET})$:

$$\begin{pmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{\beta} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ -C_1 + C_2 & 0 & C_1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \phi \\ \dot{\phi} \\ \beta \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \quad (10)$$

where $C_1 = \frac{9A^2\Omega^2 \cos(2(\beta_{SET}-\phi_{SET}))}{8l^2}$, $C_2 = \frac{3g \cos(\phi_{SET})}{2l}$

4.1 LQR control

To determine the control of the pendulum, the state feedback control has been used. This feedback is a linear function of the state vector, therefore the control law takes the form:

$$u = -k_1(\phi - \phi_{SET}) - k_2\dot{\phi} - k_3(\beta - \beta_{SET}) \quad (11)$$

The regulator gains k_1, k_2, k_3 can be obtained using linear-quadratic regulator (see for example [17]), for quadratic cost function:

$$J = \frac{1}{2} \int_0^{\infty} (Q_{11}\phi^2 + Q_{22}\dot{\phi}^2 + Q_{33}\beta^2 + Ru^2) dt \quad (12)$$

where: $Q_{11} = \frac{1}{0.2^2}$, $Q_{22} = \frac{1}{2^2}$, $Q_{33} = \frac{1}{0.2^2}$, $R = \frac{1}{2^2}$ (according to the Bryson's rule [17]). The regulator gains depend on desired operating point, therefore relationships between operating point ϕ_{SET} and regulator gains can be drawn

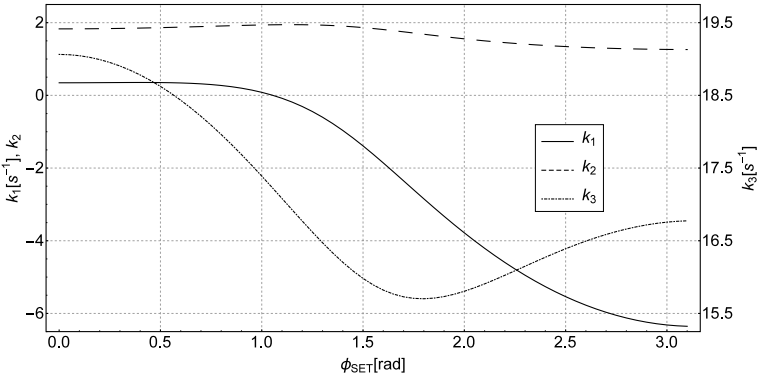


Fig. 6. LQR gains

after numerical computation of gain parameters k_1, k_2, k_3 . This relationships are shown in Fig. 6. In order to verify the correctness of designated control law, the numerical simulation for $\phi_{SET} = \pi/4$ was computed. As in the case shown in Fig. 4, $\theta(0) = \pi/4 + 0.3$ rad, $\dot{\theta}(0) = 0$ rad/s and $\beta_{SET} = 0.477$ rad. The LQR gains takes the values: $k_1 = 0.279985$ 1/s, $k_2 = 1.90619$, $k_3 = 17.9996$ 1/s. After two seconds of the numerical simulation of the equations (8) and (9), the control has been "switched on". The results of this simulation are shown in Fig. 7. As can be seen in Fig. 7, for the equation (8) describing the "real" pendulum,

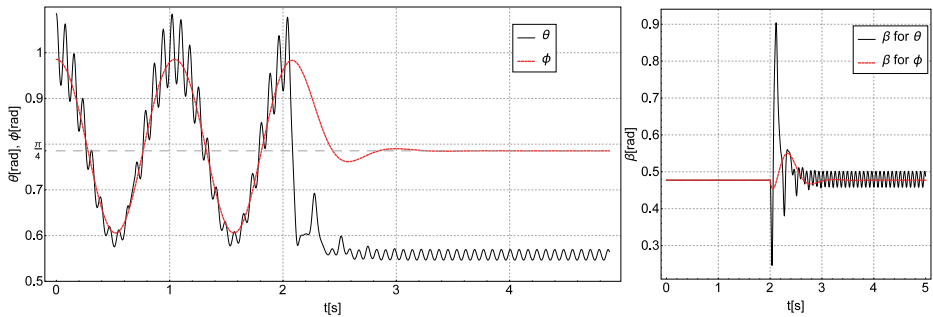


Fig. 7. Numerical simulation result of equation (8) and (9) with control law (11) ($k_1=0.279985$ 1/s, $k_2=1.90619$, $k_3=17.9996$ 1/s).

control failed: the pendulum did not reach the equilibrium state. However, the LQR control for the equation (9) describing the "smooth" pendulum completed successfully. This is due to the fact that control law was designated for the equation (9) describing smooth movement of the pendulum. To apply the designated control in the case of real pendulum described by equation (8) some smoothing filter has to be used.

4.2 Damping of the pendulum using moving average

As a filtering method the moving average method has been used. The movement of the pendulum, has to be averaged due to the rapid oscillations caused by oscillations of the pendulum's suspension point. The period of rapid oscillations of the pendulum can be calculated as $T = 2\pi/\Omega$, therefore the averaging the oscillations must be done for this the period. Average value of θ and $\dot{\theta}$ are as follows:

$$\begin{aligned}\bar{\theta}(t) &= \frac{1}{n} \sum_{i=1}^n \theta(t - (i-1)\frac{T}{n}) \\ \bar{\dot{\theta}}(t) &= \frac{1}{n} \sum_{i=1}^n \dot{\theta}(t - (i-1)\frac{T}{n})\end{aligned}\quad (13)$$

where n define the number of samples in the period T used in averaging method. The averaging has been done for $\Omega = 80$ rad/s (therefore period $T = 0.0785$ s) and for $n = 4$. The control law in the case of "real" pendulum described by equation (8) takes the form:

$$u = -k_1(\bar{\theta} - \phi_{SET}) - k_2\bar{\dot{\theta}} - k_3(\beta - \beta_{SET}) \quad (14)$$

Repeated simulation of the equation (8) with control (14) and the equation (9) with control (11) for $\phi_{SET} = \pi/4$ rad, $\theta(0) = \pi/4 + 0.3$ rad, $\dot{\theta}(0) = 0$ rad/s and $\beta_{SET} = 0.477$ rad gave the results shown in Fig. 8. As can be seen in Fig. 8,

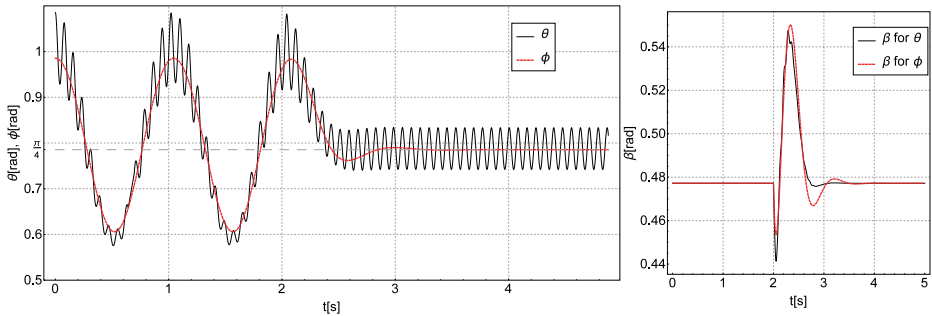


Fig. 8. Numerical simulation result of the equation (8) with control (14) and the equation (9) with control (11) ($k_1=0.279985$ 1/s, $k_2=1.90619$, $k_3=17.9996$ 1/s).

for the equation (8) with control law (14), the damping of the pendulum using moving average completed successfully: the pendulum reach the equilibrium state $\pi/4$ rad. The presence of small oscillations of the pendulum after damping should be explained by a non-zero torque of gravity force acting on the pendulum in state $\pi/4$ rad. In the case when $\phi_{SET} = 0$ rad, $\theta(0) = 0.3$ rad, $\dot{\theta}(0) = 0$ rad/s and $\beta_{SET} = 0$ rad (vertical oscillations of the pendulum's suspension point presented in Fig. 2) this small oscillations disappear what can be seen in Fig. 9.

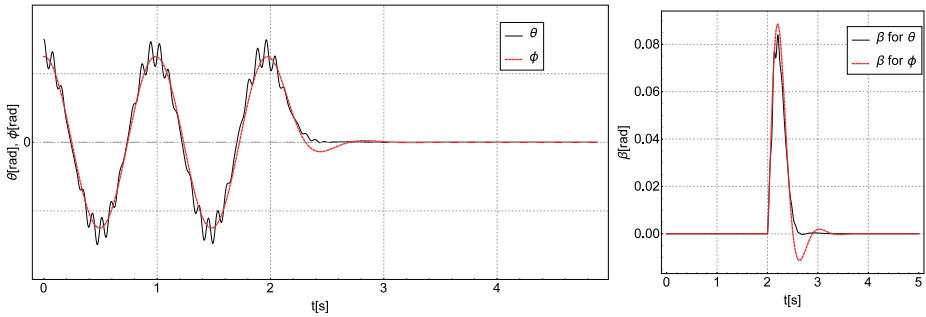


Fig. 9. Numerical simulation result of the equation (8) with control (14) and the equation (9) with control (11) ($k_1=0.345846$ 1/s, $k_2=1.82928$, $k_3=19.06351$ 1/s).

5 Conclusions

The results presented in this paper demonstrate the possibility of dumping of the pendulum using linear–quadratic regulator. The numerical solutions of equations (8) with control law (11) directly obtained from LQR showed that some smoothing filter has to be used. Using the moving average filter define in equation (13) caused that the aim of control has been achieved what can be seen in Fig. 8 and Fig. 9. The next step of research will be validation of this LQR control in the presence of random disturbances and friction force and finally experimental realization of damping of the pendulum during dynamic stabilization in arbitrary angle position.

Acknowledgments

The studies have been carried out in the framework of work No. S/WM/1/2016 and financed from the funds for science by the Polish Ministry of Science and Higher Education.

References

1. Stephenson, A.: On a new type of dynamic stability. *Memories and Proceeding of the Manchester Literary and Philosophical Society.* 52, 1–10 (1908)
2. Stephenson, A.: On induced stability. *Philosophical Magazine.* 15, 233–226 (1908)
3. Lowenstern, E.R.: The stabilizing effect of imposed oscillations of high frequency on a dynamical. *Philosophical Magazine.* 13(84), 458–486 (1932)
4. Kapica, P.L.: Pendulum with a vibrating suspension. *Usp. Fiz. Nauk.* 44, 7–15 (1951)
5. Gilyar, I., Moiseyev, N., Rahav, S., Fishman, S.: Trapping of particles by lasers: the quantum Kapitza pendulum. *Journal of Physics A.* 36(25), L409–L415 (2003)
6. Saito, H., Ueda, M.: Dynamically Stabilized Bright Solitons in a Two-Dimensional Bose-Einstein Condensate. *Phys. Rev. Lett.* 90(4), 040403 (2003)
7. Bullo F.: Averaging and vibrational control of mechanical systems. *SIAM Journal on Control and Optimization.* 41(2), 542–562 (2003)

8. Wickramasinghe, I.P.M., Berg, J.M.: Vibrational control without averaging. *Automatica*. 58,72-81 (2015)
9. Wickramasinghe, I.P.M., Berg, J.M.: Vibrational control of Mathieu's equation. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics. pp. 686-691 (2013)
10. Nakamura, Y., Suzuki, T., Koinuma, M.: Nonlinear behavior and control of a non-holonomic free-joint manipulator. *IEEE Transactions on Robotics and Automation*. 13(6), 853-862 (1997)
11. Wickramasinghe, I.P.M., Berg, J.M.: A Linearization-Based Approach to Vibrational Control of Second-Order Systems. In: ASME 2013 Dynamic Systems and Control Conference. (2013)
12. Arkhipova, I., Luongo, A., Seyranian, A.: Vibrational stabilization of upper statically unstable position of double pendulum. *Journal of Sound and Vibration*. 331(2), 457-469 (2012)
13. VanDalen, G.J.: The Driven Pendulum at Arbitrary Drive Angle. *American Journal of Physics*. 72(4), 484-491 (2004)
14. Cieżkowski, M.: Stabilization of Pendulum in Various Inclinations Using Open-Loop Control. *Acta Mechanica et Automatica*. 5(4), 22-28 (2011)
15. Cieżkowski, M.: Dynamic stabilization of the pendulum in a moving potential well. In: 21th International Conference on Methods and Models in Automation and Robotics MMAR'2016. pp. 54-58 (2016)
16. Murray, R.M., Sastry, S.S., Li Z.: *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton (1994)
17. Bryson, A.E., Ho, Y.C.: *Applied Optimal Control: optimization, estimation, and control*. Blaisdell, Waltham (1969)

Simple example of dual control problem with almost analytical solution

Piotr Bania¹

Abstract: Example of dual control of linear uncertain system have been presented. The control task with short horizon ($N=2$) were solved using dynamic programming. It was shown that the optimal solution is ambiguous, the cost function is non-convex and has many local minima. Optimal control depends in a discontinuous manner on the initial conditions. It was also observed that active learning occurs only when the uncertainty of the initial state exceeds a certain threshold. In this case, the amount of information transmitted from sensor to the controller is much greater than in the case of passive learning.

Keywords: Dual control, adaptive control, stochastic control, optimal filter, uncertain parameter, information, entropy, active learning.

Introduction. Synthesis of the optimal feedback in stochastic systems is one of the most important tasks of the adaptive control theory. In particular, the problem of this type has been precisely formulated by Feldbaum (1960) and it's known as dual control. The primary objective of the control can't be achieved without state estimation. However, state estimation is often contrary to the control task. Dual regulator is trying to reach a compromise between control and estimation. Usually this is done by minimizing the expected value of a certain objective function. Feldbaum (1960, 1961, 1965) showed that the solution can be obtained by dynamic programming. It is well known fact that this method leads to the great computational difficulties and only few very simple tasks have been solved by dynamic programming. Despite the great efforts of many researchers, the dual control problem still remains unsolved. However, a number of properties of optimal solution have been identified, which led to the development of many approximate methods.

The easiest way to obtain a suboptimal solution is to replace all stochastic variables by their expected values and finding a solution of deterministic problem. This approximation is known as *Certainty Equivalent principle* (CE, Åström and Wittenmark 1995). The *Open Loop Feedback* method (OLF, Tse 1974, Filatov and Unbehauen 2000) is based on the solution of the corresponding stochastic control problem excluding future measurements and with the initial distribution obtained from the optimal filter. Linearization around the reference trajectory, cost decomposition and various other types of approximation have been studied in the work of Tse and Bar-Shalom (1973), Tse *et. al.* (1973a), Bar-Shalom and Tse (1976), Bar-Shalom (1981). Bi-criterial approach has been developed by Filatov and Unbehauen (2004). The first criteria was mean square of tracking error. The second was predicted covariance of unknown parameters. Various approximations of dual control are also analyzed by Lindoff *et. al* (1999). Chen and Loparo (1991) studied the problem with finite number of uncertain parameters. They showed that total cost can be decomposed into CE cost which is independent on the measurements and the dual cost. Problem with uncertain parameters was also studied by Casiello and Loparo (1989),

¹ Piotr Bania is with AGH University of Science and Technology, Department of Automatics and Biomedical Engineering, Al. A. Mickiewicza 30, 30-059 Krakow, Poland, pba@agh.edu.pl.

Chen (1990), Li *et. al.* (2003, 2008) and Tenno (2010). Dual Model Predictive Control is practically important area of research. Typical dual MPC controller uses open loop feedback (OLF) with modified cost function. Modification of the cost usually enforces active learning, see Kumar *et. al.* (2015), Potschka *et. al.* (2016), Heiring *et. al.* (2017). Predictive algorithms typically require the solution of the filtering problem. The optimal filter for linear systems with uncertain parameters has been proposed by Bania and Baranowski (2016). The existence of the optimal control for continuous time systems was proved by Fleming and Pardoux (1982). Bensoussan (1983) gave the stochastic maximum principle, which in contrast to the deterministic one, leads to large complications. More information about dual and adaptive control and a comprehensive review of the literature can be found in books of Filatov and Unbehauen (2004) and Astrom and Wittenmark (1995). Many researchers point out that dual control is substantially associated with the active exchange of information between the sensor and controller. The role of entropy is also significant. Hijab (1984) showed that the entropy of the state occurs naturally in the problem of dual control. The entropic formulation of dual and adaptive control has been given by Saridis (1988, 2001) and Tsai *et. al.* (1992). Hordjewicz and Kozłowski (2006, 2007) studied active learning problem. The criterion of learning was entropy of the state and parameters. Necessary condition of optimality in the problem of active learning has been given by Banek and Kozłowski (2005, 2006, 2010). Banek (2010) significantly developed the algorithm given by Rishel (1986, 1990) and proved a necessary condition of optimality in the case when the same noise disturbs the system and measurements. This article suggests that the optimality and exchange of information are closely linked. Application of methods of information theory for optimal filtration and experimental design was studied by Feng and Loparo (1997) and Uciński and Patan (2016). Touchette and Lloyd (2000) showed that the amount of information transmitted from the sensor to the controller may not be less than the reduction in entropy resulting from the application of the given controller. They also showed (2004), that basic concepts of control theory, such as controllability, observability etc. can be defined in the context of information theory (see also Fang *et. al.* 2017). Sagawa and Ueda (2013) studied relationship between the amount of information exchanged by two dynamical systems (eg. object and regulator) and the production of entropy. The so called *Information Based Control* (Alpcan *et. al.* 2013, Alpcan and Shames 2015, Scardovi 2005) is a relatively new method which minimizes auxiliary cost $J_k = J_k^{OLF} + \lambda_k J_k^{INF}$

where J_k^{OLF} represents OLF method and J_k^{INF} represents the mutual information between the measurements and the state. Available literature contains only a few examples of dual control tasks for which the optimal solution is known. Examples of these are given in the works of Åström and Helmersson (1986), Åström and Wittenmark (1971), Bernhardsson (1989), Bohlin (1969), Cao *et. al.* (2016), Chen and Loparo (1991), Chen (1990), Feldbaum (1960, 1961, 1965), Sternby (1976). The work of Sternby and the examples given by Feldbaum are rather far from typical applications of control theory. Cao *et. al.* (2016) provide exact solution for linear systems, in which only the observation matrix depends on an unknown parameter. The solution for linear systems with uncertain or random parameters is not known. Therefore, the primary motivation, and the result of this work is to expand the list of dual control problems with known solutions. Section 1 contains a formulation of the task.

Next the solution of the filtering problem and short description of dynamic programming has been given. Example of dual control are given in section 2. Due to the high complexity of formulas the number of time steps is equal two. The article ends with conclusions and list of literature.

1. Dual control problem. Let us consider following stochastic system

$$x_{k+1} = A(\theta, u_k)x_k + B(\theta, u_k)u_k + G(\theta, u_k)w_k, \quad k = 0, 1, 2, \dots, \quad (1)$$

$$y_k = C(\theta)x_k + v_k, \quad k = 1, 2, \dots, \quad (2)$$

$$x_k \in R^n, y_k \in R^m, w_k \in R^{n_w}, v_k \in R^m, w_k \sim N(0, I), v_k \sim N(0, V(\theta)), \quad (3)$$

$$u_k \in U, U = \{u \in R^r : u_{\min} \leq u \leq u_{\max}\}, \quad k = 0, 1, 2, \dots, N-1. \quad (4)$$

Equations (1) and (2) depends on parameter $\theta \in \Omega \subset R^P$. The matrix functions A, B, G, C, V are of C^1 class. The prior distribution of parameter θ will be denoted by p_0 . The set of all initial distributions of θ is defined as $\pi_0 = \{p_0 \in L_1(\Omega; R); p_0(\theta) \geq 0 \wedge \|p_0\|_1 = 1\}$. The set of symmetrical and positive definite matrices of dimension n will be denoted by $S^+(n)$. Assume that the initial distribution of the variables (x_0, θ) has the form

$$q_0(x_0, \theta) = p_0(\theta)N(x_0, m_0, S_0)^2, \quad m_0 \in R^n, S_0 \in S^+(n), p_0 \in \pi_0. \quad (5)$$

The first measurement is performed at time $k=1$. Measurements made until time $k \geq 1$ are denoted by $Y_k = (y_1, y_2, \dots, y_k) \in R^{mk}$ and additionally $Y_0 = \{\emptyset\}$. The set of mappings $\varphi_k : R^{mk} \times R^n \times S_0 \times \pi_0 \rightarrow U$,

$$u_k = \varphi_k(Y_k, m_0, S_0, p_0), \quad k = 0, 1, 2, \dots, N-1, \quad (6)$$

will be called control strategy. Dual control task is to find the best strategy that minimizes the functional

$$J(\varphi_0, \dots, \varphi_{N-1}) = E \left\{ \frac{1}{2} \sum_{k=1}^N |x_k|_{Q_k}^2 + |\varphi_{k-1}|_{R_k}^2 \right\}, \quad N \geq 2, \quad (7)$$

where the expectation is calculated with respect to $x_0, \theta, w_0, w_1, \dots, w_{N-1}, v_1, \dots, v_{N-1}$. The matrices $Q_k, R_k \geq 0$ are symmetric. The optimal strategy in the k -th step is denoted by φ_k^* . Optimal control corresponding to the concrete realization of a variable Y_k will be denoted by $u_k^* = \varphi_k^*(Y_k, m_0, S_0, p_0)$. Calculation of the expectation in (7) requires knowledge

² $N(x, m, S) = (2\pi)^{-\frac{1}{2}n} |S|^{-\frac{1}{2}} \exp(-(x-m)^T S^{-1}(x-m))$, $N(m, S)$ denotes normal distribution with mean m and covariance S .

of the distribution of state, parameters and measurements. Theorem 1 gives way to calculate this distribution.

Theorem 1. Let (u_0, \dots, u_{N-1}) be fixed sequence and let $A_i(\theta) = A(\theta, u_i)$, $B_i(\theta) = B(\theta, u_i)$, $D_i(\theta) = G(\theta, u_i)G(\theta, u_i)^T$. If all previous assumptions are fulfilled, then joint density of variables x_{k+1}, θ, Y_k , $k = 0, 1, 2, \dots, N-1$ equals

$$p(x_{k+1}, \theta, Y_k) = p_0(\theta)N(x_{k+1}, m_{k+1}^-(\theta), S_{k+1}^-(\theta)) \prod_{i=1}^k N(y_i, C_i(\theta)m_i^-(\theta), W_i(\theta)) \quad (8)$$

where $m_i^-(\theta)$, $S_i^-(\theta)$, $W_i(\theta)$ are given by the recurrence

$$m_0(\theta) = m_0, S_0(\theta) = S_0, \quad (9)$$

$$m_i^-(\theta) = A_{i-1}(\theta)m_{i-1}(\theta) + B_{i-1}(\theta)u_{i-1}, \quad (10)$$

$$S_i^-(\theta) = A_{i-1}(\theta)S_{i-1}(\theta)A_{i-1}(\theta)^T + D_{i-1}(\theta), \quad (11)$$

$$W_i(\theta) = V(\theta) + C(\theta)S_i^-(\theta)C(\theta)^T, \quad (12)$$

$$S_i(\theta) = S_i^-(\theta) - S_i^-(\theta)C(\theta)^T W_i^{-1}(\theta)C(\theta)S_i^-(\theta), \quad (13)$$

$$m_i(\theta) = m_i^-(\theta) + S_i(\theta)C(\theta)^T V(\theta)^{-1}(y_i - C(\theta)m_i^-(\theta)), \quad (14)$$

$$i = 1, 2, \dots, k+1. \blacksquare \quad (15)$$

Proof of the theorem can be found in (Bania and Baranowski 2016, see also Zabczyk 1996). Dual control problem can be solved by dynamic programming. For this purpose let us define the following sets

$$\mathbf{I}_0 = \{m_0, S_0, p_0\}, \mathbf{I}_1 = \{\mathbf{I}_0, \varphi_0\}, \mathbf{I}_k = \{\mathbf{I}_{k-1}, \varphi_{k-1}, y_1\}, k = 2, 3, \dots \quad (16)$$

All the information about the initial data, measurements and strategies used until time $k-1$ is included in the set \mathbf{I}_k . Let the minimum cost (7) at fixed y_1, \dots, y_{k-1} , $\varphi_0, \dots, \varphi_{k-1}$, m_0, S_0, p_0 will be equal $V_k(\mathbf{I}_k)$ and let $V_N(\mathbf{I}_N) = 0$. The optimal strategy can be found by solving Bellman's equation (see e.g. Filatov and Unbehauen 2004, p. 7, Zabczyk 1996)

$$V_{k-1}(\mathbf{I}_{k-1}) = \int p(Y_{k-1}) \left(\min_{\varphi_{k-1}} \left\{ E_{x, \theta} \left(\frac{1}{2} |x_k|_{Q_k}^2 \mid \mathbf{I}_k \right) + \frac{1}{2} |\varphi_{k-1}|_R^2 + V_k(\mathbf{I}_k) \right\} \right) dy_{k-1}, \quad (17)$$

$$k = 2, \dots, N,$$

$$V_0(\mathbf{I}_0) = \min_{\varphi_0} \left\{ E_{x, \theta} \left(\frac{1}{2} |x_1|_{Q_1}^2 \mid \mathbf{I}_1 \right) + \frac{1}{2} |\varphi_0|_R^2 + V_1(\mathbf{I}_1) \right\} \quad (18)$$

where the expectation of the measurable function $F(x_k, \theta)$ is defined as

$$E_{x, \theta} (F(x_k, \theta) \mid \mathbf{I}_k) = \int p(x_k, \theta \mid Y_{k-1}) F(x_k, \theta) dx d\theta. \quad (19)$$

The expression in braces in (17), (18) will be called partial risk and will be denoted by

$$R_{k-1}(\mathbf{I}_k) = E_{x,\theta} \left(\frac{1}{2} |x_k|^2_{Q_k} | \mathbf{I}_k \right) + \frac{1}{2} | \varphi_{k-1} |^2_R + V_k(\mathbf{I}_k). \quad (20)$$

The optimal strategy at time $k-1$ minimizes the partial risk i.e.

$$\varphi_{k-1}^*(\mathbf{I}_k) = \arg \min_{\varphi_{k-1}} R_{k-1}(\mathbf{I}_k). \quad (21)$$

As we can see, the method of dynamic programming generates enormous computational difficulties and only very simple problems can be solved by it. Therefore below, we will limit ourselves to one and two-dimensional systems with $N=2$. We'll perform only one measurement y_1 and seek strategies φ_0 and φ_1 .

2. Dual control of first order system with random gain. Consider first order deterministic system

$$\dot{\xi}(t) = -a_{2c}\xi(t) + (b_c + \zeta(t))u(t) + g_{2c}w_2(t), \quad (22)$$

where ζ and w_2 represent a change in gain and noise at the input of the system, respectively. If we assume that ζ is a Wiener process, and w_2 is white noise, the equation (22) can be written in the form of two Ito equations

$$dx = (A_c(u)x + B_c u)dt + G_c dw, \quad A_c = \begin{bmatrix} 0 & 0 \\ u & -a_{2c} \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ b_c \end{bmatrix}, \quad G_c = \text{diag}(g_{1c}, g_{2c}). \quad (23)$$

The first of equations (23) models the random gain variations, the second corresponds to the equation (22). The initial condition is Gaussian i.e. $x(0) \sim N(m_0, S_0)$. The measurements have the form

$$y_k = x_2(t_k) + v_k, \quad v_k \sim N(0, V), \quad t_k = kT_0, \quad T_0 > 0, \quad k = 1, 2, \dots \quad (24)$$

Assuming that control u is piecewise constant i.e. $u(t) = u_k, t \in [t_{k-1}, t_k)$, we can make the discretization of (23). Discrete-time system corresponding to (23,24) has the form

$$x_{k+1} = A(u_k)x_k + Bu_k + G(u_k)w_k, \quad y_k = Cx_k + v_k, \quad x_0 \sim N(m_0, S_0), \quad C = [0, 1], \quad (25)$$

$$A(u_k) = \begin{bmatrix} a_1 & 0 \\ a_3 u_k & a_2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ b \end{bmatrix}, \quad D(u_k) = \begin{bmatrix} d_1 & u_k d_2 \\ u_k d_2 & d_3 + d_4 u_k^2 \end{bmatrix}, \quad G(u_k)G(u_k)^T = D(u_k), \quad (26)$$

wherein individual elements of the matrix A, B, D can be calculated from formulas

$$A(u_k) = e^{A_c(u_k)T_0}, \quad B = \int_0^{T_0} e^{A_c(u_k)\tau} B_c d\tau, \quad D(u_k) = \int_0^{T_0} e^{A_c(u_k)\tau} G_c^2 e^{A_c(u_k)^T \tau} d\tau. \quad (27)$$

Note that (25) is a special case of the system (1), (2) with a fixed parameter θ . We're looking for minimum of the performance index

$$J(\varphi_0, \varphi_1) = \frac{1}{2} E \left\{ q_1 x_{2,1}^2 + r_0 \varphi_0^2 + q_2 x_{2,2}^2 + r_1 \varphi_1^2 \right\} \quad (28)$$

where $x_{2,1}$, $x_{2,2}$ denote second component of vector x_k at time $k=1,2$ and $q_k \geq 0$, $r_k > 0$. Matrices $A(u_k)$, $D(u_k)$ can be written as

$$A(u_k) = \bar{A}_0 + \bar{A}_1 u_k, \quad D(u_k) = \bar{D}_0 + \bar{D}_1 u_k + \bar{D}_2 u_k^2, \quad (29)$$

where

$$\bar{A}_0 = \begin{bmatrix} a_1 & 0 \\ 0 & a_2 \end{bmatrix}, \quad \bar{A}_1 = \begin{bmatrix} 0 & 0 \\ a_3 & 0 \end{bmatrix}, \quad \bar{D}_0 = \begin{bmatrix} d_1 & 0 \\ 0 & d_3 \end{bmatrix}, \quad \bar{D}_1 = \begin{bmatrix} 0 & d_2 \\ d_2 & 0 \end{bmatrix}, \quad \bar{D}_2 = \begin{bmatrix} 0 & 0 \\ 0 & d_4 \end{bmatrix}. \quad (30)$$

Let A, B be square matrices of dimension n , and let

$$\langle A, B \rangle = \sum_{i,j=1}^n A_{i,j} B_{i,j}. \quad (31)$$

Using thm.1 and (17) and (19), after a rather laborious transformations we obtain the formula for risk in the last step

$$R_1(\varphi_0, \varphi_1, y_1) = \frac{1}{2} \alpha_1(\varphi_1, \varphi_0) \varphi_1^2 + \beta_1(\varphi_1, \varphi_0) \varphi_1 + \gamma_1(\varphi_1, \varphi_0) \quad (32)$$

where

$$\alpha_1(\varphi_0, y_1) = (\bar{A}_1 m_1 + B)^T Q_2 (\bar{A}_1 m_1 + B) + \langle \bar{A}_1 S_1 \bar{A}_1^T + \bar{D}_2, Q_2 \rangle + r_1, \quad (33a)$$

$$\beta_1(\varphi_0, y_1) = (\bar{A}_1 m_1 + B)^T Q_2 \bar{A}_0 m_1 + \frac{1}{2} \langle \bar{A}_0 S_1 \bar{A}_1^T + \bar{A}_1 S_1 \bar{A}_0^T + \bar{D}_1, Q_2 \rangle, \quad (33b)$$

$$\gamma_1(\varphi_0, y_1) = \frac{1}{2} m_1^T \bar{A}_0^T Q_2 \bar{A}_0 m_1 + \frac{1}{2} \langle \bar{A}_0^T S_1 \bar{A}_0 + \bar{D}_0, Q_2 \rangle, \quad (33c)$$

$$S_1(\varphi_0) = S_1^-(\varphi_0) - S_1^-(\varphi_0) C^T (V + C S_1^-(\varphi_0) C^T)^{-1} C S_1^-(\varphi_0) \quad (34)$$

$$m_1(\varphi_0, y_1) = m_1^-(\varphi_0) + S_1(\varphi_0) C^T V^{-1} (y_1 - C m_1^-(\varphi_0)) \quad (35)$$

and $Q_k = \text{diag}([0, q_k])$, $k=1,2$. It can be seen that $\alpha_1, \beta_1, \gamma_1$ are rational functions of the variable φ_0 , and the second degree polynomial for y_1 . The optimal strategy minimizes R_1 , hence φ_1^* and minimal risk are equal

$$\varphi_1^*(\varphi_0, y_1) = -\frac{\beta_1(\varphi_0, y_1)}{\alpha_1(\varphi_0, y_1)}, \quad R_1^* = \gamma_1(\varphi_0, y_1) - \frac{\beta_1(\varphi_0, y_1)^2}{2\alpha_1(\varphi_0, y_1)}. \quad (36)$$

From the Bellman equation (17) we get

$$V_1(\varphi_0) = \int N(y_1, C m_1^-(\varphi_0), V + C S_1^-(\varphi_0) C^T) R_1(\varphi_0, \varphi_1^*(\varphi_0, y_1), y_1) dy_1. \quad (37)$$

Risk at the first step equals

$$R_0(\varphi_0) = \frac{1}{2} \alpha_0 \varphi_0^2 + \beta_0 \varphi_0 + \gamma_0 + V_1(\varphi_0), \quad (38)$$

$$\alpha_0 = (\bar{A}_1 m_0 + B)^T Q_1 (\bar{A}_1 m_0 + B) + \langle \bar{A}_1 S_0 \bar{A}_1^T + \bar{D}_2, Q_1 \rangle + r_0, \quad (39a)$$

$$\beta_0 = (\bar{A}_1 m_0 + B)^T Q_1 \bar{A}_0 m_0 + \frac{1}{2} \langle \bar{A}_0 S_0 \bar{A}_1^T + \bar{A}_1 S_0 \bar{A}_0^T + \bar{D}_1, Q_1 \rangle, \quad (39b)$$

$$\gamma_0 = \frac{1}{2} m_0^T \bar{A}_0^T Q_1 \bar{A}_0 m_0 + \frac{1}{2} \langle \bar{A}_0 S_0 \bar{A}_0^T + \bar{D}_0, Q_1 \rangle. \quad (39c)$$

The strategy φ_0^* minimizes (38). Minimal cost is equal $V_0(\mathbf{I}_0) = R_0(\varphi_0^*)$. The integral (37) is calculated numerically. Parameters of the continuous time system and the sampling period were $a_{2c} = 1$, $b_c = 1$, $g_{1c} = g_{2c} = \sqrt{2}$, $T_0 = 0.1$. Parameters of corresponding discrete-time system are equal $a_1 = 1$, $a_2 = 0.9048$, $a_3 = 0.09516$, $b_2 = 0.09516$, $d_1 = 0.2$, $d_2 = 0.009675$, $d_3 = 0.1813$, $d_4 = 0.6189 \cdot 10^{-3}$. The weights were $r_0 = r_1 = 10^{-3}$, $q_1 = 0$, $q_2 = 1$. Initial conditions were equal $m_0 = (0, m_{20})^T$, $S_0 = \text{diag}(s_{10}, s_{20})$, $s_{20} = 0.1$. Fig. 1 shows the graph of the R_0 (see. 38) for $m_{20} = 0$ and several values of s_{10} .

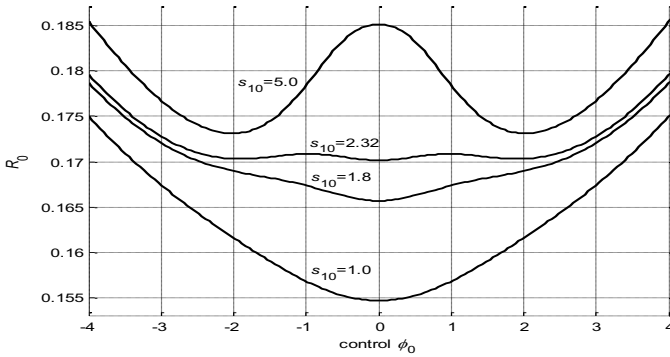


Fig 1. Graph of function R_0 for $m_{20} = 0$ and several values of s_{10} .

According to (22) and (23) the number s_{10} represents the uncertainty of the gain. If the uncertainty is small, the optimal control in the first step is zero. The increase in uncertainty above a certain threshold causes a zero is no longer optimal control despite the initial condition is concentrated around zero. Non-zero control in the first step gives information about the gain of the system and use it in a second step. This is the result of control duality. Note that this effect occurs only above a certain threshold of uncertainty. In order to explore the optimal controller, Fig. 2 shows the dependence of optimal control φ_0^* on initial condition m_{20} , for different values of uncertainty s_{10} . A small uncertainty (Fig. 2a) gives a proportional controller. The increase in uncertainty locally reduces the gain and yields discontinuous dependence on the initial condition m_{20} . Further growth of uncertainty (Fig. 2b) results in active learning and the rapid increasing in the amount of information transmitted from the sensor to the controller. Measure of the amount of information is mutual information between (x_1, x_2) and y_1 . It can be shown that this information is equal to

$$I((x_1, x_2); y_1) = \frac{1}{2} \ln |V + CS_1^-(\varphi_0)C^T| - \ln |V|. \quad (40)$$

Fig. 3a shows the dependence of mutual information on control u_0 . Zero control does not contain information about the gain of the system, which also follows directly from (22), (23) and (24). Control with larger amplitudes generates more information. Fig. 3b shows the amount of information transmitted from the sensor to the optimal controller. Rising uncertainty over a certain threshold enforces active learning.

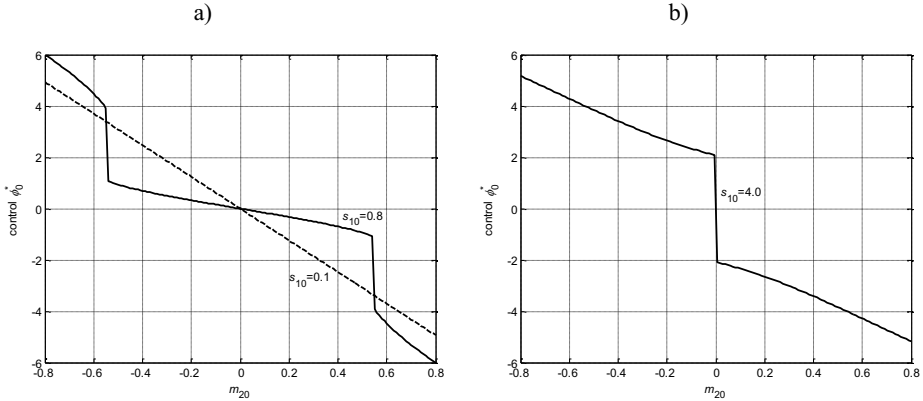


Fig 2. Dependence of optimal control ϕ_0^* on initial condition m_{20} .

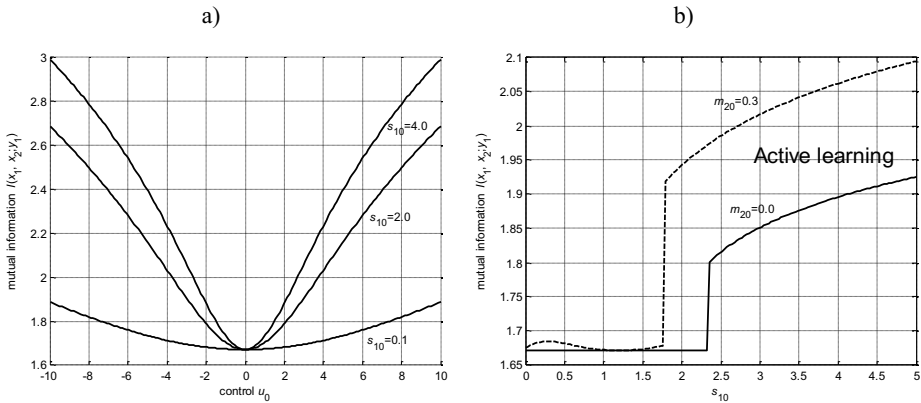


Fig. 3. a) Mutual information according to the initial control u_0 . b) The amount of information transmitted from the sensor to the optimal controller as a function of initial uncertainty s_{10} .

4. Summary. Simple problem of dual control of linear system with random parameter and quadratic cost has been solved. The number of time steps was 2. A small number of steps allowed to save the solution in a relatively simple form. Finding a solution requires a numerical calculation of one dimensional integral, which is not currently a serious challenge. It was found that cost function can have many local minima and the optimal solution can be ambiguous. The optimal controller is generally discontinuous function of the initial data. In both examples, the increased uncertainty resulted in a qualitative change in the nature of the control and rapid increase in the amount of information transmitted from the sensor to the controller. This additional information reduces the uncertainty. When the uncertainty is small, the dual effect is not present.

Similar phenomena can be observed among living organisms. When the uncertainty is small, the optimal decisions are generally known. Then there is no need for substantial modification of behavior. The large uncertainty typically enforces a change of strategy and creates a new quality. One can therefore formulate a risky but not unfounded hypothesis, that dual control theory may explain the qualitative changes of the behaviour of living organisms caused by increasing uncertainty. These changes may in fact result from minimizing the expected value of some cost.

5. References

- [1] Alpcan T., Shames I., (2015): *An Information-Based Learning Approach to Dual Control*. IEEE Trans. on Neural Networks and Learning Systems Vol.26, Issue 11 Pages: 2736 – 2748.
- [2] Alpcan T.; Shames I; Cantoni M.; Nair G. (2013): *Learning and information for dual control* 2013 9th Asian Control Conference (ASCC), Pages: 1 - 6
- [3] Åström K, Wittenmark B. (1995): *Adaptive Control*. Addison-Wesley, 2nd edition.
- [4] Åström K., Helmersson A. (1986): *Dual control of an integrator with unknown gain*. Comp. & Maths. With Appls.,12A:6, pp. 653-662.
- [5] Åström K., Wittenmark B. (1971): *Problems of identification and control*. Journal of Mathematical Analysis and Applications, 34, pp. 90-113.
- [6] Banek T. (2010): *Incremental value of information for discrete-time partially observed stochastic systems* Control and Cybernetics vol. 39, No. 3.
- [7] Banek T., Kozłowski E. (2005): *Active and passive learning in control processes application of the entropy concept*, Systems Sciences, vol. 31, N.2, pp. 29 - 44.
- [8] Banek T., Kozłowski E. (2006): *Adaptive control of system entropy* Contr. and Cybernetics 35, 2.
- [9] Banek T., Kozłowski E. (2010): *Active learning in discrete-time stochastic systems* In: Jerzy Jozefczyk and Donat Orski (ed.), *Knowledge-based Intelligent System Advancements: Systemic and Cybernetic Approaches*, pp.350-371.
- [10] Bania P., Baranowski J. (2016): *Field Kalman Filter and its approximation*. In proc of 55th IEEE Conference on Decision and Control December 12-14, Las Vegas, USA.
- [11] Bar-Shalom Y., Tse E. (1976): *Caution, probing, and the value of information in the control of uncertain systems*. Ann. Econ. Social. Measurement. Vol 5. pp. 323-337.
- [12] Bar-Shalom Y. (1981): *Stochastic Dynamic Programming: Caution and Probing*. IEEE trans Aut. Contr. Vol AC-26, No 5 pp. 1184-1195.
- [13] Bensoussan, A. (1983), *Maximum Principle and Dynamic Programming Approach of the Optimal control of Partially Observed Diffusions*, Stochastics, vol.9, issue 3, 169–222.
- [14] Bernhardsson B. (1989): *Dual control of a first-order system with two possible gains*. Int. J. of Adaptive Control and Signal Processing, 3, pp. 15-22.
- [15] Bohlin T. (1969): *Optimal dual control of a simple process with unknown gain*. Report PT 18.196, IBM Nordic Laboratory, Lidingö, Sweden.
- [16] Cao S., Qian F., Wang X. (2016): *Exact optimal solution for a class of dual control problems*. International Journal of Systems Science, 47:9, 2078-2087.
- [17] Casiello F., Loparo K. A. (1989): *Optimal control of unknown parameter systems*. IEEE Trans. Aut. Contr. AC-34, pp.1092-1094.
- [18] Chen R., Loparo K. A. (1991): *Dual control of linear stochastic systems with unknown parameters*. IEEE International Conference on Systems Engineering pp. 65-68.
- [19] Chen, R., (1990): *Dual Control of Linear Stochastic Systems with Unknown Parameters*. Ph.D. Dissertation, Systems Engineering, Case Western Reserve University.
- [20] Fang S., Chen J., Hideaki I., (2017): *Towards integrating control and information theories*. Lecture notes in control and information sciences 465. Springer.
- [21] Feldbaum A.A. (1965): *Optimal control systems*. Elsevier Science.
- [22] Feldbaum, A.A. (1960): *Dual Control Theory I–II*, J. Aut. Remote Cont., 21, 874–880, 1033–1039.
- [23] Feldbaum, A.A. (1961), *Dual Control Theory III–IV*, J. Aut. Remote Cont., 22, 1–12, 109–121.
- [24] Feng X., Loparo K., (1997): *Optimal State Estimation for Stochastic Systems: An Information Theoretic Approach* IEEE Transactions on Automatic Control, Vol. 42, No. 6.
- [25] Filatov N. M, Unbehauen H. (2000): *Survey of adaptive dual control methods*. IEE Proceedings - Control Theory and Applications 147(1):118 – 128.
- [26] Filatov, N.M., Unbehauen, H. (2004): *Adaptive Dual Control: Theory and Applications*, Lecture Notes in Control and Information Sciences No. 302.

- [27] Fleming, W.H., Pardoux, H. (1982): *Optimal Control of Partially Observed Diffusion*, SIAM Journal on Control and Optimization, 20, 261–285.
- [28] Heirung T. A., Ydstie B., Foss B. (2017): *Dual Adaptive Model Predictive Control*. Automatica, Feb. 2017, to appear.
- [29] Hijab O. (1984): *Entropy and dual control* Proc. of 23rd C DC, Las Vegas NV.
- [30] Hordjewicz T., Kozłowski E. (2006): *Comparison of stochastic optimal controls with different level of self-learning* Annales UMCS Informatica AI 5 (2006) 343-356.
- [31] Hordjewicz T., Kozłowski E. (2007): *The self-learning active problem in dynamic systems* Annales UMCS Informatica AI 7 (2007) 171-180
- [32] Kumar K, Heirung T. A., N., Patwardhan S. C., Foss B. (2015): *Experimental Evaluation of a MIMO Adaptive Dual MPC*. 9th IFAC Symposium on Advanced Control of Chemical Processes ADCHEM 2015 Whistler, Canada, 7–10 June 7 – 10, 2015, IFAC-PapersOnLine Volume 48, Issue 8, 545-550 .
- [33] Li D.; Qian F.; Fu P. (2008): *Optimal nominal dual control for discrete-time linear-quadratic Gaussian problems with unknown parameters* Automatica 44, 119 – 127
- [34] Li D; Fu P.; Qian F. (2003): *Optimal nominal dual control for discrete-time LQG problem with unknown parameters*, SICE Annual Conference, <http://ieeexplore.ieee.org/document/1323396/>
- [35] Lindoff B., Holst J., Wittenmark B. (1999): *Analysis of approximations of dual control*. Int. J. Adapt. Control Signal Process. 13, 593-620.
- [36] Potschka H. C, Schloder J. P., Bock H. G. (2016): *Dual Control and Information Gain in Controlling Uncertain Processes*. 11th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems June 6-8, 2016. NTNU, Trondheim, Norway..
- [37] Rishel, R. (1986): *An Exact Formula for a Linear Quadratic Adaptive Stochastic Optimal Control Law*, SIAM Journal on Control and Optimization, 24, 667–674.
- [38] Rishel, R. (1990), *A Comment on a Dual Control Problem* in Proc. of 19th IEEE Conference on Decision and Control Including the Symposium on Adaptive Processes, pp. 337–340.
- [39] Sagawa T., Ueda M. (2013): *Role of mutual information in entropy production under information exchanges*. New J. Phys. 15 125012,
- [40] Saridis G. N. (1988): *Entropy formulation of optimal and adaptive control*. IEEE Trans. Aut. Cont. Vol: 33, Issue: 8.
- [41] Saridis G. N. (2001): *Entropy in control engineering*. Series in Intelligent Control and Intelligent Automation. World Scientific Publishing.
- [42] Scardovi L. (2005): *Information based control for state and parameter estimation*. PhD thesis. University of Genoa Dep. of Communication, Faculty of Engineering Computer and System Sciences.
- [43] Sternby J. (1976): *A simple dual control problem with an analytical solution*. IEEE Trans. Aut. Cont., 21(6):840–844.
- [44] Tenno R. (2010), *Dual adaptive controls for linear system with unknown constant parameters*, International Journal of Control, 83:11, 2232-2240
- [45] Touchette H., Lloyd S. (2000): *Information-theoretic limits of control*. Phys Rev Lett. 2000 Feb 7;84(6):1156-9.
- [46] Touchette H., Lloyd S. (2004): *Information-theoretic approach to the study of control systems*. Phys. A 331, 140-172.
- [47] Tsai Y. A., Casiello F. A., Loparo K. A. (1992): *Discrete-time entropy formulation of optimal and adaptive control problems*. IEEE Trans. Aut. Cont. vol 37, No. 7.
- [48] Tse E. (1974): *Adaptive Dual Control Methods*. Annals of Economic and Social Measurement, Vol. 3. No. 1. (1974)
- [49] Tse, E., and Bar-Shalom, Y. (1973), *An Actively Adaptive Control for Linear Systems with Random Parameters via the Dual Control Approach*, IEEE Trans. Aut. Control, 18, 109–117.
- [50] Tse, E., and Bar-Shalom, Y. (1976), *Actively Adaptive Control for Nonlinear Stochastic Systems*, Proceedings of the IEEE, 64, 1172–1181.
- [51] Tse, E., Bar-Shalom, Y., Meier L. (1973a): *Wide sense adaptive dual control for nonlinear stochastic systems*. IEEE Trans. Aut. Control, 18, 98–108.
- [52] Uciński D., Patan M. (2016): *D-optimal spatio-temporal sampling design for identification of distributed parameter systems* In proc of 55th IEEE Conference on Decision and Control December 12-14, Las Vegas, USA, 3985-3990.
- [53] Wittenmark B. (1995): *Adaptive dual control methods; an overview*. Proc. Of 5th IFAC symposium of adaptive systems in control and signal processing. Budapest, pp. 67-72.
- [54] Zabczyk J. (1996): *Chance and decision. Stochastic control in discrete time*. Quaderni Scuola Normale di Pisa

A Comparison of LQR and MPC Control Algorithms of an Inverted Pendulum

Andrzej Jezierski, Jakub Mozaryn, Damian Suski

Institute of Automatic Control and Robotics, Warsaw University of Technology, ul.
Sw. A. Boboli 8, Warsaw, Poland

Abstract. The subject of this paper is a comparison of two control strategies of an inverted pendulum on a cart. The first one is a linear-quadratic regulator (LQR), while the second is a state space model predictive controller (SSMPC). The study was performed on the simulation model of an inverted pendulum, determined on the basis of the actual physical parameters collected from the laboratory stand AMIRA LIP100. It has been shown that the LQR algorithm works better for fixed-value control and disturbance rejection, while the SSMPC controller is more suitable for the trajectory tracking task. Furthermore, the system with SSMPC controller has smoother changes in the control signal, that can be beneficial for an actuator, while LQR controller may generate adverse, rapid changes in the control signal.

1 Introduction

The **inverted pendulum on a cart** (pole-cart system) is a popular benchmark used for illustrating non-linear control techniques. Such system is inherently unstable, when it has its center of mass above its pivot point. The control objective is to bring and keep the pendulum in the upper unstable equilibrium position by moving the cart (pivot point) horizontally.

An inverted pendulum on a cart belongs to a broad class of devices and systems found in robotics and control theory called **underactuated systems**. There are many unmanned underactuated systems, recently reaching popularity such as legged, swimming and flying robots. Also robot manipulation is usually an underactuated task [1]. In general, the underactuated system is a system that has fewer control inputs than degrees of freedom. Deficiency of input signals causes the problem because external generalized forces are not able to command instantaneous accelerations in all directions in the configuration space. The control of such systems is an open and interesting problem.

There are many approaches to tackle the problem of inverted pendulum control. They range from PID control [11] through state space optimal linear control algorithms [5], model predictive control [12] to nonlinear approaches such as sliding mode [10], neural-network [8], fuzzy logic [7] and energy-based [9] controllers.

Recent developments in Programmable Logic Controllers (PLC) and Programmable Automation Controllers (PAC) allow engineers to adopt in industrial

applications control strategies based on a state-space description of a control system. One can observe popularity of two such algorithms in industrial practice - the Linear-Quadratic Regulator (LQR) [14] and the Model Predictive Controller (MPC) [13],[17], [20]. Both belong to the class of pseudo-optimal algorithms.

The aim of this paper is to compare two control algorithms - namely LQR and SSMPC (State Space Model Predictive Control) for positioning of the inverted pendulum in the underactuated pole-cart system.

The scope of the article is as follows. Chapter 2 describes the mathematical model of an inverted pendulum and its linearization in the upward position of the pendulum. In Chapter 3 implemented LQR and SSMPC control algorithms are presented. Chapter 4 gathers the results of simulation studies of the designed control systems including trajectory tracking and response to disturbances. Finally in Chapter 5 research summary and suggestions for further work are presented.

2 Mathematical model of a cart-pole system

An inverted pendulum, considered in the article, is a pendulum (pole) that has its center of mass above its pivot point, which is mounted on a horizontally movable platform (cart). The pendulum is free to swing about its pivot point and it has no direct control actuation. Therefore pole in an upright position is inherently unstable and must be actively balanced. In considered case, it can be done, by moving the pivot point (cart) horizontally, using a force applied to it, as a part of a feedback system.

Schematic diagram of the considered cart-pole system and the picture of the laboratory stand AMIRA LIP100 are given in Fig. 1.

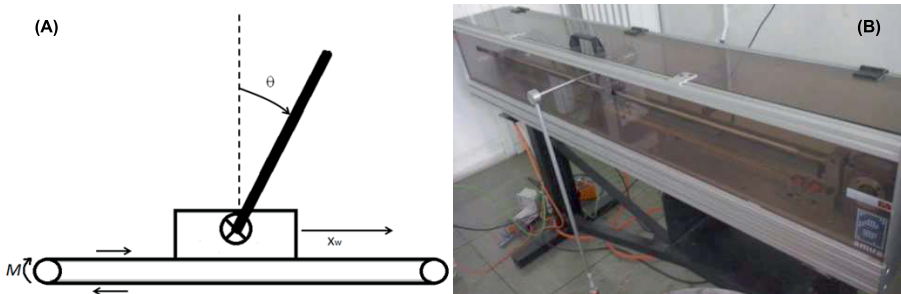


Fig. 1. An inverted pendulum: (A) a schematic diagram of an inverted pendulum on a cart (B), the laboratory stand: AMIRA LIP100

The non-linear model of the pole-cart system can be described with the following equations [4]

$$\begin{cases} \ddot{x}_W = \frac{M}{r} - F_x - b\dot{x}_W \\ \ddot{\theta} = \frac{-F_x l \cos \theta + F_y l \sin \theta}{I} \\ F_x = m_P \ddot{x}_W - m_P l \dot{\theta}^2 \sin(\theta) + m_P l \ddot{\theta} \cos(\theta) \\ F_y = m_P g - m_P l \dot{\theta}^2 \cos(\theta) - m_P l \ddot{\theta} \sin(\theta) \end{cases} \quad (1)$$

where: x_W - a distance of the cart's center of mass from its initial position (output), θ - an angular position of the pendulum rod (output), M - a torque generated by the actuator (control input), r - an actuator ratio, b - a viscous friction coefficient, l - a distance from the pivot point to the center of gravity of the pendulum, I - an inertia of the pendulum about its center of gravity, m_P - a mass of the pendulum, m_W - a mass of the cart, g - an acceleration due to gravity.

For the system given in Fig. 1: $\theta = 0$ means that the pendulum is in the upright position and $x_W = 0$ means that the cart is in the middle of the guiding bar.

Values of physical coefficients of the pole-cart system base on the measurements performed at the laboratory stand AMIRA LIP100 and are gathered in Tab.1.

Table 1. Values of physical coefficients of the pole-cart system: AMIRA LIP100

Parameter \rightarrow	l	m_W	m_P	b	I	r	g
	m	kg	kg	$\frac{N \cdot s}{m}$	$kg \cdot m^2$	m	$\frac{m}{s^2}$
	0.437	4	0.241	6.5	0.0053	0.1	9.81

In order to design LQR and SSMPC controllers, model (1) was linearized at the operating point, chosen as the upright position of the pendulum ($\theta = 0$, $\dot{\theta} = 0$, $x_W = 0$, $\dot{x}_W = 0$), and rewritten in the form of a state-space model

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \quad (2)$$

where

$$x(t) = \begin{bmatrix} x_W \\ \dot{x}_W \\ \theta \\ \dot{\theta} \end{bmatrix}, y(t) = \begin{bmatrix} x_W \\ \theta \end{bmatrix}, n_x = 4, n_y = 2 \quad (3)$$

are the state vector and the output vector respectively and state matrices can be written as

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-b(I + m_P l^2)}{I(m_W + m_P) + m_W m_P l^2} & \frac{g m_P^2 l^2}{I(m_W + m_P) + m_W m_P l^2} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{-m_P I b}{I(m_W + m_P) + m_W m_P l^2} & \frac{m_P g l (m_W + m_P)}{I(m_W + m_P) + m_W m_P l^2} & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 \\ \frac{I + m_P l^2}{r I (m_W + m_P) + r m_W m_P l^2} \\ 0 \\ \frac{m_P l}{r I (m_W + m_P) + r m_W m_P l^2} \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, D = [0]$$

After substitution of the values from the Tab. 1 in (2) and discretization, the obtained discrete-time state-space model has the following form

$$\begin{cases} x(k) = A_d x(k) + B_d u(k) \\ y(k) = C_d x(k) + D_d u(k) \end{cases} \quad (4)$$

where

$$A_d = \begin{bmatrix} 1.0000 & 0.0099 & 2 \cdot 10^{-5} & 9 \cdot 10^{-8} \\ 0 & 0.9840 & 0.0052 & 2 \cdot 10^{-5} \\ 0 & -0.0002 & 1.0011 & 0.0100 \\ 0 & -0.0329 & 0.2120 & 1.0011 \end{bmatrix}, B_d = \begin{bmatrix} 0 \\ 0.0246 \\ 0.0003 \\ 0.0506 \end{bmatrix}, C_d = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, D_d = [0]$$

where: k - a discrete time, T_p - a sampling time (in simulations $T_p = 0.01$ s), $t = kT_p$.

Internal states of the system i.e. pendulum and cart velocities were calculated using two-point numerical differentiation as follows

$$\dot{x}_W(k-1) = \frac{x_W(k) - x_W(k-1)}{T_p}, \quad \dot{\theta}(k-1) = \frac{\theta(k) - \theta(k-1)}{T_p} \quad (5)$$

3 Description of control algorithms

3.1 Linear-Quadratic Regulator - LQR

LQR optimal control problem is defined as the problem of finding an optimal control such that the following cost function is minimized [19]

$$J_{LQR} = \sum_{k=0}^{\infty} (\|x(k)\|_Q^2 + \|u(k)\|_R^2), \quad Q \in R^{n_x \times n_x}, R \in R^{n_u \times n_u} \quad (6)$$

where: Q - symmetric nonnegative definite matrix, R - symmetric positive definite matrix.

Usually, matrices Q and R are chosen to be diagonal matrices that determine the significance of internal states and control signals on the cost function.

The optimal state-feedback LQR controller is a simple matrix gain of the form

$$u(k) = -K_{LQR}x(k-1) \in R^{n_u}, K_{LQR} \in R^{n_u \times n_x} \quad (7)$$

Assuming that matrices A , B , Q , R are all time-invariant, the control matrix gain K_{LQR} in (7) which minimizes the cost function (6), can be calculated as follows [2]

$$K_{LQR} = (R + B^T P(k) B)^{-1} B^T P(k) A \quad (8)$$

where $P(k)$ is a solution of an algebraic Riccati equation [15]

$$P(k-1) = Q + A^T [P(k) - P(k) B (R + B^T P(k) B)^{-1} B^T P(k)] A \quad (9)$$

3.2 State Space Model Predictive Controller - SSMPC

In general a Model Predictive Control (MPC) is a family of different algorithms i.e. Model Algorithmic Control (MAC), Dynamic Matrix Control (DMC), Generalized Predictive Control (GPC), State Space MPC (SSMPC) [16],[18].

In MPC approach, actual values of output and input signals and their predicted values are taken into consideration to determine the control signal that over the prediction horizon N minimizes the cost function of the form

$$J_{SSMPC}(k) = \sum_{p=1}^N \|e(k+p|k)\|_{\Psi_p}^2 + \sum_{p=0}^{N_u-1} \|\Delta u(k+p|k)\|_{\Lambda_p}^2, \Psi_p \in R^{n_y \times n_y}, \Lambda_p \in R^{n_u \times n_u} \quad (10)$$

where: $k+p|k$ - predicted values for the moment $k+p$ at the step k , N_u - a control horizon, Ψ_p - a symmetric nonnegative definite matrix, Λ_p - a symmetric positive definite matrix and $e(k+p|k)$ is an output error defined as

$$e(k+p|k) = y(k+p|k) - y_{SP}(k+p|k) \quad (11)$$

where y_{SP} is a reference trajectory.

The SSMPC controller determines the behavior of a plant model over a given prediction horizon N . Solution of the cost function can be written in the form

$$\Delta U(k) = K_{SSMPC} (Y_{SP}(k) - Y_0(k)), K_{SSMPC} \in R^{N_u \cdot n_u \times N \cdot n_y} \quad (12)$$

where: $\Delta U(k) = [\Delta u(k|k), \dots, \Delta u(k+N_u-1|k)]^T$ - a control increments vector calculated for the entire control horizon, $Y_0(k) = [y_0(k+1|k), \dots, y_0(k+N|k)]^T$ - an expected output free trajectory, dependent only on past controls, $Y_{SP}(k) = [y_{SP}(k+1|k), \dots, y_{SP}(k+N|k)]^T$ - an output reference trajectory over the prediction horizon N .

The gain matrix K_{SSMPC} minimizing the cost function (10) over the prediction horizon N can be determined as follows [21]

$$K_{\text{SSMPC}} = \left(\tilde{C}P^T \Psi \tilde{C}P + \Lambda \right)^{-1} (\tilde{C}P)^T \quad (13)$$

where

$$\tilde{C} = \begin{bmatrix} C_d & 0 & \cdots & 0 \\ 0 & C_d & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & C_d \end{bmatrix} \in R^{N \cdot n_y \times N \cdot n_x} \quad (14)$$

$$P = \begin{bmatrix} & B_d & \cdots & 0 \\ & \vdots & \vdots & \vdots \\ (A_d^{N_u-1} + \cdots + A_d + I)B_d & \vdots & & B_d \\ (A_d^{N_u} + \cdots + A_d + I)B_d & \vdots & & (A_d + I)B_d \\ & \vdots & \vdots & \vdots \\ (A_d^{N-1} + \cdots + A_d + I)B_d & \vdots & (A_d^{N-N_u} + \cdots + A_d + I)B_d & \end{bmatrix} \in R^{N \cdot n_x \times N_u \cdot n_u} \quad (15)$$

If the SSMPC controller determines control in each successive moment $k, k+1, k+2, \dots$, to control the plant there are used only n_u first rows of the calculated matrix K_{SSMPC} specifying increment of the control signal in the current step k

$$\Delta u(k) = \Delta u(k|k) = K_{\text{SSMPC}_1} [Y_{SP}(k) - Y_0(k)], K_{\text{SSMPC}_1} \in R^{n_u \times N \cdot n_y} \quad (16)$$

3.3 Properties of LQR and SSMPC algorithms

Comparing the cost functions of the LQR controller (6) and SSMPC controller (10) there are significant differences.

First, controllers use different variables when determining the control matrix. In the case of the LQR controller, the weight matrices correspond to the control $u(k)$ and internal states $x(k)$. Thereupon, in the case of the underactuated system, it does not have a direct impact on the weight of the outputs of the process if the model is not implemented with some of the internal states corresponding directly to its outputs. SSMPC controller determine the optimum control signal using output error $e(k)$ and increments of control signal $\Delta u(k)$. In contrary to LQR algorithm the choice of SSMPC cost function allows to adjust weights directly for each output. Unfortunately, because the second minimized parameter use control increments, it is not possible to put weight directly on the control signal.

The second significant difference is a form of the feedback that is used. The solution of the LQR is the control matrix, which on the basis of the current state determines the subsequent control. The criterion function SSMPC sets

the control over the prediction horizon N which means, that in the case of the inaccurate model, the control signal should be calculated at every step k .

It should also be noted that if the reference trajectory is known over the prediction horizon, the SSMPC controller allows to enter information about its course into the algorithm and thus to adjust the control signal to the upcoming changes.

An important issue for the design LQR and SSMPC controllers is the choice of the coefficients in weight matrices. They are diagonal matrices which determine the significance of internal states, output signals, control signals or control signal increments for the cost function.

In case of the LQR controller, the initial weights can be selected on the basis of the Bryson's rule [3]

$$Q_{ii} = \frac{1}{x_{iacc}^2}, i = 1, \dots, n_x, R_{jj} = \frac{1}{u_{jacc}^2}, j = 1, \dots, n_u \quad (17)$$

where: x_{iacc} , u_{jacc} - maximum acceptable values of the i -th internal state and j -th control signal.

Elements outside diagonals are set to 0. Determination of the maximum acceptable values in (17) depends on the requirements and limitations put on the proposed control system. In a situation when there are no limitations imposed on some signals or internal states, the corresponding weights based on Bryson's rule should be equal to 0. If values of the diagonal weight matrix are chosen using Bryson's rule, the impact of each signal on the cost function is averaged.

It should be emphasized that the Bryson's rule has been proposed for LQR control algorithm but a similar analysis can be performed for SSMPC. Therefore, we propose to set the weighting matrices as follows

$$\Psi_{ii} = \frac{1}{e_{iacc}^2}, i = 1, \dots, n_y, \Lambda_{jj} = \frac{1}{\Delta u_{jacc}^2}, j = 1, \dots, n_u \quad (18)$$

where: y_{jacc} and Δu_{jacc} - maximum acceptable values of the i -th output error signal and j -th control signal increment. Elements outside the diagonals should be set to 0.

Bryson's rule usually is a starting point for a variety of iterative methods (trial-and-error) which attempt to achieve desired properties of the control system.

4 Simulations

4.1 Controllers setup

For comparison, both controllers were designed to control the inverted pendulum described by the set of equations (1). Weight matrices of LQR and SSMPC controllers were selected according to Bryson's rule.

Settings of LQR controller were chosen as follows

$$Q = \begin{bmatrix} 16 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, R = [0.44], K_{\text{LQR}} = [-5.8924 \ -8.9697 \ 61.4290 \ 8.8536],$$

Settings of SSMPC controller were chosen as follows

$$N = 4 \text{ s}, N_u = 2 \text{ s}, \Psi(k) = \begin{bmatrix} 500 & 0 \\ 0 & 100 \end{bmatrix}, A(k) = [4]$$

During tests, both controllers were complemented with a two-point limit on the maximum input signal, representing the constraints of the actuator and a saturation of a control signal to limit its value to $\pm 1.5 \text{ Nm}$. Additionally SSMPC controller had a limit on the control signal increment in the range of $\pm 0.5 \text{ Nm}$.

4.2 Quality criteria of the control system

The quality of the each control system was analyzed in the time domain using following criteria:

$e_{x \text{ stat}}$ i $e_{\theta \text{ stat}}$ - **Steady state error** of the cart linear position and the pendulum angular position.

$e_{x \text{ max}}$ i $e_{\theta \text{ max}}$ - **Maximum error** of the cart linear position and the pendulum angular position.

t_r - **Transient response time** which is the time between the beginning of input change (t_0) and the moment when the error signal of chosen output reaches a fixed value inside a boundary $\delta = 5\%e_{\text{max}}$. Transient response time was chosen as the higher value for both outputs of the system.

$u_{\text{max}} = \max(u(k))$ - **The maximum value of the control signal** $u(k)$ generated by the controller after the moment of change (t_0) of the reference trajectory or introduction of disturbances.

$IAC = \int_{t_0}^{t_r} |u(t)| dt$ - **Absolute integral control** quality index.

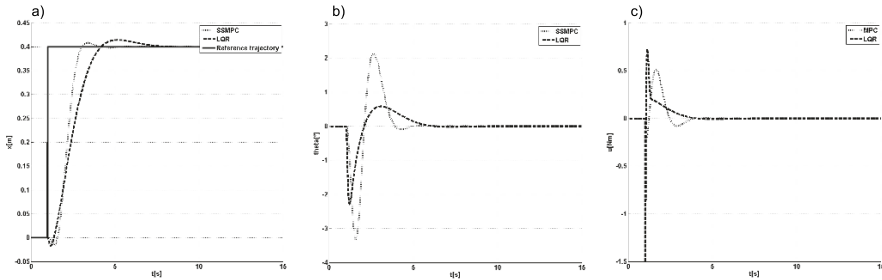
4.3 Comparison of SSMPC and LQR control systems

Trajectory tracking. In order to compare the quality of SSMPC and LQR controllers, there were checked their step responses of cart position setpoint change of 0.4 m .

As shown in Tab. 2 transient response time for LQR controller was equal to 5.14 s while for SSMPC controller 3.73 s . LQR controller worked slightly better in terms of pendulum angle adjustment, the maximum absolute deflection of the pendulum for LQR controller was 2.31° while for SSMPC controller it was equal to 3.71° .

Table 2. Quality indices - step response

Parameter \rightarrow	t_r	$e_x \text{ stat}$	$e_x \text{ max}$	$e_\theta \text{ stat}$	$e_\theta \text{ max}$	u_{max}	IAC
	[s]	[m]	[m]	[deg]	[deg]	[Nm]	[Nm·s]
SSMPC	3.730	0.000	0.418	0	3.71	0.579	0.486
LQR	5.140	0.000	0.418	0	2.31	1.500	0.435

**Fig. 2.** Step response of LQR and SSMPC control systems - a) position of the cart, b) pendulum angular position, c) control signal

In Fig. 2 attention should be paid to the shape of the control signal. In the case of LQR controller, this signal is changing very rapidly, especially when the setpoint change occurs, while the control signal of SSMPCS controller is smoother. This may be important during the further selection of actuators because these devices are adversely affected by sudden changes of the control signal.

Disturbance rejection. Then there was compared the response of both controllers to the appearance of interference in the control signal. Given disturbance took the form of the pulse with a value of 0.5 Nm, lasting 1 s. The obtained results are shown in Fig. 3 and gathered in Tab. 3. In this case, transient response time for LQR controller was equal to 3.3 s while for SSMPC controller t_r was equal to 2.74 s.

Table 3. Quality indices - response to disturbances

Parameter \rightarrow	t_r	$e_x \text{ stat}$	$e_x \text{ max}$	$e_\theta \text{ stat}$	$e_\theta \text{ max}$	u_{max}	IAC
	[s]	[m]	[m]	[deg]	[deg]	[Nm]	[Nm·s]
SSMPC	2.740	0	0.045	0.000	1.610	0.746	0.729
LQR	3.300	0	0.043	0.000	0.530	0.655	0.609

Analyzing the quality indices given in the Tab. 3, it was found that the values of both $e_x \text{ max}$ and $e_\theta \text{ max}$ are higher for SSMPC controller comparing with LQR

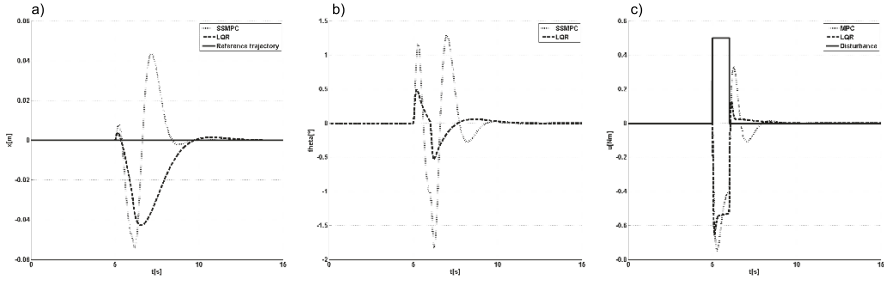


Fig. 3. The response to disturbance in the control signal of LQR and SSMPC control systems - a) position of the cart, b) pendulum angular position, c) control signal

controller. This may be due to the fact that to determine the value of the control signal, SSPMC analyzes the whole prediction horizon. As a result, elimination of the influence of the interference is averaged for the entire time horizon. Therefore the prediction horizon should not be too long, because it slows down the reaction of a controller to disturbances in the system. This problem does not occur in the LQR control system, because it determines the control only on the basis of an actual situation, without prediction.

5 Conclusions

From the comparison of the obtained results for different control cases, it is clearly visible that LQR algorithm is performing better in the control task with the disturbance rejection, while SSMPC controller gives better results in the case of the trajectory tracking task. Moreover, SSMPC controller is characterized by smooth changes in the control signal while LQR controller generates rapid changes of the control signal, which is the important drawback because it affects significantly actuator wear.

The size of the control matrix of the LQR controller depends only on the number of internal states of the object. SSMPC controller sets the control matrix for the whole prediction horizon. Therefore size of SSMPC control matrix depends not only on the number of internal states but increases proportionally with increasing the prediction horizon and shortening the sampling period. This involves the increasing number of calculations that must be performed at any time the control signal is calculated. This limits the possible applications of SSMPC algorithm and is dependent on the capabilities of the used industrial controller.

The proposed future work includes the experimental research at the laboratory stand AMIRA LIP100 with an application of modern industrial controllers, the investigation of robustness of described LQR and SSMPC control systems of an inverted pendulum, the investigation of Bryson's rule and iterative choice of weighting matrices in cost functions using multi-objective optimization methods.

References

1. Tedrake, R.: Underactuated Robotics: Algorithms for Walking, Running, Swimming, Flying, and Manipulation (Course Notes for MIT 6.832). Downloaded on 14.01.2017 from <http://underactuated.mit.edu/>
2. Jacobson, D. H. , Martin, D. H. and Pachter, M., Geveci, T.: Extensions of Linear-Quadratic Control Theory, Lecture Notes in Control and Information Sciences, Vol. 27, Springer-Verlag, Berlin Heidelberg, (1980)
3. Bryson, A.E., Ho, Y.-C.: Applied Optimal Control - Optimization, Estimation, and Control, Hemisphere Publishing Corporation, Washington (1975).
4. Fantoni, I., Lozano, R.: Non-linear Control for Underactuated Mechanical Systems, Springer-Verlag, London (2002).
5. Prasad, L.B., Tyagi, B. and Gupta, H. O.: Optimal Control of Nonlinear Inverted Pendulum System Using PID Controller and LQR: Performance Analysis Without and With Disturbance Input, International Journal of Automation and Computing, 11(6), 661–670, (2014).
6. El-Hawwary, M. I., Elshafei, A. L., Emara, H. M. and Fattah, H. A. A.: Adaptive Fuzzy Control of the Inverted Pendulum Problem, IEEE Transactions on Control Systems Technology, 14(6), 1135–1144, (2006).
7. Chmielewski, A. , Guminski, R., Maciag, P. and Maczka, J.: The Use of Fuzzy Logic in the Control of an Inverted Pendulum, Dynamical Systems: Theoretical and Experimental Analysis, 182, (Springer Proceedings in Mathematics & Statistics), 71–82, (2016)
8. Yang, C., Li, Z., Cui, R. and Xu, B.: Neural Network-Based Motion Control of an Underactuated Wheeled Inverted Pendulum Model, IEEE Transactions on Neural Networks and Learning Systems, 25(11), 2004–2016, (2014).
9. Delgado, S., Kotyczka, P.: Energy Shaping for Position and Speed Control of a Wheeled Inverted Pendulum in Reduced Space, Automatica, 74, 222-229, (2016).
10. Huang, J., Guan, Z. H., Matsuno, T., Fukuda, T. and Sekiyama K.: Sliding-Mode Velocity Control of Mobile-Wheeled Inverted-Pendulum Systems, IEEE Transactions on Robotics, 26(4), 750–758, (2010).
11. Wang, J.J.: Simulation Studies of Inverted Pendulum Based On PID Controllers, Simulation Modelling Practice and Theory, 19, 440-449 (2011).
12. Magni, L., Scattolini, R., Astrom, K.J.: Global Stabilization of the Inverted Pendulum Using Model Predictive Control, 15th IFAC World Congress, IFAC Proceedings Volumes, 35(1),141-146, (2002).
13. Valencia-Palomo, G., Rossiter, J.A.: Efficient Suboptimal Parametric Solutions to Predictive Control for PLC Applications, Control Engineering Practice, 19(7), 732–743,(2011).
14. Ozana, S., Pies, M., Slanina, Z. and Hajovsky, R.: Design and Implementation of LQR Controller for Inverted Pendulum by Use of REX Control System, 12th International Conference on Control, Automation and Systems, (2012).
15. Kucera, V.: The Discrete Riccati Equation of Optimal Control, Kibernetika, 8(5):430447, (1972).
16. Tatjewski, P.: Advanced Control of Industrial Processes - Structures and Algorithms, Springer (2007).
17. Tarnawski, J.: Implementation of Predictive Control Algorithm in Programmable Logic Controllers (in polish), Pomiary Automatyka Robotyka, 6, 100–107 (2013).
18. Lawrynczuk, M.: Computationally Efficient Model Predictive Control Algorithms - A Neural Network Approach, Springer (2014)

19. Kwakernaak, H., Sivan, R.: Linear Optimal Control Systems. WileyInterscience, New York, (1972).
20. Lundh, M., Molander, M.: State-Space Models in Model Predictive Control, ABB (1999).
21. Wang, L.: Model Predictive Control System Design and Implementation Using MATLAB, Springer (2009).

Design of modified PID controllers for 3D crane control

Cellmer A., Bartosz Banach, Piotrowski R.

Faculty of Electrical and Control Engineering, Gdansk University of Technology, Gdansk,
Poland,

agata.cellmer@gmail.com, banach.bartoszl@gmail.com,
robert.piotrowski@pg.gda.pl

Bartosz Banach

Abstract. From the control viewpoint, 3D crane is a dynamic, nonlinear and multidimensional electromechanical system. In this paper, five control systems using a set-point weighted PID controllers (modified controllers) are designed. These structures and properties are presented. Optimization process of controllers settings based on integral performance indices is made. Simulation tests of control systems are presented. Comparison of integral indices for control systems using classical PID controllers and modified PID controllers on physical model is presented.

Keywords: modified PID controller, control system, optimization, 3D crane

1 Introduction

3D crane is used for large and heavy objects transport, e.g., in production halls and sea ports. Its target is moving load from point to point in workspace simultaneously minimalizing deviations of payload.

In this paper, five control systems for 3D crane on physical model, made by INTECO [1], are designed. Three of them are being designed to control the carriage moving in three axes (X, Y and Z). Another two controllers are being designed to control deviations of the payload (X and Y axes). Modified PID controllers are employed.

The issue is still current and is considered in different research works. 3D crane can be controlled from Matlab/Simulink [1] or LabVIEW [2]. It gives great opportunities in the implementation of various control algorithms. In [3] 3D crane control issue has been taken. Authors suggested to control 3D crane using fuzzy controller. Control system using LQR is designed in [4] and PID controller is applied in [5].

2 Description of 3D crane

2.1 Physical model

3D crane is a nonlinear, multidimensional, dynamic electromechanical system. The object can be divided into a hardware and software. The hardware includes a frame, which size is 1 x 1 x 1 m, rail with a trolley on which the payload is mounted, DC (Direct Current) motors and incremental encoders (Fig. 1) [6].

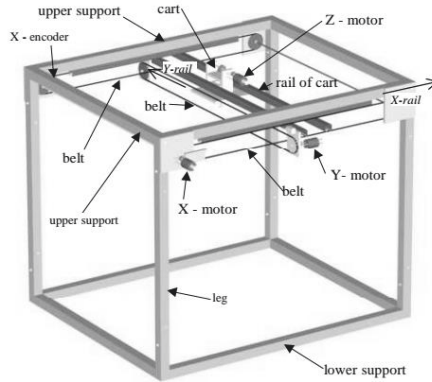


Fig. 1. 3D crane construction [6]

The payload can be lifted and lowered by line lift. Rotary motion is converted into plane motion using belt transmission. There are five incremental encoders measuring five variables: the payload position in horizontal axis (X) and vertical axis (Y), the lift-line length (Z axis) and two deviation angles of the payload (γ and δ angles). High resolution of encoders (4096 pulses per revolution) allows measuring displacement and deviation of the payload with high precision. The payload can be moved by three DC motors in three axes (X, Y, Z). There are also Power Interface Unit and data acquisition board (RT-DAC/PCI) which are connected to the PC. This interface amplifies control signals sent from the PC to DC motors. It also converts signals from incremental encoders into a digital 16-bit form which can be read by the PC. 3D crane can be controlled by using an external toolbox in Matlab/Simulink which is supplied by INTECO [6].

2.2 Input and output signals

The 3D crane input signals are sequence of PWM (Pulse Width Modulation) to control of DC motors. However, output signals are: position of the payload in three axes (X, Y, Z) and deviation angles (γ , δ). Furthermore, there is information from limit switches. These signals are sending to the PC by external data acquisition board (Fig. 2).



Fig. 2. Data exchange diagram

3 Design of modified PID controllers

3.1 Set-point weighted PID controller

The classical PID controller bases on control error signal $e(t)$ for each correction term. Modified structures of this controller can also base on controlled signal $y(t)$. This is achieved through adding a second degree of freedom. It is necessary to specify the value of two additional parameters: α and β , where α is set-point weighting parameter for proportional controller and β is set-point weighting parameter for derivative controller [7]. The general structure of set-point weighted PID controller is illustrated in Fig. 3.

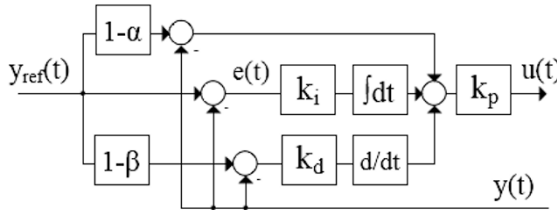


Fig. 3. Structure of set-point weighted PID controller [7]

where k_p , k_i and k_d are non-negative proportional gain, integral gain and derivative gain, respectively; $y_{ref}(t)$ is reference signal; $u(t)$ is control signal; $y(t)$ is controlled signal; $e(t)$ is control error signal.

The control signal is of the form:

$$u(t) = k_p \left((1 - \alpha)y_{ref}(t) + k_i \int_0^t y_{ref}(\tau) d\tau + (1 - \beta)k_d \frac{dy_{ref}(t)}{dt} \right) - k_p \left(y(t) + k_i \int_0^t y(\tau) d\tau + k_d \frac{dy(t)}{dt} \right) \tag{1}$$

In table 1 α and β parameters for modified PID controllers are presented [7].

Table 1. Parameters α and β for set-point weighted PID controllers [7]

Set-point weighting parameters		Controller structure
α	β	
0	0	PID
0	1	PI-D
1	0	ID-P
1	1	I-PD
$0 < \alpha < 1$	1	PI-PD

3.2 PI-D controller

When $\alpha=0$ and $\beta=1$, PI-D controller is obtained. It makes that proportional and integral parts respond for control error signal $e(t)$, while derivative part bases on controlled signal $y(t)$. This configuration avoids a large overshoot, which is the result of a step change in control error, involving a sudden change in set-point. At the same time it achieves a high quality control for a step change in disturbances because derivative term is not working on the control error signal [8].

3.3 I-PD controller

When $\alpha=\beta=1$, then I-PD controller is obtained. In this structure, only integral part of controller works based on control error signal $e(t)$, while proportional and derivative parts respond for controlled signal $y(t)$. It makes that this controller provides lower overshoot by avoiding a sudden of step change caused by proportional and derivative parts, compared to PI-D controller. This configuration makes that setting time for step change in reference signal is extended relative to PI-D controller [8].

3.4 ID-P controller

When $\alpha=1$ and $\beta=0$, ID-P controller is obtained. In this structure, integral and derivative parts respond for control error signal $e(t)$. However, proportional part works based on controlled signal $y(t)$. Compared to I-PD controller, ID-P controller response for step change in set-point is short due to the fact that derivative part bases on controlled signal. Simultaneous, this structure avoids overshoot caused by proportional part. Consequently, it provides a smooth reference tracking response compared to PI-D controller [7].

3.5 PI-PD controller

When α value is between 0 and 1, and $\beta=1$, PI-PD controller is obtained. It makes that integral part responds for control error signal $e(t)$, while derivative part responds for controlled signal $y(t)$. Parameter α is responsible for weighting set-point given to proportional part. When α is closer to a value of 1, PI-PD controller reacts similar to I-PD controller. When α is closer to a value of 0, PI-PD controller reacts similar to PI-D controller. It makes that PI-PD controller is a compromise between I-PD and PI-D controllers [7].

3.6 Control systems

Control systems for 3D crane were designed and presented in Fig. 4. Limits on the minimum and maximum value of control signal $u(t)$ and rate of that signal $\Delta u(t)$ were included.

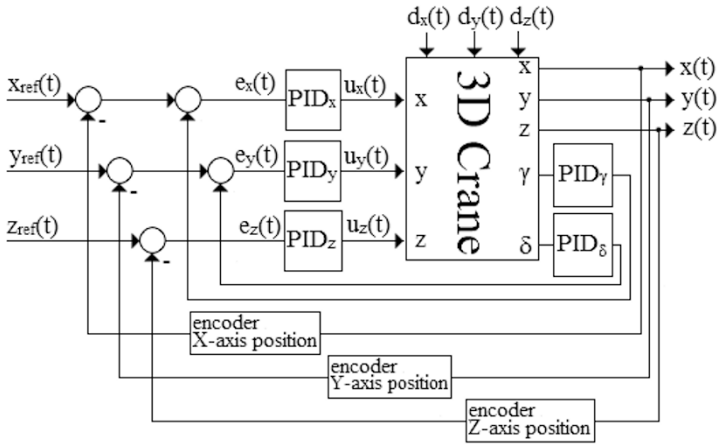


Fig. 4. 3D crane control systems

where $d_x(t)$, $d_y(t)$, $d_z(t)$ are disturbances signals (e.g. friction, inertia).

4 Simulation tests and results analysis

4.1 Optimization of PID controller parameters

Optimization process was carried out in Matlab environment using quasi-Newton method [9]. The following optimization problem was formulated as:

- decision variables:

$$k_{p_{X,Y,Z,\gamma,\delta}}, k_{i_{X,Y,Z,\gamma,\delta}}, k_{d_{X,Y,Z,\gamma,\delta}} \quad (2)$$

- constraints:

$$0 \leq k_{p_{X,Y,Z,\gamma,\delta}} \leq 60 \quad (3)$$

$$0 \leq k_{i_{X,Y,Z,\gamma,\delta}} \leq 100 \quad (4)$$

$$0 \leq k_{d_{X,Y,Z,\gamma,\delta}} \leq 30 \quad (5)$$

For optimization process, as performance functions, four integral indices based on control error signal were selected:

- Integral of Square Error – ISE:

$$\min I_1 = \min \int_0^\infty e^2(t)dt \quad (6)$$

- Integral of Absolute Error – IAE:

$$\min I_2 = \min \int_0^\infty |e(t)|dt \quad (7)$$

- Integral of Time and Absolute Error – ITAE:

$$\min I_3 = \min \int_0^{\infty} t|e(t)|dt \quad (8)$$

- Integral of Sum of Square Error and Weighted Square of Derivative Error:

$$\min I_4 = \min \int_0^{\infty} (e^2(t) + 0.5(\dot{e}(t))^2)dt \quad (9)$$

Optimization process was carried out in the following way:

- X axis position and deviation controllers parameters optimization,
- Y axis position and deviation controllers parameters optimization,
- Z axis position controller parameters optimization.

Optimization process was made on mathematical model, supplied by INTECO [6]. This model is described by nonlinear differential equations, e.g., sine, cosine and quadratic functions. A detailed description can be found in [4]. The best results were implemented on the physical model. In table (2)–(3) obtained controllers settings are presented.

Simulation tests on physical model were carried out for two cases:

- case I: X axis and γ angle – PI-PD controllers, Y axis and δ angle – PI-PD controllers, Z axis – ID-P controller,
- case II: X axis – I-PD controller, Y axis – ID-P controller, Z axis – PI-D controller, γ angle – PI-PD controller, δ angle – PI-D controller.

Table 2. Controllers settings (case I)

Controller	Application	k_p	k_i	k_d
PI-PD	X axis	8.3317	0.9605	0
PI-PD	Y axis	8.5583	0.3659	0.2729
ID-P	Z axis	31.4108	1.0308	0
PI-PD	γ angle	9.4954	0.05	0.05
PI-PD	δ angle	1.2885	0.0469	0.05

Table 3. Controllers settings (case II)

Controller	Application	k_p	k_i	k_d
I-PD	X axis	7.1967	1.8982	0.5041
ID-P	Y axis	6.9642	0.6817	0
PI-D	Z axis	17.3198	0.006	0.0375
PI-PD	γ angle	9.4954	0.05	0.05
PI-D	δ angle	2.7855	0.8709	0.05

4.2 Implementation of control systems on physical model

In case I, the characteristic of position in X axis (Fig. 5) is without overshoot. Almost zero steady state control error and short setting time were achieved. The characteristic of deviation in this axis (Fig. 6) is not oscillatory. However, there is a small steady state control error. The characteristic of position in Y axis (Fig. 5) is without overshoot and smooth reference signal tracking is reached. Setting time is relatively long. The characteristic of deviation in this axis (Fig. 6) is oscillatory, but amplitude is slight. The characteristic of position in Z axis (Fig. 7) is without overshoot. Almost zero steady state control error and short setting time were achieved. It proves that control effects are satisfying.

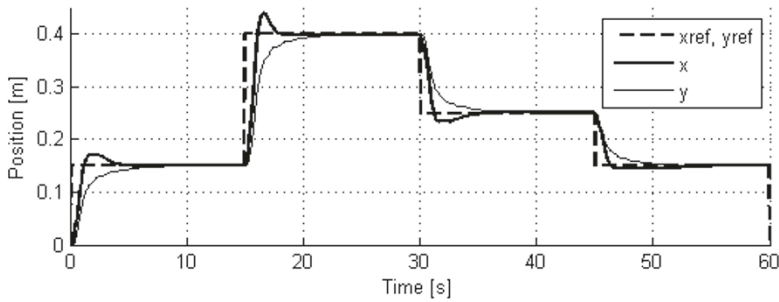


Fig. 5. Characteristic of horizontal and vertical position for PI-PD controllers (physical model)

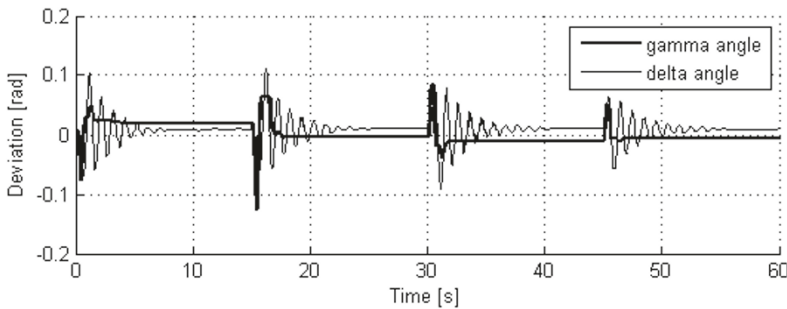


Fig. 6. Characteristic of deviations of the payload for PI-PD controllers (physical model)

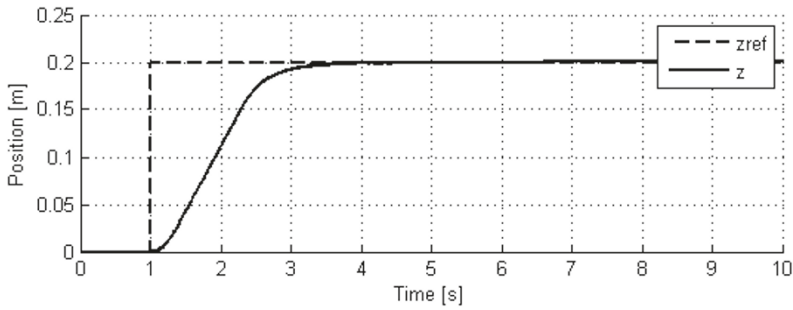


Fig. 7. Characteristic of Z axis position for ID-P controller (physical model)

In case II, the characteristic of position in X axis (Fig. 8) is oscillatory. However, short setting time was reached. The characteristic of deviation in this axis (Fig. 9) is not oscillatory. There is also a small steady state control error. The characteristic of position in Y axis (Fig. 8) is without overshoot and smooth reference signal tracking is achieved. Setting time is relatively long, similar like in case I. The characteristic of deviation in this axis (Fig. 9) is not oscillatory and steady state control error is almost zero. The characteristic of position in Z axis (Fig. 10) is without overshoot. Almost zero steady state control error and short setting time were achieved, similar like in case I for this axis. It proves that control effects are satisfying.

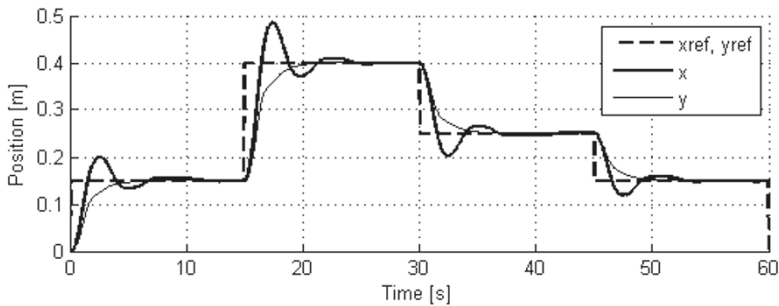


Fig. 8. Characteristic of horizontal position for I-PD controller and vertical position for ID-P controller (physical model)

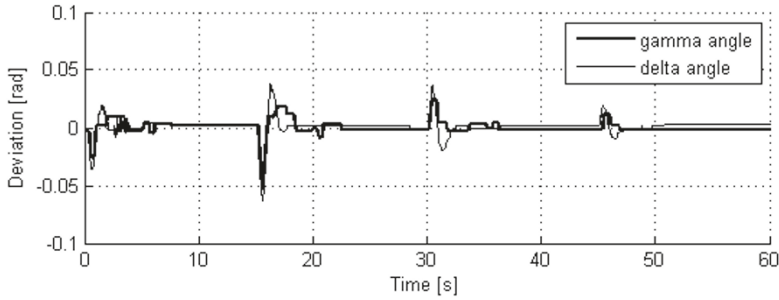


Fig. 9. Characteristic of deviations of the payload in X axis for PI-PD controller and Y axis for PI-D controller (physical model)

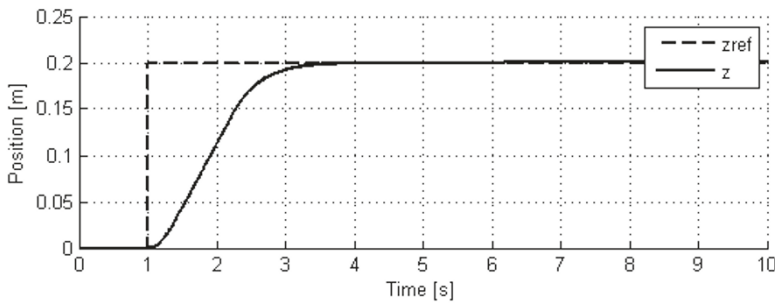


Fig. 10. Characteristic of Z axis position for PI-D controller (physical model)

In table 4, comparison of performance functions for case I is presented. For X axis position and deviation control, I_3 indicator value is less for control systems with modified PID controllers. For Y axis position and deviation control, I_3 indicator value is also less for control systems with modified PID controllers. The value of indicator I_4 is not much greater for control system with modified PID controller for Z axis position control. In consequence, setting time is slightly longer.

Table 4. Comparison of integral indices for control systems with classical and modified PID controllers (case I)

Application	Controller	Indicator	Modified PID	Classical PID
X axis and γ angle	PI-PD	I_3	0.6502	2.368
Y axis and δ angle	PI-PD	I_3	0.4452	6.153
Z axis	ID-P	I_4	0.02699	0.0131

In table 5, comparison of performance functions for case II is illustrated. For X axis position control, I_1 indicator value is a little bit less for control system with modified PID controller and control effects are satisfactory. Deviation control in this axis is effective and I_3 indicator value is also less. For Y axis position and deviation control, satisfactory control results and effective payload stabilization are reached. The value of I_4 indicator for deviation control in Y axis is minimally less for control sys-

tem with modified PID controller. For Z axis position control, I_3 indicator value is less for control system with modified PID controller which means effective reference signal tracking and short setting time were achieved.

Table 5. Comparison of integral indices for control systems with classical and modified PID controllers (case II)

Application	Controller	Indicator	Modified PID	Classical PID
X axis	I-PD	I_1	0.05295	0.6664
Y axis	ID-P	I_2	0.8253	0.5831
Z axis	PI-D	I_3	0.2887	0.3093
γ angle	PI-PD	I_3	0.2717	0.6722
δ angle	PI-D	I_4	0.0008244	0.00094

5 Conclusions

In this paper, 3D crane control systems were presented and control results analysis was done. Optimization process using several integral indices based on control error signal was carried out. The evaluation of obtained control results was presented. The implementation of modified PID controllers for 3D crane control systems was successful. In some configurations, values of integral indices were less compared to control systems using classical PID controllers. In many cases, the application of set-point weighted PID controller allows to obtain more effective control results. This is a right possibilities extension of classical PID controller.

References

1. 3D Crane Getting Started, <http://www.inteco.com.pl>.
2. Kuck R., Pauluk M.: *LabVIEW in control 3D crane*. Measurements Automation Robotics 6/2010, 17-22 (in Polish).
3. Antić D., Jovanović Z., Perić S., Nikolić S., Milojković M., Milošević M. (2012). *Anti-swing fuzzy controller applied in a 3D crane system*. Engineering, Technology & Applied Science Research, Vol. 2, No. 2, pp. 196-200.
4. Pauluk M. (2016). *Optimal and robust control of 3D crane*. Przegląd Elektrotechniczny 2/2016, 206-212.
5. Thuan N.Q., Veselý V. (2011). *Robust decentralized controller design for 3D crane*. Proc. of the 18th International Conference on Process Control, June 14-17, Tatranská Lomnica, Slovakia.
6. 3D Crane Installation Manual, <http://www.inteco.com.pl>.
7. Rajinikanth V., Latha K. (2012). *Setpoint weighted PID controller tuning for unstable system using heuristic algorithm*. Archives of Control Sciences, Vol. 22(LVIII), No. 4, pp. 481-505.
8. Świder Z., Trybus L.: *Extended PID algorithm for the industrial temperature controller with self-tuning*. Measurements Automation Robotics 2/2013, 432-435 (in Polish).
9. Nocedal J., Wright S. J. (2006). *Numerical Optimization. Second Edition*. Springer Science+Business Media, New York, 135-163.

Part II
Optimization, Estimation and Prediction

Lyapunov Matrices Approach to the Parametric Optimization of Time Delay System with PID Controller

Józef Duda

AGH University of Science and Technology, Krakow, Poland

Abstract. In the paper a Lyapunov matrices approach to the parametric optimization problem of time delay system with a PID-controller is presented. The value of integral quadratic performance index is equal to the value of Lyapunov functional for the initial function of the time delay system. The Lyapunov functional is determined by means of the Lyapunov matrix. The example of the inertial system with time delay and with a PID-controller is presented.

Keywords: neutral system, Lyapunov matrix, Lyapunov functional

1 Introduction

The Lyapunov functionals are used to test the stability of systems [13], in calculation of the robustness bounds for uncertain time delay systems [10], in computation of the exponential estimates for the solutions of time delay systems [9,11]. The Lyapunov quadratic functionals are also used to calculation of a value of a quadratic performance index in the process of the parametric optimization for time delay systems, see for instance [3,5,6,7]. One constructs a functional for a system with time delay with a given time derivative whose is equal to the negative definite quadratic form of a system state. The value of that functional at the initial state of time delay system is equal to the value of a quadratic performance index. There are two methods of determination of the Lyapunov functionals. The first method of determination of the Lyapunov functional for a time delay system was presented by Repin [14] and developed by Duda, see for example [2,4]. The second method of determination of the Lyapunov functional for a time delay by means of Lyapunov matrices is presented for example in [11-15].

In the paper a Lyapunov matrices approach to the parametric optimization problem of a time delay system with the PID controller is presented. This paper extends the results presented in [3,5,6,7].

2 Calculation of the Performance Index Value

Let us consider a neutral system

$$\begin{cases} \frac{dx(t)}{dt} - C \frac{dx(t-r)}{dt} = Ax(t) + Bx(t-r) \\ x(t_0 + \theta) = \varphi(\theta) \end{cases} \quad (1)$$

for $t \geq t_0$, $\theta \in [-r, 0]$, $r > 0$, where $x(t) \in \mathbb{R}^n$, $A, B, C \in \mathbb{R}^{n \times n}$, function $\varphi \in PC^1([-r, 0], \mathbb{R}^n)$ - the space of piece-wise continuously differentiable vector valued functions defined on the segment $[-r, 0]$ with the uniform norm $\|\varphi\|_{PC^1} = \sup_{\theta \in [-r, 0]} \|\varphi(\theta)\|$.

We assume that the matrix C is Schur stable i.e. its eigenvalues lie in the interior of the unit disk of the complex plane.

The theorems of existence, continuous dependence and uniqueness of solutions of equation (1) are given in [8,10]. The controllability of the systems with time delay is presented in [12]. The stabilization problem of time delay system is considered in [1].

In parametric optimization problem is often used the index

$$J = \int_{t_0}^{\infty} x^T(t, \varphi) W x(t, \varphi) dt \quad (2)$$

where $x(t, \varphi)$ is a solution of system (1), with the initial function $\varphi \in PC^1([-r, 0], \mathbb{R}^n)$ for $t \geq t_0$.

The value of the performance index (2) is equal to the value of the Lyapunov functional v for the initial function $\varphi \in PC^1([-r, 0], \mathbb{R}^n)$, see [5]

$$\begin{aligned} J = v(\varphi) = & \varphi^T(0)[U(0) - U(-r)C - C^T U^T(-r) + C^T U(0)C] \varphi(0) + \\ & + 2\varphi^T(0) \int_{-r}^0 [U(-\theta - r) - C^T U(-\theta)] [B\varphi(\theta) + C \frac{d}{d\theta} \varphi(\theta)] d\theta + \\ & + \int_{-r}^0 \int_{-r}^0 [B\varphi(\theta) + C \frac{d}{d\theta} \varphi(\theta)]^T U(\theta - \xi) [B\varphi(\xi) + C \frac{d}{d\xi} \varphi(\xi)] d\theta d\xi \end{aligned} \quad (3)$$

The Lyapunov matrix U is obtained by solution of the set of equations (4)-(6)

$$\frac{d}{d\xi} U(\xi) - \frac{d}{d\xi} U(\xi - r)C = U(\xi)A + U(\xi - r)B \quad (4)$$

$$U(-\xi) = U^T(\xi) \quad (5)$$

$$\begin{aligned} -W = & A^T U(0) + U(0)A - A^T U(-r)C - C^T U^T(-r)A + \\ & + B^T U^T(-r) + U(-r)B - B^T U(0)C - C^T U(0)B \end{aligned} \quad (6)$$

Formula (5) implies

$$U(\xi - r) = U^T(-\xi + r) \quad (7)$$

and (4) takes a form

$$\frac{d}{d\xi} U(\xi) - \frac{d}{d\xi} U^T(-\xi + r)C = U(\xi)A + U^T(-\xi + r)B \quad (8)$$

We introduce a new variable $\tau = -\xi + r$. The term (8) for a new variable has a form

$$\frac{d}{d\tau}U^T(-\tau + r) - C^T \frac{d}{d\tau}U(\tau) = -A^T U^T(-\tau + r) - B^T U(\tau) \quad (9)$$

One obtains a set of equations

$$\begin{cases} \frac{d}{d\xi}U(\xi) - \frac{d}{d\xi}U^T(-\xi + r)C = U(\xi)A + U^T(-\xi + r)B \\ \frac{d}{d\xi}U^T(-\xi + r) - C^T \frac{d}{d\xi}U(\xi) = -A^T U^T(-\xi + r) - B^T U(\xi) \end{cases} \quad (10)$$

We introduce a new function

$$Z(\xi) = U^T(-\xi + r) \quad (11)$$

The set of equations (10) can be written in a form

$$\begin{cases} \frac{d}{d\xi}U(\xi) - \frac{d}{d\xi}Z(\xi)C = U(\xi)A + Z(\xi)B \\ \frac{d}{d\xi}Z(\xi) - C^T \frac{d}{d\xi}U(\xi) = -A^T Z(\xi) - B^T U(\xi) \end{cases} \quad (12)$$

or in an equivalent form

$$\begin{cases} \frac{d}{d\xi}U(\xi) - C^T \frac{d}{d\xi}U(\xi)C = U(\xi)A - B^T U(\xi)C + Z(\xi)B - A^T Z(\xi)C \\ \frac{d}{d\xi}Z(\xi) - C^T \frac{d}{d\xi}Z(\xi)C = -B^T U(\xi) + C^T U(\xi)A - A^T Z(\xi) + C^T Z(\xi)B \end{cases} \quad (13)$$

for $\xi \in [0, r]$ with the initial conditions $U(0)$ and $Z(0)$.

The formulas (5) and (11) imply

$$U(-r) = U^T(r) = Z(0) \quad (14)$$

Taking into account (14) one can write the algebraic property (6) in a form

$$\begin{aligned} -W &= A^T U(0) + U(0)A - A^T Z(0)C - C^T Z^T(0)A + \\ &+ B^T Z^T(0) + Z(0)B - B^T U(0)C - C^T U(0)B \end{aligned} \quad (15)$$

Equation (13) can be written in a form

$$\begin{bmatrix} \frac{d}{d\xi} \text{col}U(\xi) \\ \frac{d}{d\xi} \text{col}Z(\xi) \end{bmatrix} = \mathcal{H} \begin{bmatrix} \text{col}U(\xi) \\ \text{col}Z(\xi) \end{bmatrix} \quad (16)$$

Solution of (16) is given

$$\begin{bmatrix} \text{col}U(\xi) \\ \text{col}Z(\xi) \end{bmatrix} = \begin{bmatrix} \Phi_{11}(\xi) & \Phi_{12}(\xi) \\ \Phi_{21}(\xi) & \Phi_{22}(\xi) \end{bmatrix} \begin{bmatrix} \text{col}U(0) \\ \text{col}Z(0) \end{bmatrix} \quad (17)$$

where a matrix $\Phi(\xi) = \begin{bmatrix} \Phi_{11}(\xi) & \Phi_{12}(\xi) \\ \Phi_{21}(\xi) & \Phi_{22}(\xi) \end{bmatrix}$ is a fundamental matrix of system (16).

We determine the initial conditions $\text{col}U(0)$, $\text{col}Z(0)$. The term (11) implies $Z(r) = U^T(0)$. From (17) we obtain

$$\text{col}Z(r) = \text{col}U^T(0) = \Phi_{21}(r)\text{col}U(0) + \Phi_{22}(r)\text{col}Z(0) \quad (18)$$

In this way we attain the set of algebraic equations which enables us to calculate the initial conditions of (17).

$$\begin{aligned} A^T U(0) + U(0)A - A^T Z(0)C - C^T Z^T(0)A + B^T Z^T(0) + \\ + Z(0)B - B^T U(0)C - C^T U(0)B = -W \end{aligned} \quad (19)$$

$$\Phi_{21}(r)\text{col}U(0) - \text{col}U^T(0) + \Phi_{22}(r)\text{col}Z(0) = 0 \quad (20)$$

3 First Order System with a PID Controller

Let us consider a first order system with time delay and with a PID-controller

$$\begin{cases} \frac{dx(t)}{dt} = -\frac{1}{T}x(t) + \frac{k_0}{T}u(t-r) \\ u(t) = -px(t) - \frac{1}{T_i} \int_0^t x(\xi)d\xi - T_d \frac{dx(t)}{dt} \\ x(\theta) = \varphi_1(\theta) \end{cases} \quad (21)$$

$t \geq 0$, $x(t) \in \mathbb{R}$, $\theta \in [-r, 0]$, $k_0, T, T_i, T_d, p \in \mathbb{R}$, $r \geq 0$. The parameter k_0 is a gain of a plant, T is a system time constant, φ_1 - is the initial function. The parameters of a PID controller p, T_i and T_d are tuning parameters.

One introduces the state variables $x_1(t)$ and $x_2(t)$ as follows

$$\begin{cases} x_1(t) = x(t) \\ x_2(t) = \frac{1}{T_i} \int_c^t x(\xi)d\xi \\ x_2(0) = x_{20} \end{cases} \quad (22)$$

where c is a constant real number

One can reshape (21) to a form

$$\begin{cases} \frac{dx_1(t)}{dt} + \frac{k_0 T_d}{T} \frac{dx_1(t-r)}{dt} = -\frac{1}{T}x_1(t) - \frac{k_0 p}{T}x_1(t-r) - \frac{k_0}{T}x_2(t-r) \\ \frac{dx_2(t)}{dt} = \frac{1}{T_i}x_1(t) \\ x_1(\theta) = \varphi_1(\theta) \\ x_2(\theta) = \frac{1}{T_i} \int_0^\theta \varphi_1(\xi)d\xi = \psi(\theta) \\ x_2(0) = x_{20} \end{cases} \quad (23)$$

for $t \geq 0$ and $\theta \in [-r, 0]$.

In parametric optimization problem we use the performance index

$$J = \int_0^\infty [x_1(t, \varphi_1, \psi), x_2(t, \varphi_1, \psi)] W \begin{bmatrix} x_1(t, \varphi_1, \psi) \\ x_2(t, \varphi_1, \psi) \end{bmatrix} dt \quad (24)$$

where $W = \begin{bmatrix} w_1 & 0 \\ 0 & w_2 \end{bmatrix} > 0$ is a symmetric positive definite matrix and $x_i(t, \varphi_1, \psi)$ for $i = 1, 2$ is a solution of (23) for initial function (φ_1, ψ)

Problem 1. Determine the tuning parameters of a PID controller, p , T_i and T_d , whose minimize an integral quadratic performance index (24).

System (23) can be written as a matrix equation (1), where matrices A , B and C have a form

$$A = \begin{bmatrix} -\frac{1}{T} & 0 \\ \frac{1}{T_i} & 0 \end{bmatrix} \quad (25)$$

$$B = \begin{bmatrix} -\frac{k_0 p}{T} & -\frac{k_0}{T} \\ 0 & 0 \end{bmatrix} \quad (26)$$

$$C = \begin{bmatrix} -\frac{k_0 T_d}{T} & 0 \\ 0 & 0 \end{bmatrix} \quad (27)$$

System of equations (16) takes a form

$$\frac{d}{d\xi} \begin{bmatrix} U_{11}(\xi) \\ U_{21}(\xi) \\ U_{12}(\xi) \\ U_{22}(\xi) \\ Z_{11}(\xi) \\ Z_{21}(\xi) \\ Z_{12}(\xi) \\ Z_{22}(\xi) \end{bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} U_{11}(\xi) \\ U_{21}(\xi) \\ U_{12}(\xi) \\ U_{22}(\xi) \\ Z_{11}(\xi) \\ Z_{21}(\xi) \\ Z_{12}(\xi) \\ Z_{22}(\xi) \end{bmatrix} \quad (28)$$

where

$$Q_{11} = \begin{bmatrix} -\frac{T+k_0^2 p T_d}{T^2-k_0^2 T_d^2} & 0 & \frac{T^2}{T^2-k_0^2 T_d^2} & 0 \\ -\frac{k_0^2 T_d}{T^2} & -\frac{1}{T} & 0 & \frac{1}{T_i} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (29)$$

$$Q_{12} = \begin{bmatrix} -\frac{k_0 T_d+k_0 p T}{T^2-k_0^2 T_d^2} & \frac{k_0 T T_d}{T^2-k_0^2 T_d^2} & 0 & 0 \\ 0 & -\frac{k_0 p}{T} & 0 & 0 \\ -\frac{k_0}{T} & 0 & 0 & 0 \\ 0 & -\frac{k_0}{T} & 0 & 0 \end{bmatrix} \quad (30)$$

$$Q_{21} = \begin{bmatrix} \frac{k_0 T_d+k_0 p T}{T^2-k_0^2 T_d^2} & 0 & -\frac{k_0 T T_d}{T^2-k_0^2 T_d^2} & 0 \\ \frac{k_0}{T} & 0 & 0 & 0 \\ 0 & 0 & \frac{k_0 p}{T} & 0 \\ 0 & 0 & \frac{k_0}{T} & 0 \end{bmatrix} \quad (31)$$

$$Q_{22} = \begin{bmatrix} \frac{T+k_0^2 p T_d}{T^2-k_0^2 T_d^2} - \frac{\frac{T^2}{T_i}}{T^2-k_0^2 T_d^2} & 0 & 0 \\ 0 & 0 & 0 \\ \frac{k_0^2 T_d}{T^2} & 0 & \frac{1}{T} - \frac{1}{T_i} \\ 0 & 0 & 0 \end{bmatrix} \quad (32)$$

Initial conditions of system (28) one obtains solving the set of algebraic equations (19)-(20), which take a form

$$\begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} U_{11}(0) \\ U_{21}(0) \\ U_{12}(0) \\ U_{22}(0) \\ Z_{11}(0) \\ Z_{21}(0) \\ Z_{12}(0) \\ Z_{22}(0) \end{bmatrix} = \begin{bmatrix} -w_1 \\ 0 \\ 0 \\ -w_2 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (33)$$

where

$$G_{11} = \begin{bmatrix} -\frac{2(T+k_0^2 p T_d)}{T^2} & \frac{1}{T_i} & \frac{1}{T_i} & 0 \\ -\frac{k_0^2 T_d}{T^2} & -\frac{1}{T} & 0 & \frac{1}{T_i} \\ -\frac{k_0^2 T_d}{T^2} & 0 & -\frac{1}{T} & \frac{1}{T_i} \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (34)$$

$$G_{12} = \begin{bmatrix} -\frac{2k_0(T_d+pT)}{T^2} & \frac{2k_0 T_d}{T T_i} & 0 & 0 \\ -\frac{k_0}{T} & -\frac{k_0 p}{T} & 0 & 0 \\ -\frac{k_0}{T} & -\frac{k_0 p}{T} & 0 & 0 \\ 0 & -\frac{2k_0}{T} & 0 & 0 \end{bmatrix} \quad (35)$$

$$G_{21} = \begin{bmatrix} \Phi_{51}(r) - 1 & \Phi_{52}(r) & \Phi_{53}(r) & \Phi_{54}(r) \\ \Phi_{61}(r) & \Phi_{62}(r) & \Phi_{63}(r) - 1 & \Phi_{64}(r) \\ \Phi_{71}(r) & \Phi_{72}(r) - 1 & \Phi_{73}(r) & \Phi_{74}(r) \\ \Phi_{81}(r) & \Phi_{82}(r) & \Phi_{83}(r) & \Phi_{84}(r) - 1 \end{bmatrix} \quad (36)$$

$$G_{22} = \begin{bmatrix} \Phi_{55}(r) & \Phi_{56}(r) & \Phi_{57}(r) & \Phi_{58}(r) \\ \Phi_{65}(r) & \Phi_{66}(r) & \Phi_{67}(r) & \Phi_{68}(r) \\ \Phi_{75}(r) & \Phi_{76}(r) & \Phi_{77}(r) & \Phi_{78}(r) \\ \Phi_{85}(r) & \Phi_{86}(r) & \Phi_{87}(r) & \Phi_{88}(r) \end{bmatrix} \quad (37)$$

$$\Phi(\xi) = [\Phi_{ij}(\xi)] \quad (38)$$

for $i, j = 1, \dots, 8$ is a fundamental matrix of solutions of (28).

The value of the performance index (24) is equal to the value of the Lyapunov functional (3) for an initial function (φ_1, ψ) .

We calculate the value of the performance index for the function φ_1 given by a term

$$\varphi_1(\theta) = \begin{cases} x_0 & \text{for } \theta = 0 \\ 0 & \text{for } \theta \in [-r, 0) \end{cases} \quad (39)$$

Formula (23) implies $\psi(\theta) = \int_0^\theta \varphi_1(\xi) d\xi$. For the function φ_1 , given by a term (39), $\psi(\theta) = 0$ for $\theta \in [-r, 0]$.

After calculations one obtains

$$J = [x_0 \ x_{20}] \begin{bmatrix} U_{11}(0) & U_{12}(0) \\ U_{21}(0) & U_{22}(0) \end{bmatrix} \begin{bmatrix} x_0 \\ x_{20} \end{bmatrix} \quad (40)$$

We search for optimal parameters of PID controller which minimize the index (40). Optimization results are given in Table 1. These results are obtained by means of Matlab function *fminsearch* for $x_0 = 1$, $x_{20} = 0.5$, $w_1 = 1$, $w_2 = 1$, $k_0 = 1$ and $T = 5$.

Table 1. Optimization results

time delay	popt	1/Ti opt	Td opt	Index value
0.5	8.7152	0.8815	2.3490	5.8360
1.0	4.4092	0.4506	2.3671	6.6486
1.5	2.9776	0.3069	2.3860	7.4337
2.0	2.2652	0.2349	2.4060	8.1890
2.5	1.8408	0.1917	2.4274	8.9142
3.0	1.5606	0.1629	2.4500	9.6096
3.5	1.3627	0.1422	2.4739	10.2762
4.0	1.2163	0.1267	2.4992	10.9154
4.5	1.0935	0.1144	2.5603	11.5293
5.0	1.0160	0.1049	2.5536	12.1180

Figure 1 shows elements of the Lyapunov matrix $U(\xi)$ for optimal values of the PID Controller parameters.

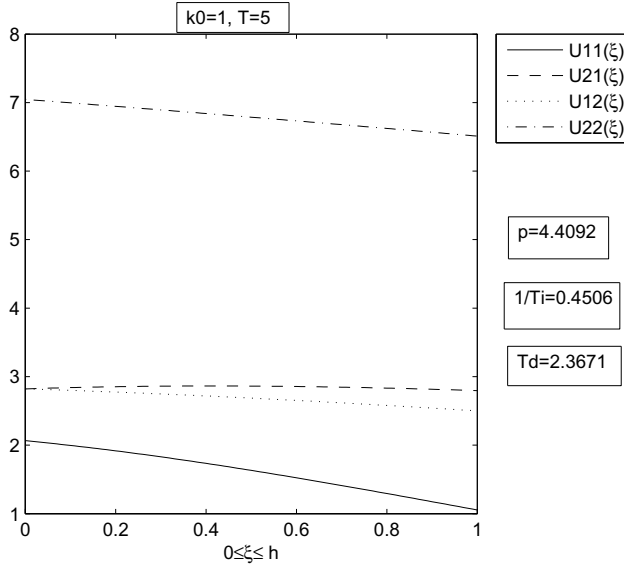


Fig. 1. Elements of Lyapunov matrix $U(\xi)$

4 Conclusions

In the paper a Lyapunov matrices approach to the parametric optimization problem of a time delay system with a PID controller is presented. The value of integral quadratic performance index is equal to the value of Lyapunov functional for the initial function of time delay system. The Lyapunov functional is determined by means of the Lyapunov matrix.

References

1. Baranowski, J.: Stabilization of a Second Order System with a Time Delay Controller. *Control Engineering and Applied Informatics* 18, 11-19 (2016)
2. Duda, J.: A Lyapunov functional for a neutral system with a time-varying delay. *Bulletin of the Polish Academy of Sciences Technical Sciences* 61, 911-918 (2013)
3. Duda, J.: Lyapunov matrices approach to the parametric optimization of time-delay systems. *Archives of Control Sciences* 25, 279-288 (2015)
4. Duda, J.: A Lyapunov functional for a neutral system with a distributed time delay. *Mathematics and Computers in Simulation* 119, 171-181 (2016)
5. Duda, J.: Lyapunov matrices approach to the parametric optimization of a neutral system. *Archives of Control Sciences* 26, 81-93 (2016)
6. Duda, J.: Lyapunov matrices approach to the parametric optimization of a time delay system with a PD controller, *Proceedings of the 2016, 17th International Carpathian Control Conference (ICCC 2016)*, 172 - 177 (2016)

7. Duda, J.: Lyapunov Matrices Approach to the Parametric Optimization of a Time Delay System with a PI Controller, Proceedings of the 21st International Conference on Methods and Models in Automation and Robotics (MMAR 2016), 1206-1210 (2016)
8. Górecki, H., Fuksa, S., Grabowski, P., Korytowski, A.: Analysis and Synthesis of Time Delay Systems. John Wiley & Sons, Chichester, New York, Brisbane, Toronto, Singapore, (1989).
9. Kharitonov, V.: On the uniqueness of Lyapunov matrices for a time-delay system. *Systems & Control Letters* 61, 397-402 (2012)
10. Kharitonov, V.: Time-delay systems. Basel, Birkhauser (2013)
11. Kharitonov, V.: An extension of the prediction scheme to the case of systems with both input and state delay. *Automatica* 50, 211-217 (2014)
12. Klamka, J.: Controllability of Dynamical Systems. Kluwer Academic Publishers, Dordrecht (1991)
13. Medvedeva, V., Zhabko, A.: Synthesis of Razumikhin and Lyapunov-Krasovskii approaches to stability analysis of time-delay systems. *Automatica* 51, 372-377 (2015)
14. Repin, Yu.: Quadratic Lyapunov functionals for systems with delay. *Prikl. Mat. Mekh.* 29, 564-566 (1965)

Sensitivity Analysis of Optimal Control Parabolic Systems with Retardations

Adam Kowalewski¹, Zbigniew Emirsajłow², Jan Sokółowski³, and Anna Krakowiak⁴

¹ AGH University of Science and Technology,
Institute of Automatics and Biomedical Engineering, al. Mickiewicza 30, 30-059
Cracow, Poland, ako@agh.edu.pl

² Institute of Control Engineering, West Pomeranian University of Technology,
ul. 26 kwietnia 10, 71-126 Szczecin, emirsaj@zut.edu.pl

³ Institut Elie Cartan, UMR 7502 Nancy-Université-CNRS-INRIA,
Laboratoire de Mathématiques, Université Henri Poincaré,
Nancy I, B.P. 239, 54506 Vandoeuvre lés Nancy Cedex, France,
Jan.Sokolowski@univ-lorraine.fr

⁴ Institute of Mathematics, Technical University of Cracow,
ul. Warszawska 24, 31-155 Cracow, Poland, skrakowi@pk.edu.pl

Abstract. The first order sensitivity analysis is performed for a class of optimal control problems for time lag parabolic equations in which retarded arguments appear in the integral form with $h \in (0, b)$ in the state equations and with $k \in (0, c)$ in the Neumann boundary conditions. The optimality system is analyzed with the respect to a small parameter. The directional derivative of the optimal control is obtained as a solution to an auxiliary optimization problem. The control constraints for the auxiliary optimization problem are received.

Keywords: sensitivity analysis, optimal control, parabolic systems with retardations

1 Introduction

Sensitivity analysis of optimal control systems constitute very important field of automatic control, mechanical engineering, mathematical control theory and optimization theory. Such problems have been investigated, discussed and subsequently published for a wide class of distributed parameter systems.

For example, in the papers [1], [2], [4] the first order sensitivity analysis was performed for a class of optimal control problems for parabolic [1] and hyperbolic [4] systems and for parabolic equations with constant time lags [2] and with integral time lags [3]. Moreover, the bibliography concerning the sensitivity analysis of optimal control problems was presented.

In particular, the purpose of the paper [4] is to perform sensitivity analysis of optimal control problems described by the hyperbolic equation. The small

parameter describes the size of an imperfection in the form of a small hole or cavity in the geometrical domain of integration. The initial state equation in the singularly perturbed domain is replaced by the equation in a smooth domain. Consequently the imperfection is replaced by its approximation defined by a suitable Steklov's type differential operator. For approximate optimal control problems the well-posedness is verified. One term asymptotics of optimal control are derived and motivated for the approximate model. The crucial role in the arguments is played by the so called "hidden regularity" of boundary traces generated by hyperbolic solutions.

In this paper the first order sensitivity analysis is performed for a class of optimal control problems for time lag parabolic equations in which retarded arguments appear in the integral form with $h \in (0, b)$ in the state equations and with $k \in (0, c)$ in the Neumann boundary conditions.

Sufficient conditions for the existence of a unique solution for such retarded parabolic systems are presented [5].

The cost function has a quadratic form. The time horizon is fixed. Finally, we impose some constraints on the boundary control. Making use of Lion's framework [6] necessary and sufficient conditions of optimality with the quadratic cost function and constrained control are derived for the Neumann problem.

We consider an optimal control problem in the domain with small geometrical defect. The size of the defect is measured by small parameter $\rho > 0$ (Fig. 1). The presence of the defect results in the singular perturbation of the parabolic time lag equation. Such a perturbation is transformed to the regular perturbation in the truncated domain Ω_R for any $R > \rho > 0$ (Fig. 2). The domains $B(\rho)$, Ω_ρ , Ω_R are defined in [1], [2], [4] respectively. We perform the sensitivity analysis in the truncated domain using the Steklov-Poincaré operator defined on the circle Γ_R . The construction of asymptotic approximation for the Steklov-Poincaré operator is given in [8].

The optimal control problem in a singularly perturbed geometrical domain Ω_ρ is investigated with the respect to a small parameter $\rho > 0$. The one-term asymptotic expansion of optimal controls is derived. The auxiliary optimal control problem is formulated. The first term of the expansion of the order ρ^2 is uniquely determined as the optimal solution for such control problem. Using a property of conical differentiability of metric projection in L^2 spaces, the control constraints for the auxiliary control problem are obtained. Our approach is innovative and can lead to numerical procedures for determination of the first order approximations of the optimal controls.

2 Preliminaries

Consider now the distributed parameter system described by the following parabolic time lag equation

$$\left. \begin{aligned}
 \frac{\partial y}{\partial t} - \Delta y + \int_0^b y(x, t-h) dh &= f && \text{in } \Omega_\rho \times (0, T) \times (0, b) \\
 y(x, t') &= \Phi_0(x, t') && \text{in } \Omega_\rho \times (0, T) \times (0, b) \\
 & && \text{in } \Omega_\rho \times [-b, 0) \\
 \frac{\partial y}{\partial \eta} &= \int_0^c y(x, t-h) dh + v && \text{on } \Gamma \times (0, T) \times (0, c) \\
 y(x, t') &= \Psi_0(x, t') && \text{on } \Gamma \times [-c, 0) \\
 \frac{\partial y}{\partial \eta} &= 0 && \text{on } \Gamma_\rho \times (0, T) \\
 y(x, 0) &= y_0(x) && \text{in } \Omega_\rho
 \end{aligned} \right\} \quad (1)$$

where: $\Delta = \sum_{i=1}^n \frac{\partial^2}{\partial x_i^2}$, $\frac{\partial}{\partial \eta}$ is a normal derivative at Γ_ρ directed towards the exterior of Ω_ρ .

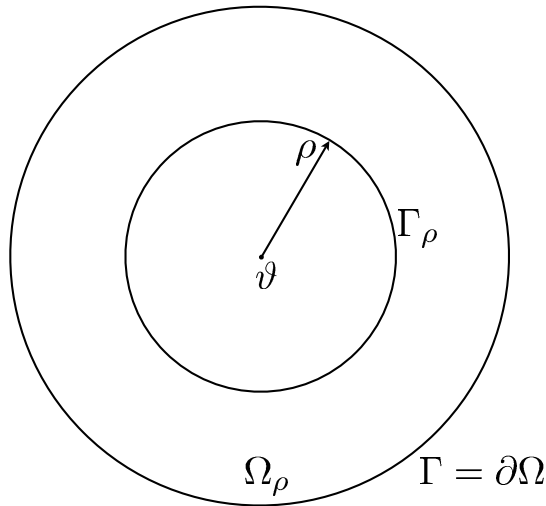


Fig. 1. The domain Ω_ρ in two spatial dimensions.

First we shall present sufficient conditions for the existence of a unique solution of the mixed initial-boundary value problem (1).

For this purpose for any pair of real numbers $r, s \geq 0$, we introduce the Sobolev space $H^{r,s}(Q)$ ([7]) defined by

$$H^{r,s}(Q) = H^0(0, T; H^r(\Omega)) \cap H^s(0, T; L^2(\Omega)) \tag{2}$$

which is a Hilbert space normed by

$$\left(\int_0^T \|y(t)\|_{H^r(\Omega)}^2 dt + \|y\|_{(H^s(0,T;L^2(\Omega)))}^2 \right)^{1/2} \tag{3}$$

where: the spaces $H^r(\Omega)$ and $H^s(0, T; L^2(\Omega))$ are defined in [7] (Vol.1, Chapter 1).

The case of parabolic system (1) in which time lags appear in the integral form with $h \in (0, b)$ in the state equations and with $k \in (0, c)$ in the Neumann boundary conditions is very sophisticated. We cannot use in this case a classical constructive method in the proof about the existence of a unique solution of the parabolic system (1), since the values of lower limits of integration are equal to zero. Consequently, using the transposition method and interpolation Theorem 14.1 ([7], Vol. 2, p.68) we can omit such restriction.

The following theorem about existence and uniqueness of solution is presented in [5].

Theorem 1 *Let y_0, Φ_0, v and f be given with $y_0 \in H^{1/2}(\Omega_\rho), \Phi_0 \in H^{3/2,3/4}(\Omega_\rho \times [-b, 0)), \Psi_0 \in L^2(\Gamma \times [-c, 0)), v \in L^2(\Gamma \times (0, T)),$ and $f \in (H^{1/2,1/4}(\Omega_\rho \times (0, T)))'$. Then there exists a unique solution $y \in H^{3/2,3/4}(\Omega_\rho \times (0, T))$ for the problem (1).*

3 Problem formulation. Optimality conditions.

We shall now consider the optimal boundary control problem in domains Ω_ρ and Ω_R respectively. Let us denote by $U = L^2(\Gamma \times (0, T))$ the space of controls. The time horizon T is fixed in our problem.

Let us consider in $\Omega_\rho \times (0, T)$ the following retarded parabolic equation

$$\left. \begin{aligned} \frac{\partial y}{\partial t} - \Delta y + \int_0^b y(x, t-h) dh &= f && \text{in } \Omega_\rho \times (0, T) \times (0, b) \\ y(x, t') &= \Phi_0(x, t') && \text{in } \Omega_\rho \times [-b, 0) \\ \frac{\partial y}{\partial \eta} &= \int_0^c y(x, t-h) dh + v && \text{on } \Gamma \times (0, T) \times (0, c) \\ y(x, t') &= \Psi_0(x, t') && \text{on } \Gamma \times [-c, 0) \\ \frac{\partial y}{\partial \eta} &= 0 && \text{on } \Gamma_\rho \times (0, T) \\ y(x, 0) &= y_0(x) && \text{in } \Omega_\rho; \text{ supp } y_0 \subset \Omega_R \end{aligned} \right\} \tag{4}$$

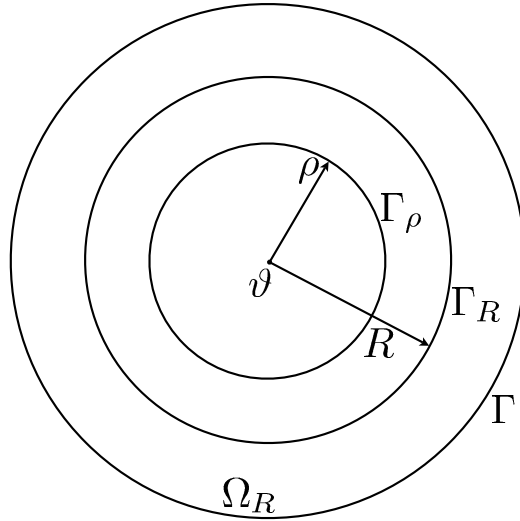


Fig. 2. The domain Ω_R .

The cost function is given by

$$I(v) = \frac{1}{2} \int_{\Omega_R} |y(x, T; v) - z_d|^2 dx + \frac{\alpha}{2} \int_0^T \int_{\Gamma} |v|^2 dx dt \tag{5}$$

Finally, we assume the following constraints on the control $v \in U_{ad}$:

$$U_{ad} = \{v \in L^2(\Gamma \times (0, T)), 0 \leq v(x, t) \leq 1\} \tag{6}$$

We consider in $\Omega_R \times (0, T)$ the following retarded parabolic equation

$$\left. \begin{aligned} \frac{\partial y}{\partial t} - \Delta y + \int_0^b y(x, t-h) dh &= f && \text{in } \Omega_R \times (0, T) \times (0, b) \\ y(x, t') &= \Phi_0(x, t') && \text{in } \Omega_R \times [-b, 0) \\ \frac{\partial y}{\partial \eta} &= \int_0^c y(x, t-h) dh + v && \text{on } \Gamma \times (0, T) \times (0, c) \\ y(x, t') &= \Psi_0(x, t') && \text{on } \Gamma \times [-c, 0) \\ \frac{\partial y}{\partial \eta} &= A_\rho(y) && \text{on } \Gamma_R \times (0, T) \\ y(x, 0) &= y_0(x) && \text{in } \Omega_R \end{aligned} \right\} \tag{7}$$

The cost function and constraints on the control are given by (5) and (6).

Result: The solution of the problem (7) (in the domain Ω_R) is a restriction of the solution of the problem (4) (in the domain Ω_ρ) to Ω_R . Thus, we have the possibility of replacing the singular perturbation of the domain $B(\rho)$ by the regular perturbation of boundary conditions on the artificial boundary Γ_R in a smaller domain Ω_R . We shall analyse the optimal boundary control problem (5)-(7) in the domain Ω_R . We assume that the small parameter $\rho > 0$ is fixed. The solving of the formulated optimization problem is equivalent to seeking for a control $v_0 \in U_{ad}$ such that $I(v_0) \leq I(v) \forall v \in U_{ad}$.

From Lions' scheme (Theorem 1.3 [6], p. 10) it follows that for $\alpha > 0$ a unique optimal control v_0 is characterized by the following condition

$$I'(v_0)(v - v_0) \geq 0 \quad \forall v \in U_{ad} \tag{8}$$

Using the form of the cost function (5) we can express (8) in the following form:

$$\int_{\Omega_R} (y(x, T; v_0) - z_d)(y(x, T; v) - y(x, T; v_0))dx + \frac{\alpha}{2} \int_0^T \int_{\Gamma} v_0(v - v_0)dxdt \geq 0 \quad \forall v \in U_{ad} \tag{9}$$

To simplify (9), we introduce the adjoint equation and for every $v \in U_{ad}$. we define the adjoint variable $p = p(v) = p(x, t; v)$ as the solution of the following equation

$$\left. \begin{aligned} -\frac{\partial p}{\partial t} - \Delta p + \int_0^b p(x, t+h)dh &= 0 && \text{in } \Omega_R \times (0, T) \times (0, b) \\ p(x, t; v) = 0 &&& \text{in } \Omega_R \times (T, T+b) \\ \frac{\partial p}{\partial \eta} = \int_0^c p(x, t+h)dh &&& \text{on } \Gamma \times (0, T) \times (0, c) \\ \frac{\partial p}{\partial \eta} = A_\rho(p) &&& \text{on } \Gamma_R \times (0, T) \\ p(x, T; v) = y(x, T; v) - z_d &&& \text{in } \Omega_R \end{aligned} \right\} \tag{10}$$

Theorem 2 *Let the hypothesis of Theorem 1 be satisfied. Then for given $z_d \in L^2(\Omega_R)$ and any $v_0 \in U_{ad}$, there exists a unique solution $p(v_0) \in H^{3/2, 3/4}(Q)$ for the problem (10).*

We simplify (9) using the adjoint equation (10). Consequently, after transformations we obtain the following formula

$$\int_0^T \int_{\Gamma} (p + \alpha v)(v - v_0)dxdt \geq 0 \quad \forall v \in U_{ad} \tag{11}$$

Theorem 3 For the problem (7) with the cost function (5) with $z_d \in L^2(\Omega_R)$ and $\alpha > 0$, and with constraints on the control (6), there exists a unique optimal control v_0 which satisfies optimality condition (11). Moreover, $v_0 = P_{U_{ad}} \left(-\frac{1}{\alpha} p \right)$ where $P_{U_{ad}}$ is the L^2 -projection onto the set of admissible controls.

4 The sensitivity of optimal controls

We obtain the following expansion for the Steklov-Poincaré operator A_ρ [8]:

$$\begin{aligned} A_\rho &= A_0 + \rho^2 B + O(\rho^4) \\ &\text{in the operator norm} \\ &\mathcal{L}(H^{1/2}(\Gamma_R), H^{-1/2}(\Gamma_R)) \end{aligned} \quad (12)$$

where: the remainder $O(\rho^4)$ is uniformly bounded on bounded sets in the space $H^{1/2}(\Gamma_R)$.

We shall search the expansion of the solution y^ρ in $\Omega_R \times (0, T)$:

$$y^\rho = y^0 + \rho^2 y^1 + \tilde{y} = y^0 + \rho^2 y^1 + \rho^4 \hat{y} \quad (13)$$

The Neumann boundary condition in (7) can be rewritten as

$$\frac{\partial y^\rho}{\partial \eta} = A_\rho(y^\rho) = A_0(y^\rho) + \rho^2 B(y^\rho) + \rho^4 \tilde{A}(y^\rho) \quad (14)$$

Substituting (13) into (14) we obtain

$$\begin{aligned} \frac{\partial y^0}{\partial \eta} + \rho^2 B \frac{\partial y^1}{\partial \eta} + \frac{\partial \tilde{y}}{\partial \eta} &= A_0(y^0 + \rho^2 y^1 + \tilde{y}) + \\ &+ \rho^2 B(y^0 + \rho^2 y^1 + \tilde{y}) + \rho^4 \tilde{A}(y^\rho) \end{aligned} \quad (15)$$

Comparing components with the same powers we get

$$\left. \begin{aligned} \rho^0 : \frac{\partial y^0}{\partial \eta} &= A_0(y^0) \\ \rho^2 : \rho^2 \frac{\partial y^1}{\partial \eta} &= \rho^2 [A_0 y^1 + B y^0] \end{aligned} \right\} \quad (16)$$

Thus it follows the following expansion of solutions:

Let us denote by y^0 the solution of the problem (7) corresponding to a given parameter $\rho = 0$.

Then, y^1 corresponding to a given parameter ρ^2 is a solution of the equation (18).

The optimization problem (in a singularly perturbed geometrical domain Ω_ρ) is examined with the respect to a small parameter $\rho > 0$. The one-term asymptotic expansion of optimal controls is derived. The first term of the expansion of the order ρ^2 is uniquely determined as the optimal solution to the auxiliary optimization problem.

Theorem 4 We have the following expansion of the optimal control in $L^2(\Gamma \times (0, T))$, with respect to the small parameter,

$$v_\rho = v_0 + \rho^2 q + o(\rho^2) \quad (17)$$

for $\rho > 0$.

We assume that ρ is a sufficiently small. The function q in (17) is a unique solution of the auxiliary optimal control problem:

The state equation

$$\left. \begin{aligned} \frac{\partial w}{\partial t} - \Delta w + \int_0^b w(x, t-h) dh &= 0 && \text{in } \Omega_R \times (0, T) \times (0, b) \\ w(x, t') &= \Phi_0(x, t') && \text{in } \Omega_R \times [-b, 0) \\ \frac{\partial w}{\partial \eta} &= \int_0^c w(x, t-h) dh + q && \text{on } \Gamma \times (0, T) \times (0, c) \\ w(x, t') &= \Psi_0(x, t') && \text{in } \Gamma \times [-c, 0) \\ \frac{\partial w}{\partial \eta} &= A_0(w) + B(y^0) && \text{on } \Gamma_R \times (0, T) \\ w(x, 0) &= 0 && \text{in } \Omega_R \end{aligned} \right\} \quad (18)$$

where: $w = y^1$.

The cost function

$$I(u) = \frac{1}{2} \int_{\Omega_R} |w(T, x)|^2 dx + \frac{\alpha}{2} \int_0^T \int_{\Gamma} |u|^2 dx dt \quad (19)$$

The adjoint equation

$$\left. \begin{aligned} -\frac{\partial z}{\partial t} - \Delta z + \int_0^b z(x, t+h) dh &= 0 && \text{in } \Omega_R \times (0, T) \times (0, b) \\ z(x, t; v) &= 0 && \text{in } \Omega_R \times (T, T+b) \\ \frac{\partial z}{\partial \eta} &= \int_0^c z(x, t+h) dh && \text{on } \Gamma \times (0, T) \times (0, c) \\ \frac{\partial z}{\partial \eta} &= A_0(z) + B(p^0) && \text{on } \Gamma_R \times (0, T) \\ z(x, T) &= w(x, T) && \text{in } \Omega_R \end{aligned} \right\} \quad (20)$$

where: $z = p^1$.

Then, the optimal control q is characterized by

$$\int_{\Omega_R} w(x, T; q)(w(x, T; u) - w(x, T; q))dx + \int_0^T \int_{\Gamma} q(u - q)dxdt \geq 0 \quad \forall u \in S_{ad} \tag{21}$$

where: S_{ad} is a set of admissible controls such that

$$\begin{aligned} S_{ad} = & \left\{ u \in L^2(\Gamma \times (0, T)) \mid \right. \\ & u(x, t) \geq 0 \text{ on the set } E_0 = \{(x, t) \mid v_0(x, t) = 0\}, \\ & u(x, t) < 0 \text{ on the set } E_1 = \{(x, t) \mid v_0(x, t) = 1\}, \\ & \left. \int_0^T \int_{\Gamma} (p_0 + \alpha v_0) u dxdt = 0 \right\}, \end{aligned} \tag{22}$$

where: p_0 is a adjoint state for $\rho = 0$, v_0 is a optimal solution for $\rho = 0$ such that $0 \leq v_0(x, t) \leq 1$.

We simplify (21) using the adjoint equation (20). After transformations we obtain the following optimality condition in the form of variational inequality

$$\int_0^T \int_{\Gamma} (z + \alpha q)(u - q) dxdt \geq 0 \quad \forall u \in S_{ad} \tag{23}$$

The latter condition means that the optimal control q is the metric projection of $-z/\alpha$ onto the set S_{ad} .

Theorem 5 *For the retarded parabolic problem*

$$\left. \begin{aligned} \frac{\partial w}{\partial t} - \Delta w + \int_0^b w(x, t - h) dh &= 0 && \text{in } \Omega_R \times (0, T) \times (0, b) \\ w(x, t') &= \Phi_0(x, t') && \text{in } \Omega_R \times [-b, 0) \\ \frac{\partial w}{\partial \eta} &= \int_0^c w(x, t - h) dh + q && \text{on } \Gamma \times (0, T) \times (0, c) \\ w(x, t') &= \Psi_0(x, t') && \text{in } \Gamma \times [-c, 0) \\ \frac{\partial w}{\partial \eta} &= A_0(w) + B(y^0) && \text{on } \Gamma_R \times (0, T) \\ w(x, 0) &= 0 && \text{in } \Omega_R \end{aligned} \right\} \tag{24}$$

with the cost function (19) with $w(T) \in L^2(\Omega_R)$ and $\alpha > 0$, and with constraints on the control (22), there exists a unique optimal control q which satisfies optimality condition (23).

5 Conclusions

The results presented in the paper can be treated as a generalization of the results obtained in [1]- [3] onto the case of integral time lag parabolic systems in which retarded arguments appear in the integral form with $h \in (0, b)$ in the state equations and with $k \in (0, c)$ in Neumann boundary conditions.

The asymptotic analysis of the time lag parabolic equation is performed. In particular the nonlocal boundary conditions on the interior boundary of the truncated domain Ω_R are defined. The optimal control problem in Ω_ρ is replaced by a family of optimal control problems in the truncated domain Ω_R . The main results are presented (Theorem 3), and the asymptotic formula with respect to ρ for optimal controls is established (Theorem 4). Necessary and sufficient conditions of optimality (23) are proved for a linear quadratic problem (18), (19), (22), (Theorem 5). In this paper we have considered the mixed initial boundary value problems of parabolic type. We can also consider similar optimal control problems for hyperbolic systems. Finally, we can consider such optimization problems for time lag hyperbolic systems.

Acknowledgements

The research presented here was carried out within the research programme AGH University of Science and Technology, No. 11.11.120.396.

References

1. Emirsajłow, Z., Kowalewski, A., Krakowiak, A., Sokołowski, J.: Sensitivity analysis of parabolic optimal control problems. Proceedings of the 10th IEEE International Conference on Methods and Models in Automation and Robotics 1, 37–42 (29 August - 1 September 2005, Międzyzdroje, Poland)
2. Emirsajłow, Z., Kowalewski, A., Krakowiak, A., Sokołowski, J.: Sensitivity analysis of time delays parabolic optimal control problems. Proceedings of the 12th IEEE International Conference on Methods and Models in Automation and Robotics 1, 105-109 (28-31 August 2006, Międzyzdroje, Poland)
3. Kowalewski, A., Emirsajłow, Z., Sokołowski, J., Krakowiak, A.: Sensitivity of optimal controls for time delay parabolic systems. Proceedings of the 21 IEEE International Conference on Methods and Models in Automation and Robotics, 511–515 (29 August - 1 September 2016, Międzyzdroje, Poland)
4. Kowalewski, A., Lasiecka, I., Sokołowski, J.: Sensitivity analysis of hyperbolic optimal control problems. Computational Optimization and Applications 52, 147-179 (2012)
5. Kowalewski, A., Krakowiak, A.: Optimal boundary control problem of retarded parabolic systems. Archives of Control Sciences 23, 261–279 (2013)
6. Lions, J.L.: Optimal Control of Systems Governed by Partial Differential Equations. Springer-Verlag, Berlin-Heidelberg (1971)
7. Lions, J.L., Magenes, E.: Non-Homogeneous Boundary Value Problems and Applications. Springer-Verlag, Berlin-Heidelberg, vol. 1 and 2 (1972)
8. Sokołowski, J., Zochowski, A.: Modelling of topological derivatives for contact problems. Numerische Mathematik 102, 145–179 (2005)

Optimality conditions for optimal control problems modeled by integral equations.

Wojciech Rafajłowicz

Wojciech Rafajłowicz is with the Department of Computer Engineering, Faculty of Electronics, Wrocław University of Science and Technology, Wybrzeże Wyspiańskiego 27, 50 370 Wrocław, Poland. wojciech.rafajlowicz@pwr.edu.pl

Abstract. In this paper a method of solving optimal control problems of systems described by Fredholm type integral equations was described. Special attention was given to difficult problems with point constraints imposed on the state of the system.

1 Introduction

The use of integral equations in control theory have paralleled differential equations up to the 1970s. Subsequent technological changes like the decline of analog computations, have the field of integral equations at a point of stagnation. Recently however we can notice growing interest in them. In this paper we show some contemporary results for the use of integral equations in control systems. Use of them can grossly simplify mathematics used and opens the road for applying new types of control signals unobtainable with differential equations only.

The first approach to optimal control of systems described by integral equations can be seen in a multipart paper by Vinokurov [8]. With some other assumption a form of maximum principle is obtained. This result was questioned by other authors, who pointed out some mistakes [9].

At some point in time interest in optimal control of systems modeled by integral equations had diminished mainly due to changes in computation methods.

In recent years there has been a huge increase in interest in integral equations and control of systems described by them.

Integro - algebraic model is a model where some parts of system are described by integral equation and others by algebraic equation. Recently a good example was given in the book [6]

In this paper we would consider solving some control problems for systems described by Fredholm type equations. Some work in this field, using maximum principle was done previously see [4],[10],[2]. Results published here were presented in PhD thesis [3].

2 Optimality conditions for Fredholm equations

Let us consider the following Fredholm equation

$$y(\bar{x}) = y_0 + \int_{\Omega} K(\bar{x}, \bar{x}') (y(\bar{x}') + ku(\bar{x}')) d\bar{x}', \quad (1)$$

where $\Omega \subset R^d$, $d \geq 1$ is a bounded set with smooth boundary Γ . Denote by $L_2(\Omega)$ the space of all real valued, squared integrable functions on Ω . Unless otherwise stated, all the functions are assumed to be from $L_2(\Omega)$.

This kind of a problem can arise from partial differential equations

$$A_x y(\bar{x}) - y(\bar{x}) = -u(\bar{x}), \tag{2}$$

for $\bar{x} \in \Omega$ and $y(\bar{x}) = 0$ for $\bar{x} \in \Gamma$. Where $A_x = \frac{\partial^2}{\partial x^2}$ lub $A_x = \frac{\partial^4}{\partial x^4}$. For equation (1) we can find $u^* \in L_2(\Omega)$ f for which the following functional attains its minimum.

$$Q(u) = \int_{\Omega} (y^*(\bar{x}) - y(\bar{x}))^2 d\bar{x} + \gamma \int_{\Omega} u^2(\bar{x}) d\bar{x}, \tag{3}$$

where $\gamma > 0$ is weighting factor. Concerning kernel K , we assume the following:

1. symmetry $K(\bar{x}, \bar{x}') = K(\bar{x}', \bar{x})$,
2. $\int K(\bar{x}, \bar{x}) d\bar{x} < \infty$,
3. nonnegative definiteness $\forall f \in L^2(\Omega) \int \int f(\bar{x}') K(\bar{x}, \bar{x}') f(\bar{x}) d\bar{x} d\bar{x}' \geq 0$.

From [7] we know that there exists the ortogonal and complete in $L^2(\Omega)$ sequence of functions $v_1(\bar{x}), v_2(\bar{x}), \dots$ such, that

$$K(\bar{x}, \bar{x}') = \sum_{i=0}^{\infty} \lambda_i v_i(\bar{x}) v_i(\bar{x}') \tag{4}$$

where λ_i 's are eigenvalues of the integral equation with kernel K , while $v_i(x)$, are the corresponding eigenfunctions

We assume that these functions are normalized $\int_{\Omega} v_k^2(\bar{x}) dx = 1, k = 1, 2, \dots$. Usually $\lambda_i = \frac{c}{i^2}$ or even $\lambda_i = \frac{c}{i^4}$. Thus (4) can be approximated with relatively short series.

We are looking for control function in the form of

$$u^*(\bar{x}) = \sum_{i=1}^{\infty} u_i v_i(\bar{x}). \tag{5}$$

We must notice that v_i is complete so it can be used to represent any function including $u(\bar{x})$ and

$$y(\bar{x}) = \sum_{i=1}^{\infty} y_i v_i(\bar{x}).$$

The above two series represent $u^*(\bar{x})$ and $y(\bar{x})$ from $L_2(\Omega)$ in the sense that they are convergent in the $L_2(\Omega)$ norm. By substituting (4) into (1) and assuming $y_0 = 0$ we obtain

$$\int_{\Omega} K(\bar{x}, \bar{x}') (y(\bar{x}') + ku(\bar{x}')) d\bar{x}' = \tag{6}$$

$$\sum_{i=1}^{\infty} \int_{\Omega} \left[\left(\sum_{j=1}^{\infty} \lambda_j v_j(\bar{x}) v_j(\bar{x}') \right) (y_i v_i(\bar{x}') + ku_i v_i(\bar{x}')) \right] d\bar{x}' =$$

$$= \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \lambda_j v_j(\bar{x}) \int_{\Omega} v_j(\bar{x}') v_i(\bar{x}') d\bar{x}' (y_i + k u_i)$$

because

$$\int_{\Omega} v_j(\bar{x}') v_i(\bar{x}') d\bar{x}' = \delta_{ij}$$

where δ_{ij} is Kronecker delta, for the right hand side of (6) we obtain

$$= \sum_{i=1}^{\infty} \lambda_i v_i(\bar{x}) (y_i + k u_i) \quad (7)$$

Finally the equations have the following form

$$\sum_{i=1}^{\infty} y_i v_i(\bar{x}) = \sum_{i=1}^{\infty} \lambda_i v_i(\bar{x}) (y_i + k u_i) \quad (8)$$

By multiplying both sides by $v_j(x)$, integrating with respect to Ω , assuming $\lambda_i \neq 1$, we obtain

$$\forall_{i=1,2,\dots} y_i = \lambda_i (y_i + k u_i) \quad (9)$$

$$\forall_{i=1,2,\dots} y_i = \frac{\lambda_i k u_i}{1 - \lambda_i}$$

Applying the Parseval equality to (3) and using (9) we obtain the following result.

Theorem 1 *Let for K assumptions 1)-3) hold. Then, the minimization problem (3), constrained by (1) can be obtained by to the following unconstrained*

$$\min_{u_i} \sum_{i=1}^{\infty} \left(y_i^* - \frac{\lambda_i k u_i}{1 - \lambda_i} \right)^2 + \gamma u_i^2 \quad (10)$$

and substituting its solution u_i^* , $i = 1, 2, \dots$ into (5).

Remark: notice that when there are no other constraints on u and y , then (10) splits to the sequence of the quadratic minimization problems for u_i 's.

3 Example problem

In the framework of L_2 theory that was presented in the previous section one can infer about convergence of truncated orthogonal sequence in L_2 norm. When a particular sequence of v_k 's is considered, the pointwise or even uniform convergence of truncated approximation can be proved.

We illustrate this statement by considering the optimal control problem for the Fredholm equation arising from the steady-state heat conduction problem.

We shall proceed in two steps. Firstly, we derive a solution quite formally, in the L_2 framework and then we shall prove the uniform convergence of the truncated series to a continuous function. The last fact is of importance when one consider pointwise constraints imposed on the system state, because one can not uniquely define $y(\bar{x}_0)$ for $y \in L_2(\Omega)$.

Optimal control problem (3) is equivalent to q quadratic programming problem (10) when $n = \infty$. For n sufficiently large it can be approximated as a finite dimensional quadratic programming problem.

Now let's consider a heat conduction equation in the steady state

$$a \frac{\partial^2 y}{\partial x^2} - by = -u(x) \tag{11}$$

$$y(0) = 0, y(\pi) = 0, a > 0, b \geq 0$$

We know that (see [5])

$$y(x) = \int_{\Omega} K(x, x')u(x')dx',$$

where the kernel K has the following form:

$$K(x, x') = \sum_{k=1}^{\infty} \frac{1}{ak^2 + b} \sin(kx)\sin(kx')$$

so $\lambda_k = \frac{1}{k^2 a + b}$, $v_i = \sqrt{2/\pi} \sin(ix)$.

The functional (3) can be expanded into series

$$Q = \sum_{k=1}^{\infty} (y_k^* - y_k)^2 + \gamma \sum_{k=1}^{\infty} u_k^2. \tag{12}$$

As an example we consider the following function on fig. 1

$$y^*(x) = -|x - \frac{\pi}{2}| + \frac{\pi}{2}. \tag{13}$$

By expanding this function into series $v_i = \sin(ix)$ and using five terms we obtain a sufficiently good approximation shown in fig. 2.

The stated optimal control problem become an unconstrained optimization problem.

$$\min_{u_i} \sum_{i=1}^n \left(y_i^* - \frac{\lambda_i u_i}{1 - \lambda_i} \right)^2 + \gamma u_i^2 \tag{14}$$

The minimizing sequence is unique ad it has the form

$$u_k^* = \frac{(\lambda_k - 1)\lambda_k}{\gamma - 2\gamma\lambda_k + \lambda_k^2(\gamma + 1)} y_k^*, \quad k = 1, 2, \dots, \tag{15}$$

where y_k^* is k-th Fourier coefficient of y^* .

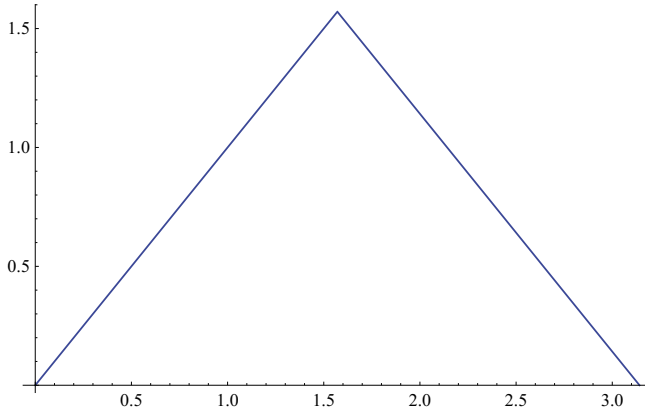


Fig. 1. Reference function y^* .

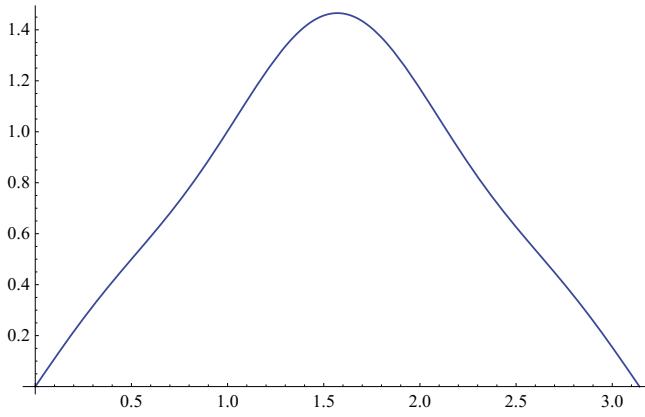


Fig. 2. Function y^* expanded into five term sinus series.

Denote by $\hat{y}_N(x)$ the system response to the truncated optimal input signal $u_N^*(x) = \sum_{k=1}^N u_k^* v_k(x)$, where

$$v_k(x) = \sqrt{\frac{2}{\pi}} \sin(kx). \tag{16}$$

According to (9), $\hat{y}_N(x)$ has the following form

$$\hat{y}_N(x) = \sum_{k=1}^N c_k y_k^* v_k(x) \tag{17}$$

where y_k^* 's are the Fourier coefficients of the reference function $y^*(x)$, while c_k 's are defined as follows

$$c_k^{-1} = \frac{\gamma}{\lambda_k^2} - \frac{2\gamma}{\lambda_k} + \gamma + 1 = \gamma(k^2 a + b)^2 - 2\gamma(k^2 a + b) + \gamma + 1. \tag{18}$$

Theorem 2 *If $y^* \in L_2(0, \pi)$ and $N \rightarrow \infty$ then the series (17) is convergent in the L_2 norm and also almost everywhere (a.e.) in $[0, \pi]$ with respect to the Lebesgue measure*

Proof. By assumption $y^* \in L_2(0, \pi)$ we have $\sum_k = 1^\infty y_k^{*2} < \infty$. From (18) it is clear that also the series $\sum_1^\infty c_k^2 y_k^{*2} < \infty$. Thus, convergence of \hat{y}_N in L_2 norm, as $N \rightarrow \infty$, to $\hat{y}(x) = \sum_{k=1}^N c_k y_k^* v_k(x)$ follows from the Parseval equality. The second statement follows immediately from the fact that $\hat{y} \in L_2(0, \pi)$ and from the Carleson theorem [1].

To prove the uniform convergence

$$\lim_{N \rightarrow \infty} \sup_{x \in (0, \pi]} |\hat{y}_N(x) - \hat{y}| = 0, \tag{19}$$

we need the following result from [11] Chapter 5.11 and the bibliography cited therein.

Theorem 3 (Channdy and Jollife (ChJ)) *If $\alpha_k \geq \alpha_{k+1} \rightarrow 0$ as $k \rightarrow \infty$, a necessary and sufficient condition for the uniform convergence of the series $\sum_{k=1}^\infty \alpha_k v_k(x)$ (v_k 's given by (16)) is that $k\alpha_k \rightarrow 0$ as $k \rightarrow \infty$.*

Theorem 4 *Let us assume that the reference signal $y^*(x)$ has uniformly convergent Fourier series in the basis (16). Furthermore, we assume that $y_{k+1}^* \geq y_k^*$. Then, the series (17) is uniformly convergent to \hat{y} in $(0, \pi]$ and $\hat{y}(x)$ is a continuous function.*

Proof. From the assumptions on y^* and ChJ theorem, we know that $k y_k^* \rightarrow 0$. With possible exception of a finite number of initial terms, sequence c_k 's is nonnegative, monotone and convergent to zero as $k \rightarrow \infty$. Thus, so is also the sequence $c_k \cdot y_k^*$, $k = 1, 2, \dots$ Additionally, the fact $k y_k^* \rightarrow 0$ implies $k c_k y_k^* \rightarrow 0$ as $k \rightarrow \infty$. Thus, we can invoke ChJ theorem once again to infer the uniform convergence sequence of continuous functions has a continuous function as its limit.

Let us take $a = 1, b = 1, k = 1$. When $\gamma = 0$ (no control cost), we get solution as in fig. 3. With $\gamma = 0.2$ result is a more flattened see fig. 4.

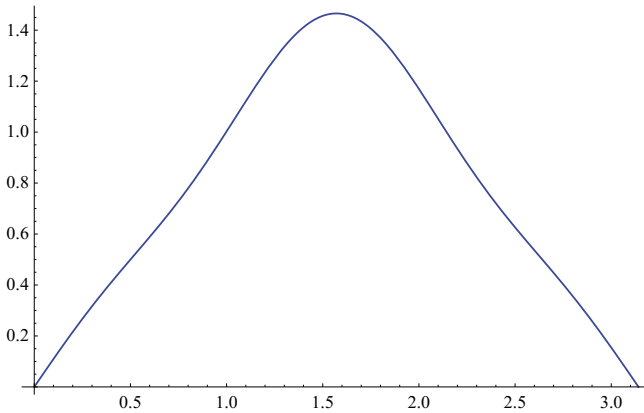


Fig. 3. Solution $\gamma = 0$.

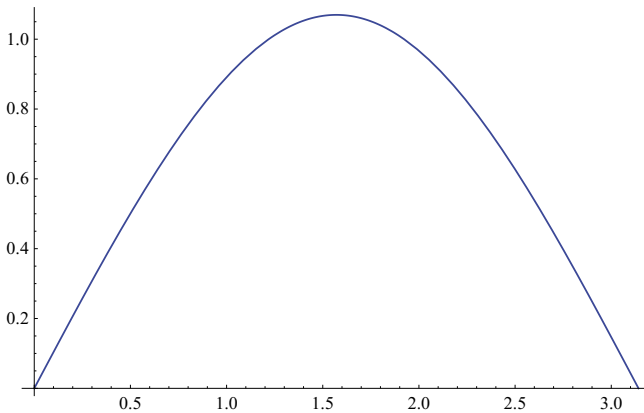


Fig. 4. Solution for $\gamma = 0.2$.

4 Handling point constraints

Handling point constraints on y requires its continuity. In the previous section sufficient conditions have been provided. Using the same reasoning as in the proof of Theorem 4 and different variants of ChJ theorem (see [11] chap. 5.11) one may obtain another conditions imposed on y^* .

In this section we assume that the solution of the system state Fredholm equation has a continuous solution.

Let us consider once more (3) with additional constraints as follows:

$$y(\xi) = \hat{y}_\xi \tag{20}$$

where \hat{y}_ξ is a required value. By expanding y into series we obtain

$$y(\xi) = \sum_{i=1}^{\infty} y_i v_i(\xi) = \hat{y}_\xi \tag{21}$$

after substituting y_i we get

$$\sum_{i=1}^{\infty} \frac{\lambda_i v_i(\xi)}{1 - \lambda_i} u_i = y_\xi$$

As a result we obtain linear constrain with respect to u_i .

Optimal control problem (3) with constraints (20) is equivalent, assuming $n \rightarrow \infty$, to a quadratic programming problem with linear constraints:

$$\begin{aligned} \min_{u_i} \sum_{i=1}^n \left(y_i^* - \frac{\lambda_i u_i}{1 - \lambda_i} \right)^2 + \gamma u_i^2 \tag{22} \\ \sum_{i=1}^n \frac{\lambda_i v_i(\xi)}{1 - \lambda_i} u_i = y_\xi \end{aligned}$$

To our previous example let's add additional requirement $y(0.3) = 0.35$, leaving other parameters unaltered.

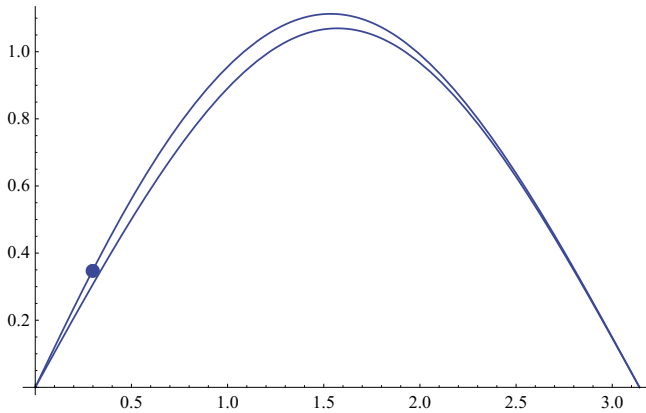


Fig. 5. Temperature with additional, point constrain.

In practice the temperature profile as in fig. 5, can be achieved by using inductive heating or laser heating. In many cases it can be cheaper to heat on

constant segments, so now we would consider step basis for control function as in fig. 6. The result is acceptable as shown in fig. 7.

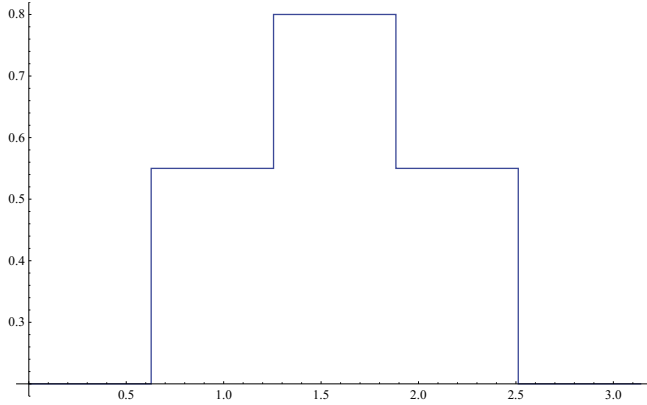


Fig. 6. Control is step base.

After numerically solving (22) we obtain solution fig. 7.

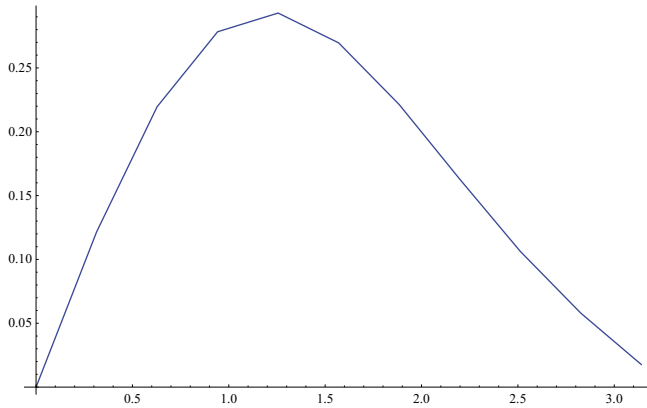


Fig. 7. Temperature profile for step control.

5 Concluding remarks

In the paper the method of solving optimal control problems for Fredholm integral equation of a second kind was described. Examples for constrained problems were provided for heat equation with and without point-wise constraints, which

are easy to handle in this approach. Further research would be desirable in order to extend the proposed approach for nonlinear equations.

References

1. Carleson L., On convergence and growth of partial sums of Fourier series. *Acta Mathematica*, Vol 116, No 1, pp-135-157, 1966
2. Li, Xungjing, and Jiongmin Yong. *Optimal control theory for infinite dimensional systems*. Springer Science & Business Media, 2012.
3. Rafajłowicz W. *Hybrid algorithms of optimal control for systems described by integro-algebraic equations*. *PhD. Thesis* University of Zielona Gora 2016 (*in polish* *Hybrydowe algorytmy optymalnego sterowania systemami cakowo-algebraicznymi.*)
4. Roubicek, T. *Optimal control of nonlinear Fredholm integral equations*. *Journal of optimization theory and applications* 97.3 (1998): 707-729.
5. Rolewicz S., *Analiza funkcjonalna i teoria sterowania*. Państwowe Wydawn. Naukowe, 1977.
6. Styczeń K., *Warunki optymalności aproksymatywnych procesów sterowania*. Wrocławskie Towarzystwo Naukowe, 2013.
7. Tricomi F.G. *Integral equations*. Dover Publications, Inc., 1985.
8. VR Vinokurov. *Optimal control of processes described by integral equations*. i. *SIAM Journal on Control*, 7(2):324–336, 1969.
9. VR Vinokurov. *Further comments on the paper optimal control of processes described by integral equations*. i by vr vinokurov. *SIAM Journal on Control*, 11(1):185–185, 1973.
10. Yousefi, S., and M. Razzaghi. Legendre wavelets method for the nonlinear Volterra-Fredholm integral equations. *Mathematics and Computers in Simulation* 70.1 (2005): 1-8.
11. Zygmund, Antoni. *Trigonometric series*. Vol. 1 and 2. Cambridge university press, 2002.

State vector estimation in descriptor electrical circuits using the shuffle algorithm

Kamil Borawski

Bialystok University of Technology,
Faculty of Electrical Engineering,
Wiejska 45D, 15-351 Białystok
kam.borawski@gmail.com

Abstract. Observers for descriptor electrical circuits by the use of the shuffle algorithm are proposed. Necessary and sufficient conditions for the existence of the observers are given. The effectiveness of the proposed method is demonstrated on a numerical example.

Keywords: Observer synthesis, descriptor, electrical circuit, shuffle algorithm.

1 Introduction

Descriptor (singular) linear systems have been investigated in [2-7, 10, 14, 17, 20, 22]. The eigenvalues and invariants assignment by state and input feedbacks have been addressed in [9, 10]. The computation of Kronecker's canonical form of a singular pencil has been analyzed in [23].

Descriptor observers for standard and fractional descriptor linear systems have been proposed in [8, 13, 15]. The minimum energy control of descriptor linear systems in has been analyzed in [11, 12, 18]. Stability of positive descriptor systems has been investigated in [24]. Comparison of three different methods for finding the solution for descriptor fractional discrete-time linear system has been presented in [21]. Comparison of two different methods of observer synthesis for descriptor discrete-time linear systems has been investigated in [1]. Fractional linear systems and electrical circuits have been analyzed in [19].

In this paper the observer synthesis for descriptor electrical circuits based on the shuffle algorithm is proposed.

The paper is organized as follows. In Section 2 some definitions and theorems concerning descriptor electrical circuits are given. The shuffle algorithm method is presented in Section 3. In section 4 the shuffle algorithm is applied to establish observers for descriptor electrical circuits. A numerical example is presented in Section 4. Concluding remarks are given in Section 5.

The following notation will be used: \mathfrak{R} - the set of real numbers, $\mathfrak{R}^{n \times m}$ - the set of $n \times m$ real matrices and $\mathfrak{R}^n = \mathfrak{R}^{n \times 1}$, I_n - the $n \times n$ identity matrix, \mathbf{C} - the field of complex numbers.

2 Preliminaries

Consider the linear electrical circuit

$$E \frac{dx(t)}{dt} = Ax(t) + Bu(t), \quad (1a)$$

$$y(t) = Cx(t), \quad (1b)$$

where $x(t) \in \mathfrak{R}^n$, $u(t) \in \mathfrak{R}^m$, $y(t) \in \mathfrak{R}^p$ are the state, input and output vectors and $E, A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times m}$, $C \in \mathfrak{R}^{p \times n}$. It is assumed that $\det E = 0$, $\text{rank } E = r$ and

$$\det[Es - A] \neq 0 \text{ for some } s \in \mathbb{C}. \quad (2)$$

It is well-known [19] that any linear electrical circuit composed of resistors, coils, capacitors and voltage (current) sources can be described in transient states by (1). Usually as state variables (components of the vector $x(t)$) currents in the coils and voltages on the capacitors are chosen.

Definition 1. The linear electrical circuit described by (1) and satisfying the assumption (2) is called a descriptor electrical circuit.

Theorem 1. [19] Linear electrical circuit is descriptor if it contains at least one mesh consisting of only ideal capacitors and voltage sources or at least one node with branches with coils.

In next section it will be shown that the electrical circuit (1) can be described in the following equivalent form

$$\frac{dx(t)}{dt} = \bar{A}x(t) + \bar{B}_0 u(t) + \bar{B}_1 \frac{du(t)}{dt} + \dots + \bar{B}_{q-1} \frac{d^{q-1}u(t)}{dt^{q-1}}, \quad (3a)$$

$$y(t) = Cx(t), \quad (3b)$$

where q is called the index of the pair (E, A) [10], $\bar{A} \in \mathfrak{R}^{n \times n}$, $\bar{B}_k \in \mathfrak{R}^{n \times m}$, $k = 0, 1, \dots, q-1$.

Theorem 2. The solution $x(t)$ of the equation (3a) for a given initial condition $x(0) = x_0$ and input $u(t)$ has the form

$$x(t) = e^{\bar{A}t} x_0 + \int_0^t e^{\bar{A}(t-\tau)} \left[\bar{B}_0 u(\tau) + \bar{B}_1 \frac{du(\tau)}{d\tau} + \dots + \bar{B}_{q-1} \frac{d^{q-1}u(\tau)}{d\tau^{(q-1)}} \right] d\tau. \quad (4)$$

Proof. The proof is similar to the one given in [10].

Definition 2. [10] The continuous-time linear system (3) is called asymptotically stable if $\lim_{t \rightarrow \infty} x(t) = 0$ for any finite $x_0 \in \mathfrak{R}^n$ and $u(t) = 0$.

Theorem 3. [10] The continuous-time linear system (3) is asymptotically stable if the zeros (the eigenvalues of the matrix \bar{A}) $\lambda_1, \dots, \lambda_n$ of the equation

$$\det[I_n \lambda - \bar{A}] = \lambda^n + \bar{a}_{n-1} \lambda^{n-1} + \dots + \bar{a}_1 \lambda + \bar{a}_0 = 0 \quad (5)$$

satisfy the condition

$$\operatorname{Re} \lambda_i < 0 \text{ for } i = 1, \dots, n, \quad (6)$$

where Re denotes the real part.

3 Shuffle algorithm method

Using the shuffle algorithm [10] we obtain the following. Performing elementary row operations on (1a) we obtain

$$\begin{bmatrix} E_1 \\ 0 \end{bmatrix} \frac{dx(t)}{dt} = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} x(t) + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(t), \quad (7)$$

where $E_1 \in \mathfrak{R}^{n_1 \times n}$ has full row rank ($\operatorname{rank} E_1 = n_1$) and $A_1 \in \mathfrak{R}^{n_1 \times n}$, $A_2 \in \mathfrak{R}^{(n-n_1) \times n}$, $B_1 \in \mathfrak{R}^{n_1 \times m}$, $B_2 \in \mathfrak{R}^{(n-n_1) \times m}$. Differentiation of the second equation of (7), i.e.

$$0 = A_2 x(t) + B_2 u(t) \quad (8)$$

with respect to time yields

$$A_2 \frac{dx(t)}{dt} = -B_2 \frac{du(t)}{dt}. \quad (9)$$

The equations (7) and (9) can be rewritten in the form

$$\begin{bmatrix} E_1 \\ A_2 \end{bmatrix} \frac{dx(t)}{dt} = \begin{bmatrix} A_1 \\ 0 \end{bmatrix} x(t) + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ -B_2 \end{bmatrix} \frac{du(t)}{dt}. \quad (10)$$

If the matrix

$$\begin{bmatrix} E_1^T & A_2^T \end{bmatrix}^T \quad (11)$$

is nonsingular then from (10) we obtain

$$\frac{dx(t)}{dt} = \begin{bmatrix} E_1 \\ A_2 \end{bmatrix}^{-1} \left(\begin{bmatrix} A_1 \\ 0 \end{bmatrix} x(t) + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ -B_2 \end{bmatrix} \frac{du(t)}{dt} \right). \quad (12)$$

If the matrix (11) is singular then performing elementary row operations on (10) we obtain

$$\begin{bmatrix} E_2 \\ 0 \end{bmatrix} \frac{dx(t)}{dt} = \begin{bmatrix} A_3 \\ A_4 \end{bmatrix} x(t) + \begin{bmatrix} B_3 \\ B_4 \end{bmatrix} u(t) + \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \frac{du(t)}{dt}, \quad (13)$$

where $E_2 \in \mathfrak{R}^{n_2 \times n}$ has full row rank ($\text{rank } E_2 = n_2$) and $\text{rank } E_1 \leq \text{rank } E_2$, $A_3 \in \mathfrak{R}^{n_2 \times n}$, $A_4 \in \mathfrak{R}^{(n-n_2) \times n}$, $B_3, C_1 \in \mathfrak{R}^{n_2 \times m}$, $B_4, C_2 \in \mathfrak{R}^{(n-n_2) \times m}$. Differentiation of

$$0 = A_4 x(t) + B_4 u(t) + C_2 \frac{du(t)}{dt} \quad (14)$$

with respect to time yields

$$A_4 \frac{dx(t)}{dt} = -B_4 \frac{du(t)}{dt} - C_2 \frac{d^2 u(t)}{dt^2}. \quad (15)$$

From (13) and (15) we have

$$\begin{bmatrix} E_2 \\ A_4 \end{bmatrix} \frac{dx(t)}{dt} = \begin{bmatrix} A_3 \\ 0 \end{bmatrix} x(t) + \begin{bmatrix} B_3 \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} C_1 \\ -B_4 \end{bmatrix} \frac{du(t)}{dt} + \begin{bmatrix} 0 \\ -C_2 \end{bmatrix} \frac{d^2 u(t)}{dt^2}. \quad (16)$$

If the matrix

$$\begin{bmatrix} E_2^T & A_4^T \end{bmatrix}^T \quad (17)$$

is nonsingular, then we can solve (16) in a similar way to (10). If the matrix (17) is singular, we repeat the procedure for (16) and for finite number of steps $q-1$ we obtain

$$x(t) = \begin{bmatrix} E_{q-1} \\ A_{q-1} \end{bmatrix}^{-1} \left(\begin{bmatrix} A_q \\ 0 \end{bmatrix} x(t) + \begin{bmatrix} B_q \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} C_q \\ 0 \end{bmatrix} \frac{du(t)}{dt} + \dots + \begin{bmatrix} 0 \\ -D_q \end{bmatrix} \frac{d^{q-1} u(t)}{dt^{q-1}} \right), \quad (18)$$

which is equivalent to the equation (3a).

4 Observer synthesis by the use of the shuffle algorithm

Definition 3. The continuous-time system

$$\frac{d\hat{x}(t)}{dt} = F\hat{x}(t) + G_0 u(t) + G_1 \frac{du(t)}{dt} + \dots + G_{q-1} \frac{d^{q-1} u(t)}{dt^{q-1}} + Hy(t) \quad (19)$$

where $\hat{x}(t) \in \mathfrak{R}^n$ is the estimate of $x(t)$ and $u(t) \in \mathfrak{R}^m$, $y(t) \in \mathfrak{R}^p$ are the same input and output vectors as in (3), $F \in \mathfrak{R}^{n \times n}$, $G_k \in \mathfrak{R}^{n \times m}$, $k=0,1,\dots,q-1$, $H \in \mathfrak{R}^{n \times p}$ is called a state observer for the system (3) if

$$\lim_{t \rightarrow \infty} [x(t) - \hat{x}(t)] = 0. \quad (20)$$

Let

$$e(t) = x(t) - \hat{x}(t). \quad (21)$$

From (21), (3) and (19) we have

$$\begin{aligned} \frac{de(t)}{dt} &= \frac{dx(t)}{dt} - \frac{d\hat{x}(t)}{dt} = \bar{A}x(t) + \bar{B}_0u(t) + \dots + \bar{B}_{q-1} \frac{d^{q-1}u(t)}{dt^{q-1}} \\ &\quad - (F\hat{x}(t) + G_0u(t) + \dots + G_{q-1} \frac{d^{q-1}u(t)}{dt^{q-1}} + HCx(t)) \\ &= (\bar{A} - HC)x(t) - F\hat{x}(t) + (\bar{B}_0 - G_0)u(t) + \dots + (\bar{B}_{q-1} - G_{q-1}) \frac{d^{q-1}u(t)}{dt^{q-1}}. \end{aligned} \quad (22)$$

If the matrices F , G_k , $k=0,1,\dots,q-1$, H are chosen so that

$$F = \bar{A} - HC, \quad G_k = \bar{B}_k, \quad k=0,1,\dots,q-1 \quad (23)$$

then from (22) we obtain

$$\frac{de(t)}{dt} = Fe(t). \quad (24)$$

If the system (24) (the matrix F) is asymptotically stable then $\lim_{t \rightarrow \infty} e(t) = 0$ and the observer (19) asymptotically reconstructs the state vector of the system (3).

Theorem 4. [10] The system (3) has a state observer (19) if and only if there exists a matrix H such that all eigenvalues of the matrix $F = \bar{A} - HC$ satisfies the condition (6).

It is well-known [10] that the finite eigenvalues of the matrix F can be arbitrary assigned if and only if the system (3) is observable, i.e. one of the following conditions is satisfied

$$1) \quad \text{rank} \begin{bmatrix} C \\ C\bar{A} \\ \vdots \\ C\bar{A}^{n-1} \end{bmatrix} = n, \quad (25)$$

$$2) \quad \text{rank} \begin{bmatrix} I_n s - \bar{A} \\ C \end{bmatrix} = n \text{ for } s \in \mathbf{C}. \quad (26)$$

5 Numerical example

Example 1. Consider the electrical circuit shown in Figure 1 with given resistances R_1, R_2, R_3 , inductances L_1, L_2, L_3 and source voltages e_1, e_2 .

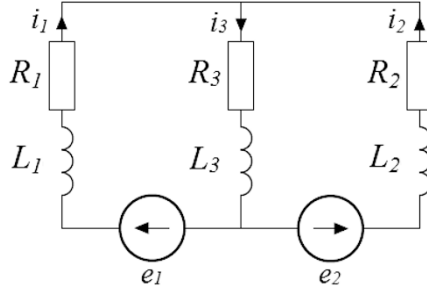


Fig. 1. Electrical circuit of Example 1

By Theorem 1 the electrical circuit is descriptor since it contains at least one node with branches with coils. Using the Kirchoff's laws we may write the equations

$$\begin{aligned} e_1 &= R_1 i_1 + L_1 \frac{di_1}{dt} + R_3 i_3 + L_3 \frac{di_3}{dt}, \\ e_2 &= R_2 i_2 + L_2 \frac{di_2}{dt} + R_3 i_3 + L_3 \frac{di_3}{dt}, \\ 0 &= i_1 + i_2 - i_3. \end{aligned} \quad (27)$$

As the output of the system we choose

$$y = i_1 + i_3. \quad (28)$$

The equations (27) and (28) can be written in the form (1)

$$E \frac{d}{dt} \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix} = A \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix} + B \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}, \quad (29a)$$

$$y = C \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix}. \quad (29b)$$

where

$$E = \begin{bmatrix} L_1 & 0 & L_3 \\ 0 & L_2 & L_3 \\ 0 & 0 & 0 \end{bmatrix}, A = \begin{bmatrix} -R_1 & 0 & -R_3 \\ 0 & -R_2 & -R_3 \\ 1 & 1 & -1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, C = [1 \ 0 \ 1]. \quad (30)$$

The pencil is regular since

$$\begin{aligned} \det[Es - A] &= \begin{vmatrix} sL_1 + R_1 & 0 & sL_3 + R_3 \\ 0 & sL_2 + R_2 & sL_3 + R_3 \\ -1 & -1 & 1 \end{vmatrix} \\ &= [L_1(L_2 + L_3) + L_2L_3]s^2 + [R_1(L_2 + L_3) + R_2(L_1 + L_3) + R_3(L_1 + L_2)]s \\ &\quad + R_1(R_2 + R_3) + R_2R_3 \end{aligned} \quad (31)$$

is nonzero for every positive values of the resistances and inductances.

For further analysis we assume the following values: $R_1 = R_2 = 2 \Omega$, $R_3 = 4 \Omega$, $L_1 = L_3 = 1 \text{ H}$, $L_2 = 2 \text{ H}$. Therefore, we have the descriptor continuous-time electrical circuit (1) with the matrices

$$E = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}, A = \begin{bmatrix} -2 & 0 & -4 \\ 0 & -2 & -4 \\ 1 & 1 & -1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, C = [1 \ 0 \ 1]. \quad (32)$$

From (32) we have

$$[E \ A \ B] = \begin{bmatrix} 1 & 0 & 1 & -2 & 0 & -4 & 1 & 0 \\ 0 & 2 & 1 & 0 & -2 & -4 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & -1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} E_1 & A_1 & B_1 \\ 0 & A_2 & B_2 \end{bmatrix}. \quad (33)$$

Performing the shuffle on (33) we obtain

$$\begin{bmatrix} E_1 & A_1 & B_1 & 0 \\ A_2 & 0 & 0 & -B_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & -2 & 0 & -4 & 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & -2 & -4 & 0 & 1 & 0 & 0 \\ 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (34)$$

The matrix

$$\begin{bmatrix} E_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & -1 \end{bmatrix} \quad (35)$$

is nonsingular and from (12) we have

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix} &= \begin{bmatrix} E_1 \\ A_2 \end{bmatrix}^{-1} \left(\begin{bmatrix} A_1 \\ 0 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix} + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -B_2 \end{bmatrix} \frac{d}{dt} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \right) \\ &= \begin{bmatrix} -1.2 & 0.4 & -1.6 \\ 0.4 & -0.8 & -0.8 \\ -0.8 & -0.4 & -2.4 \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix} + \begin{bmatrix} 0.6 & -0.2 \\ -0.2 & 0.4 \\ 0.4 & 0.2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \bar{A}x(t) + \bar{B}_0u(t), \end{aligned} \quad (36)$$

since $B_2 = 0$.

The output equation has the form (29b). The system (36), (29b) is observable since

$$\text{rank} \begin{bmatrix} C \\ \bar{C}\bar{A} \\ \bar{C}\bar{A}^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ -2 & 0 & -4 \\ 5.6 & 0.8 & 12.8 \end{bmatrix} = 3 = n. \quad (37)$$

Let $\hat{s}_1 = \hat{s}_2 = \hat{s}_3 = -10$ be the desired poles of the observer (19). Applying well-known pole assignment techniques [10] we obtain

$$H = [142.88 \quad 989.84 \quad -117.28]^T \quad (38)$$

and

$$\det[I_3s - \bar{A} + \hat{H}C] = \begin{vmatrix} s+144.08 & -0.4 & 144.48 \\ 989.44 & s+0.8 & 990.64 \\ -116.48 & 0.4 & s-114.88 \end{vmatrix} = s^3 + 30s^2 + 300s + 1000. \quad (39)$$

From (23) we have

$$F = \bar{A} - HC = \begin{bmatrix} -114.08 & 0.4 & -114.48 \\ -989.44 & -0.8 & -990.64 \\ 116.48 & -0.4 & 114.88 \end{bmatrix}, \quad G_0 = \bar{B}_0 = \begin{bmatrix} 0.6 & -0.2 \\ -0.2 & 0.4 \\ 0.4 & 0.2 \end{bmatrix} \quad (40)$$

and the matrix H has the form (38). The desired observer (19) for the system (36) has the form

$$\begin{aligned} \frac{d\hat{x}(t)}{dt} &= \frac{d}{dt} \begin{bmatrix} \hat{i}_1 \\ \hat{i}_2 \\ \hat{i}_3 \end{bmatrix} = \begin{bmatrix} -114.08 & 0.4 & -114.48 \\ -989.44 & -0.8 & -990.64 \\ 116.48 & -0.4 & 114.88 \end{bmatrix} \begin{bmatrix} \hat{i}_1 \\ \hat{i}_2 \\ \hat{i}_3 \end{bmatrix} + \begin{bmatrix} 0.6 & -0.2 \\ -0.2 & 0.4 \\ 0.4 & 0.2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \\ &+ \begin{bmatrix} 142.88 \\ 989.84 \\ -117.28 \end{bmatrix} y. \end{aligned} \quad (41)$$

Let us assume the initial conditions $i_1(0) = 1 \text{ A}$, $i_2(0) = 2 \text{ A}$, $i_3(0) = 3 \text{ A}$ for the electrical circuit (36) and zero initial conditions for the observer (41). The currents and their estimates for $e_1 = 5 \text{ V}$, $e_2 = 10 \text{ V}$ are presented in Figure 2.

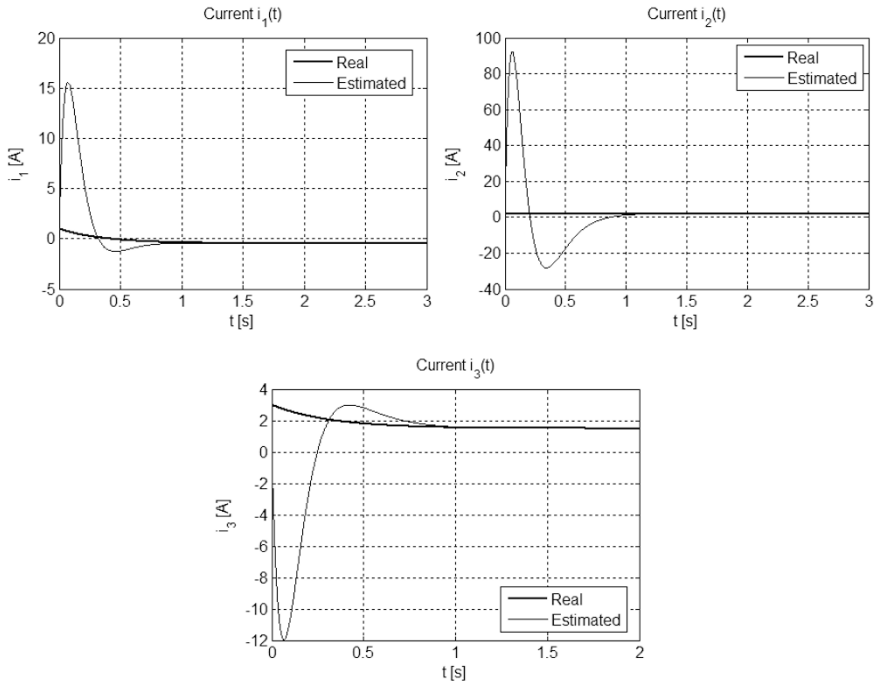


Fig. 2. Real and estimated currents of the electrical circuit

6 Concluding remarks

Observers for descriptor electrical circuits by the use of the shuffle algorithm have been proposed. Necessary and sufficient conditions for the existence of the observers have been given. The effectiveness of the proposed method has been demonstrated on a numerical example.

The considerations can be easily extended to fractional and positive descriptor electrical circuits.

Acknowledgment This work was supported by National Science Centre in Poland under work No. 2014/13/B/ST7/03467.

References

1. Borawski K.: Comparison of two different methods of observer synthesis for descriptor discrete-time linear systems. Proc. Conf. Automation 2017.

2. Campbell, S.L., Meyer, C.D., Rose, N.J.: Applications of the Drazin inverse to linear systems of differential equations with singular constant coefficients. *SIAMJ Appl. Math.* **31**(3), 411–425 (1976)
3. Dai, L.: *Singular Control Systems*, Lectures Notes in Control and Information Sciences. Springer, Berlin (1989)
4. Dodig, M., Stosic, M.: Singular systems state feedbacks problems. *Linear Algebra Appl.* **431**(8), 1267–1292 (2009)
5. Guang-Ren, D.: *Analysis and Design of Descriptor Linear Systems*. Springer, New York (2010)
6. Fahmy, M.M., O'Reilly, J.: Matrix pencil of closed-loop descriptor systems: Infinite-eigenvalues assignment. *International Journal of Control*, **49**(4), 1421-1431 (1989)
7. Kaczorek, T.: Descriptor fractional linear systems with regular pencils. *Asian Journal of Control*, **15**(4), 1051-1064 (2013)
8. Kaczorek, T.: Fractional descriptor observers for fractional descriptor continuous-time linear systems. *Archives of Control Sciences*, **24**(1), 27-37 (2014)
9. Kaczorek, T.: Infinite eigenvalue assignment by output-feedback for singular systems. *Int. J. Appl. Math. Comput. Sci.*, **14**(1), 19-23 (2004)
10. Kaczorek, T.: *Linear Control Systems vol. 1*. Research Studies Press, J. Wiley, New York (1992)
11. Kaczorek, T.: Minimum energy control of descriptor positive discrete-time linear systems. *COMPEL*, **33**(3), 976-988 (2014)
12. Kaczorek, T.: Minimum energy control of positive fractional descriptor continuous-time linear systems. *IET Control Theory and Applications*, **8**(4), 219-225 (2013)
13. Kaczorek, T.: Reduced-order fractional descriptor observers for a class of fractional descriptor continuous-time nonlinear systems. *Int. J. Appl. Math. Comput. Sci.*, **26**(2), 277-283 (2016)
14. Kaczorek, T.: Positive fractional continuous-time linear systems with singular pencils. *Bull. Pol. Acad. Sci. Techn.*, **60**(1), 9-12 (2012)
15. Kaczorek, T.: Reduced-order fractional descriptor observers for fractional descriptor continuous-time linear systems. *Bull. Pol. Acad. Sci. Techn.*, **62**(4), 889-895 (2014)
16. Kaczorek, T.: *Vectors and Matrices in Automation and Electrotechnics*. WNT, Warsaw (1998)
17. Kaczorek, T., Borawski, K.: Fractional descriptor continuous-time linear systems described by the Caputo-Fabrizio derivative. *Int. J. Appl. Math. Comput. Sci.*, **26**(3), 533-541 (2016)
18. Kaczorek, T., Borawski, K.: Minimum energy control of descriptor discrete-time linear systems by the use of Weierstrass-Kronecker decomposition. *Archives of Control Sciences*, **26**(2), 177-187 (2016)
19. Kaczorek, T., Rogowski, K.: *Fractional Linear Systems and Electrical Circuits*. Studies in Systems, Decision and Control, **13**, Springer (2015)
20. Kucera, V., Zagalak, P.: Fundamental theorem of state feedback for singular systems. *Automatica*, **24**(5), 653-658 (1988)
21. Sajewski, L.: Descriptor fractional discrete-time linear system and its solution – comparison of three different methods. *Challenges in Automation, Robotics and Measurement Techniques, Advances in Intelligent Systems and Computing*, **440**, 37-50 (2016)
22. Sajewski, L.: Solution of the state equation of descriptor fractional continuous-time linear systems with two different fractional orders. *Adv. Intell. Syst. Comput.*, **350**, 233-242 (2015)
23. Van Dooren, P.: The computation of Kronecker's canonical form of a singular pencil. *Linear Algebra and Its Applications*, **27**, 103-140 (1979)
24. Virnik, E.: Stability analysis of positive descriptor systems. *Linear Algebra and Its Applications*, **429**(10), 2640-2659 (2008)

Estimation of control improvement benefit with α -stable distribution

Paweł D. Domański & Piotr M. Marusak

Institute of Control and Computation Engineering, Warsaw University of
Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland
p.domanski@ia.pw.edu.pl, p.marusak@ia.pw.edu.pl

Abstract. The paper discusses the subject of estimation of potential financial benefits achievable with the rehabilitation (modification or tuning) of the control system. This issue appears almost in any process improvement initiative giving arguments for control upgrades. The subject exists in literature for several years with well established the same limit approach. The procedure is based on the assumption of Gaussian properties of the considered variable reflected in its histogram. Review of industrial data shows frequent situations when process variables are of different character featuring long tails. Such properties are well described by α -stable distributions. This paper presents extension of the method on such general probability density functions family. The analysis is illustrated with the simulation and industrial data examples.

1 Introduction

Industrial processes are often non-stationary time-varying systems with many cross-correlations and disturbances of unknown character and origin. Such conditions bring about challenges for the control system implementation, design and tuning. Goals are addressed by base control with single point or cascade PID loops. There are many reports showing that base control regulation fine tuning brings significant financial benefits. Further improvement may be reached with supervisory Advanced Process Control (APC) and/or Process Optimization.

Thus there is a need for methods to compare control improvement initiative cost against expected economic benefits. Such decisions are mostly based on the financial basis. There exist estimation techniques allowing calculation of the possible financial benefits resulting from the control system improvement. These profits are mostly associated with increased production efficiency, lower energy and material consumption, higher throughput or lower environmental fees. The cost element of the decision is simple, as it may be easily derived from past projects or obtained from control system vendor. The benefit part has to be evaluated specifically for each installation. The algorithm is based on the decrease of process variability leading to quantitative results. The method called *the same limit rule* assumes that the shape of the controlled variable histogram is Gaussian and normal standard deviation σ is used as the variability measure.

The rationale of this paper derives from observations gained in many commercial control improvement projects. It appears that behavior and shape of obtained trends and curves does not follow expected Gaussian properties. It is frequently fat tail. Lévy α -stable probability density function enables to achieve better fitting. To follow this observation, non-Gaussian method is proposed with modified Lévy α -stable *the same limit method*. Approach is more powerful as we may use more degrees of freedom, i.e. Probability Density Function's (PDF's) stability, skewness and scaling.

The algorithm is proposed and the analysis of the distribution factors' impact is presented. The evaluation is visualized with industrial examples.

1.1 Statistical measures

Normal distribution delivers many popular performance indicators. Mean value and standard deviation are commonly used. Standard deviation σ informs about signal variability. Higher value means larger variations and poorer control, while small values reflect opposite situation. Importance of these measures and their acceptance is unquestionable. But they are valid, when signal properties are Gaussian. It may be validated graphically through visual inspection of histogram. We may also use normality tests.

Review of control loops from various process industries [1] shows that only minority ($\approx 6\%$) has normal properties. Majority is characterized by fat tails well approximated with Lévy α -stable distribution ($> 60\%$). This may be explained by process complexity, correlations, varying delays and human impact. Thus, the use of stable PDF is justified.

An α -stable density function belongs to the family of stable distributions. It has four degrees of freedom (1) with four parameters (see Fig. 1); the characteristic function has in this case the following form:

$$S_{\alpha,\beta,\delta,\gamma}(x) = \begin{cases} \exp\left(-\gamma^\alpha|x|^\alpha\left\{1 - i\beta\text{sign}(x)\tan\left(-\frac{\pi\alpha}{2}\right)\right\} + i\delta x\right) & \text{for } \alpha \neq 1 \\ \exp\left(-\gamma|x|\left\{1 + i\frac{2}{\pi}\beta\text{sign}(x)\ln|x|\right\} + i\delta x\right) & \text{for } \alpha = 1 \end{cases}, \quad (1)$$

$0 < \alpha \leq 2$ is called *stability* index, $|\beta| \leq 1$ is a *skewness* factor, $\delta \in \mathbb{R}$ is *location* and $\gamma > 0$ is *scale* parameter. Stability parameter α is responsible for shape. Location δ keeps information about function position. Additionally we have two more shaping parameters. β informs about distribution skewness, while scaling γ keeps information about density function broadness. The family of α -stable distributions is a rich class, including following PDFs as subclasses:

- normal Gauss distributions $N(\mu, \sigma^2)$ given with $S(2, \beta, \frac{\sigma}{\sqrt{2}}, \mu)$,
- Cauchy PDF with scale γ and location δ given by $S(1, 0, \gamma, \delta)$,
- Lévy distribution (Inverse-Gaussian or Pearson V), with scale γ and location δ given by $S(\frac{1}{2}, 1, \gamma, \delta)$.

There are methods [2] dedicated to PDF to histogram fitting. α -stable case uses Koutrouvelis [3] regression approach.

2 Benefit estimation

The task to predict possible improvements associated with upgrade of a control system exists in literature for a long time [4]. From early days it was mostly associated with the implementation of the APC. There are three well established approaches called: *same limit*, *same percentage* and *final percentage* rules [5, 6]. All of them use assumption about Gaussian shapes of the controlled variable. Normal approach is followed by extensions with other PDFs.

2.1 Standard Gaussian approach

The *same limit* method is based on the evaluation of the normal distribution for selected variable keeping information about economic benefits and its modifications. Thus it assumes Gaussian properties of the process. Improvement potential is evaluated on the basis of the well-established algorithm presented below [7]:

1. Evaluate histogram of the selected variable or the performance index.
2. Fit normal distribution to the obtained histogram which is described by two parameters: mean value and standard deviation σ .
3. It is assumed that mean value (M_{improv} for the improved system and M_{now} for the original one) is kept within the same defined distance from potential upper (or lower limitation). The idea is to shift the mean value towards the respective constraint. For the confidence level of 95% it is equal to $a = 1.65$. Such a value is used in the calculations. The mean value for the improved operation is estimated. Standard deviation σ_0 relates to the original system and σ_1 to the improved one.

$$M_{improv} = M_{now} \cdot a \cdot (\sigma_0 - \sigma_1) \quad (2)$$

4. Finally percentage improvement is calculated on the basis of the following equation:

$$\Delta M = 100 \cdot \frac{M_{improv} - M_{now}}{M_{improv}} \quad (3)$$

Assume we have fitted distribution to the histogram with parameters (x_0, σ_0) . We also have to keep *the same limit* at point of $x^* = x_0 + k \cdot \sigma_0$, where k describes the same limit point position. We assume that better control improves variability by factor c , i.e. new $\sigma_1 = c \cdot \sigma_0$. Thus we may maintain the same limit with density function shifted by benefit (4).

$$M = k \cdot \sigma_0 - c \cdot \sigma_0 \sqrt{2 \left(\frac{k}{2} - \ln c \right)} \quad (4)$$

Let us assume that $k = 2.0$, $c = 0.75$, $u_0 = 1.0$, $\gamma_0 = 0.5$ (solid black line). Resulting limiting value is at point $x^* = 2.0$ with function value of 0.106 . We obtain new improved variability of $\gamma_1 = 0.38$ (blue dashed line). Thus we may shift distribution by benefit factor $M = 0.198$ towards the *same limit* (green dotted line). Fig. 2 presents graphical visualization of the above example.

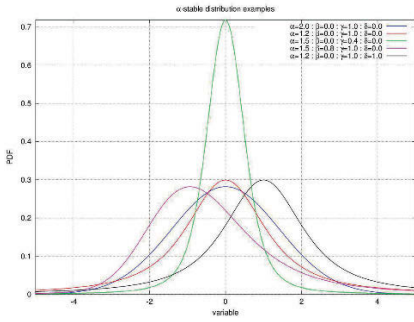


Fig. 1. Examples of Lévy PDFs

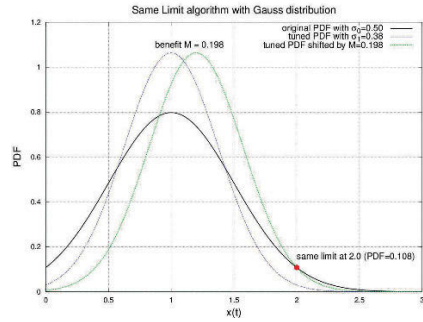


Fig. 2. Gauss *same limit* rule example

2.2 Algorithm for Lévy distribution

Control improvement benefit estimation using Lévy density function is not as straightforward as normal distribution, when there is single factor responsible for variability, i.e. σ standard deviation. Right now there are three parameters addressing variability. Scaling γ is responsible for distribution broadness, stability α for long tails and skewness β , which may also impact tail limit. We need to take into consideration combinations of these parameters in solving *the same limit* rule. It is infeasible analytically. Additionally each of these parameters addresses different aspects of the control loop tuning. Distribution broadness reflected by γ should be a measure of control quality. It plays the role of a robust Control Performance Assessment (CPA) measure [8].

Stability factor (α) may also impact control quality. It reflects persistence properties of the variable, i.e. control system. Skewness also informs about control quality. It is not always wrong. In some cases we may allow variations in the direction opposite to the constraint reducing dangerous ones in direction of the limitation (e.g. steam desuperheater control). Analyses of the α -stable factors impacting *the rule of the same limit* are evaluated through simulations.

As the function describing the PDF is complicated, a numerical approach was used to calculate the benefit obtained after control system tuning. The method relies on finding the zero of the difference of two PDF functions in the reference point, i.e. the function with "after tuning" parameters and the original one "before tuning" being shifted. The 'fsolve' Matlab function is used to obtain the result. The developed method is efficient and can be used with any PDF.

2.3 Simulation results

Simulation experiments were performed according to the predefined procedure, designed to verify assumed hypothesis. The results for the situation, when skewing factor remains unchanged during loop improvement ($\beta = 0.1$) is sketched in Fig. 3. Situation, when tuning causes symmetrization of the control error histogram, i.e. skewness factor after loop improvement becomes zero ($\beta = 0.0$) is presented in Fig. 4.

Diagrams below present respective shapes of the probability density functions change during the experiments. Selected plots show extreme cases only. Minor improvement (10%) in scale and situation of unchanged persistent behavior ($\alpha = 1.2 \rightarrow 1.2$) is sketched in Fig. 5, while Fig. 6 reflects shifting the system towards uncorrelated properties with $\alpha = 1.2 \rightarrow 2.0$. Figs. 7 ... 8 depict respective plots for large scale improvement of 80 %. Respective presentation of all the results in tabular form is presented in Tables 1 and 2.

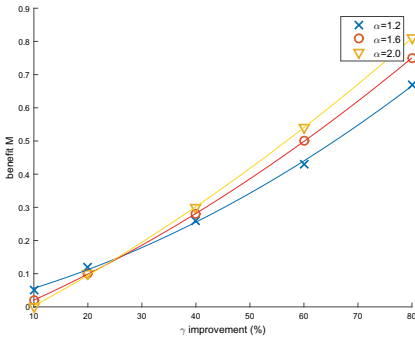


Fig. 3. Predicted improvement M versus PDF scale γ assuming unchanged skewness factor, i.e. $\beta = 0.1$ for different values of stability factor α .

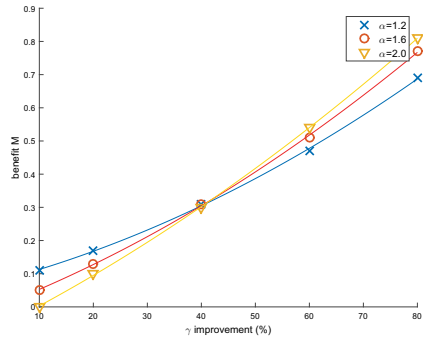


Fig. 4. Predicted improvement M versus PDF scale γ assuming symmetrization with skewness factor $\beta = 0.0$ for different values of stability factor α .

Table 1. The *same limit* rule results of M and δ shifts for symmetric scenario $\beta = 0.0$

α	$\gamma = 0.45$		$\gamma = 0.40$		$\gamma = 0.30$		$\gamma = 0.20$		$\gamma = 0.10$	
	M	δ	M	δ	M	δ	M	δ	M	δ
1.2	0.11	0.9378	0.17	0.9979	0.31	1.1353	0.47	1.3034	0.69	1.5239
1.6	0.05	0.8760	0.13	0.9566	0.31	1.1357	0.51	1.3441	0.77	1.5954
2.0	0.00	0.8345	0.10	0.9282	0.30	1.1341	0.54	1.3691	0.81	1.6433

Table 2. The *same limit* rule results of M and δ shifts for asymmetric scenario $\beta = 0.1$

α	$\gamma = 0.45$		$\gamma = 0.40$		$\gamma = 0.30$		$\gamma = 0.20$		$\gamma = 0.10$	
	M	δ	M	δ	M	δ	M	δ	M	δ
1.2	0.05	1.0396	0.12	1.0844	0.26	1.1925	0.43	1.3337	0.67	1.5313
1.6	0.02	0.8903	0.10	0.9674	0.28	1.1399	0.50	1.3427	0.75	1.5905
2.0	0.00	0.8345	0.10	0.9282	0.30	1.1341	0.54	1.3691	0.81	1.6433

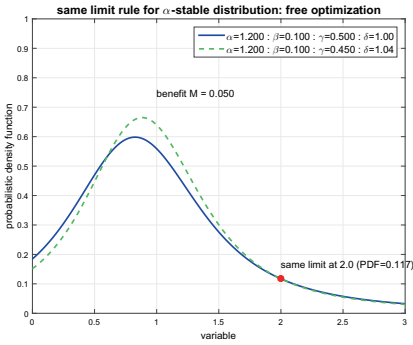


Fig. 5. 10% change in γ , $\alpha = 1.2 \rightarrow 1.2$

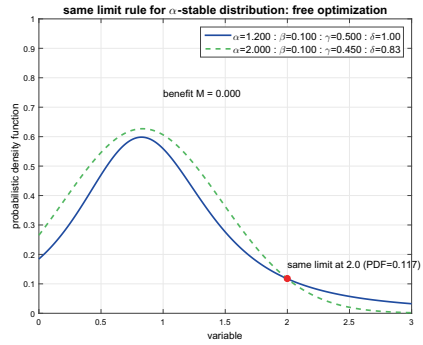


Fig. 6. 10% change in γ , $\alpha = 1.2 \rightarrow 2.0$

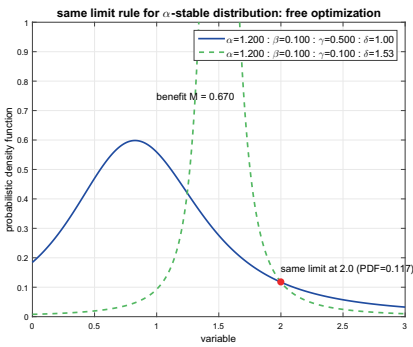


Fig. 7. 80% change in γ , $\alpha = 1.2 \rightarrow 1.2$

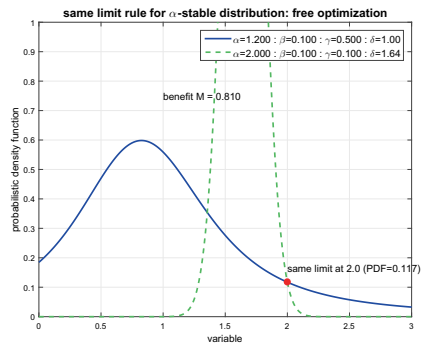


Fig. 8. 80% change in $\gamma, \alpha = 1.2 \rightarrow 2.0$

We notice that 10% change in γ , when $\alpha = 1.2 \rightarrow 2.0$ results in no improvement for both skewness scenarios. In all the other scenarios the shift in δ is accompanied with the M improvement. There are two other interesting observations. First of all the points form curves well approximated with second order polynomial. We also see that the curves cross. Larger benefit is obtained with persistent properties for small improvement in PDF broadness γ , while in case of large narrowing of the histogram, the bigger profits appear once stability factor shifts toward uncorrelated value of $\alpha = 2$.

3 Industrial validation

Industrial validation is performed on the anonymous control data from gas processing industry. The process has undertaken major tuning initiative. Thus there exists possibility to compare data *before* and *after* tuning. Four variables are selected, called Var1, Var2, Var3 and Var4. Plots of the variable histograms with fitted probability density function *before* the tuning are presented in the Figs. 9. . . 12. Next the histograms with fitted PDFs for variables *after* the tuning are presented in the consecutive Figs. 13. . . 16.

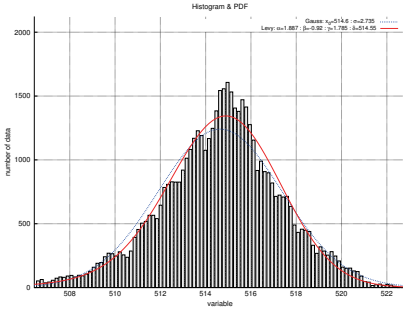


Fig. 9. Untuned Var1 histogram

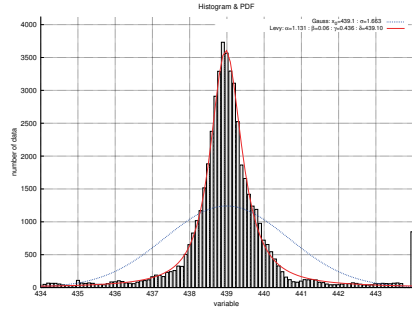


Fig. 10. Untuned Var2 histogram

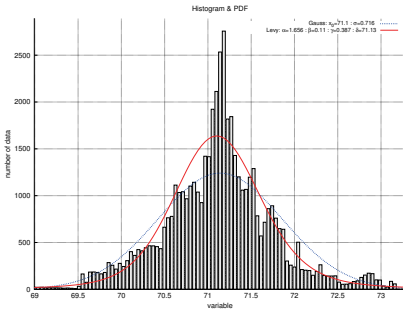


Fig. 11. Untuned Var3 histogram

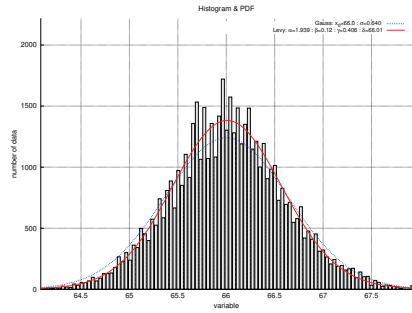


Fig. 12. Untuned Var4 histogram

Comparison of presented data consists of three elements. Table 3 shows mean square error representing fitting quality of each PDF to the variable histogram. As one can see variables Var1 and Var 4 are of Gaussian character, which is also visible in histogram graphs. In contrary two other variables hold fat tail properties. We are unable to find better normal density function fitting due to the histogram fat tails. The following two tables present comparison of the Gaus-

Table 3. PDF to histogram fitting mean square error

	Var1		Var2		Var3		Var4	
	<i>before</i>	<i>after</i>	<i>before</i>	<i>after</i>	<i>before</i>	<i>after</i>	<i>before</i>	<i>after</i>
Gauss	10.54	5.84	62.61	29.09	28.47	27.53	11.84	9.8
Lévy	8	2.65	10.59	10.93	21.61	15.27	9.63	9.39

sian (Table 4) and α -stable (Table 5) distribution factors for process variable histograms evaluated for loop *before* and *after* the tuning procedure.

First of all we may notice that the results for changes in normal mean value μ and position factor δ for all the variables are consistent and show the same

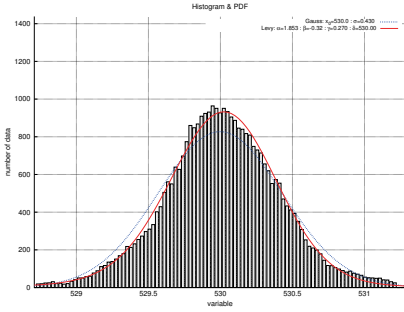


Fig. 13. Tuned Var1 histogram

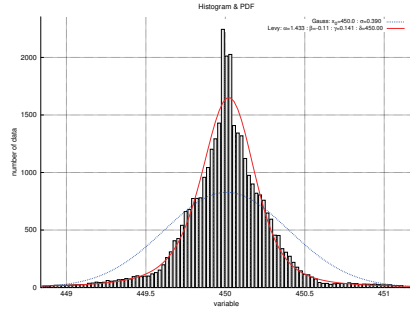


Fig. 14. Tuned Var2 histogram

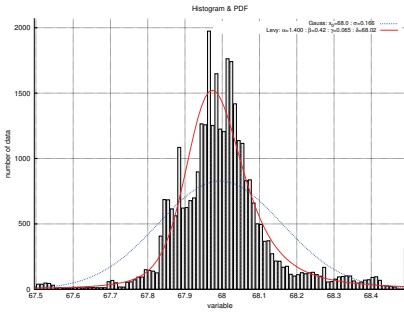


Fig. 15. Tuned Var3 histogram

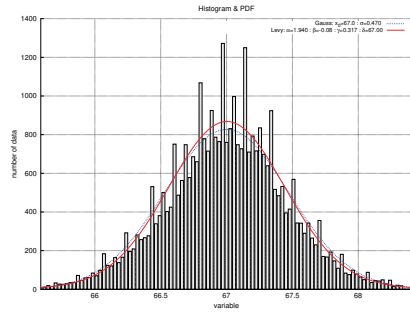


Fig. 16. Tuned Var4 histogram

Table 4. Comparison of Gaussian factors for loops *before* and *after* the tuning

	Var1		Var2		Var3		Var4	
	μ	σ	μ	σ	μ	σ	μ	σ
<i>before</i>	514.6	2.735	439.1	1.663	71.1	0.716	66.0	0.640
<i>after</i>	530.0	0.430	450.0	0.390	68.0	0.166	67.0	0.470
change %	3%	-84%	2%	-77%	-4%	-77%	2%	-27%

values. Normal variability factor of standard deviation σ shows for each of the loops significant improvement in dynamic properties. Additionally we may notice that for variables Var1 and Var4, which have the strongest Gaussian properties improvements in both factors, i.e. σ and γ are very similar to each other.

Variable Var2 features evident fat tail properties. In that case normal standard deviation shows bigger improvement in dynamics, as σ improves by 77%, while γ by 68%. Simultaneously we notice visible change in stability factor α . It improved from strongly persistent behavior towards expected to be optimal value of 2 reflecting uncorrelated time series.

Other effects are seen in the behavior of Var3 histograms. The same limit goal is achieved by two means. Improvement in the variability (density function broadness) is enhanced with asymmetric performance reflected in increased

Table 5. Comparison of α -stable PDF factors for loops *before* and *after* the tuning

	Var1				Var2			
	α	β	γ	δ	α	β	γ	δ
<i>before</i>	1.887	0.92	1.785	514.55	1.131	0.06	0.436	439.10
<i>after</i>	1.853	-0.32	0.270	530.00	1.433	-0.11	0.141	450.00
change %	-2%	-135%	-85%	3%	27%	-283%	-68%	2%

	Var3				Var4			
	α	β	γ	δ	α	β	γ	δ
<i>before</i>	1.656	0.11	0.387	71.13	1.939	0.12	0.406	66.01
<i>after</i>	1.400	0.42	0.065	68.02	1.940	-0.08	0.317	67.00
change %	-15%	282%	-83%	-4%	0%	-167%	-22%	1%

skewness β . This effect is only visible with fat tail distribution as symmetric normal PDF does not have such an ability.

4 Conclusions

The paper deals with the subject of the estimation of potential benefits that may be obtained from control system rehabilitation, i.e. structure upgrade, tuning or implementation of advanced control. There are well established methods supporting that task. They are based on the same limit rule idea and address reductions of process fluctuations resulting from control improvements. Variability is measured with the standard deviation of normal probability density function. Thus the method relies on the assumption that the properties of the analyzed variables are Gaussian.

Actually, there are frequent situations when assumption about variable normality does not hold, with evident fat tail properties. Following that observation *the same rule* method was extended towards fat-tail distributions. Respective relations were evaluated. It is also shown that comprehensive results are obtained with distributions having more degrees of freedom, like α -stable density function.

Results confirm complex properties of the applied fat tail distribution approach. More degrees of freedom in shaping of the histogram allow for more interpretations. The biggest impact was put on improvements, which directly influence control error histogram broadness. It is reflected in scaling factor γ . Earlier works confirm the hypothesis that this parameter is a good measure of the loop dynamic quality and its improvement may be achieved with the controller tuning. The other two interesting parameters of the α -stable distribution enable further interpretations.

Stability parameter α reflects persistent properties of the loop. The reasons for them are much broader and very often are associated with non-Gaussian noises, process complexity reflected in embedded correlations and human interventions into the loop operation. Thus it is not the result of direct controller

tuning. Improvement in it is rather associated with process modifications (technology). Finally asymmetric performance is the result of human interventions or process nonlinearities, often associated with installation equipment. Thus its changes are impacted with the other type of the installation improvements.

Concluding we see that application of the fat tail α -stable distribution into the process of potential benefit estimation out of control improvement delivers more degrees of freedom extending standard the same limit rule. It reflects controller tuning, with much more comprehensive perspective covering process technology and installation equipment.

The method requires further extended validation in industrial environment to detect and match proper reasons of shapes shifting behaviour. Investigation of the best practice values for associated parameters to predict control improvement benefits requires further attention as well, through thorough analysis of a large number of different cases, in particular.

References

1. Domański, P.D.: Non-Gaussian properties of the real industrial control error in SISO loops. In: Proceedings of the 19th International Conference on System Theory, Control and Computing. (2015)
2. Borak, S., Misiorek, A., Weron, R.: Models for heavy-tailed asset returns. In Cizek, P., Härdle, W., K., Weron, R., eds.: Statistical tools for finance and insurance. 2nd edn. Springer, New York (2011) 21–56
3. Koutrouvelis, I.A.: Regression-type estimation of the parameters of stable laws. *Journal of the American Statistical Association* **75**(372) (1980) 918–928
4. Tolfo, F.: A methodology to assess the economic returns of advanced control projects. In: American Control Conference 1983, IEEE (1983) 1141–1146
5. Bauer, M., Craig, I.K., Tolsma, E., de Beer, H.: A profit index for assessing the benefits of process control. *Industrial & Engineering Chemistry Research* **46**(17) (2007) 5614–5623
6. Bauer, M., Craig, I.K.: Economic assessment of advanced process control - a survey and framework. *Journal of Process Control* **18**(1) (2008) 2–18
7. Ali, M.K.: Assessing economic benefits of advanced control. In: Process Control in the Chemical Industries, Chemical Engineering Department, King Saud University: Riyadh, Kingdom of Saudi Arabia (2002) 146–159
8. Domański, P.D.: Fractal measures in control performance assessment. In: Proceedings of IEEE International Conference on Methods and Models in Automation and Robotics, Miedzyzdroje, Poland (2016) 448–453

Packet buffering, dead time identification, and state prediction for control quality improvement in a networked control system

Andrzej Tutaj, Wojciech Grega

AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical
Engineering

Department of Automatics and Biomedical Engineering
al. A. Mickiewicza 30, 30-059 Krakow*
tutaj@agh.edu.pl, wgr@agh.edu.pl

Abstract. Time-varying communication delays, data packet dropping, and other network-induced phenomena are inherent in networked control systems. Their presence can deteriorate noticeably control quality and narrow stability margins forcing designers to adopt conservative controller tuning. Control quality drop can be to some extent remedied by employment of network packet buffering or queuing, finite horizon state estimation, and continual time delay identification methods applied to networked control loops. The paper presents a modular structure for a networked control system where the named techniques are deployed in a network node located on the actuator site. A case study for a DC motor servomechanism and a periodic trajectory tracking problem is given. Simulation results are provided.

Keywords: networked control systems, distributed control systems, network induced time-varying communication delay, packet dropout, packet buffering, packet queueing, state prediction, state reconstruction, state estimation, Luenberger state observer, least mean squares method, delay time identification, linear-quadratic controller, DC motor servomechanism, tracking control, servo control

1 Introduction

Networked control systems (NCS) are more and more popular in almost all fields of automatic control including factory floor automation, building automation systems, road vehicles, and home appliances. A fieldbus or a communication network interconnects sensors, controllers, and actuators in NCS systems and helps to reduce costs involved by an excessive cabling often present in a traditional solution. On the other hand, presence

* This work was supported by AGH University of Science and Technology.

of a network introduces several new phenomena into the control loop. Depending on the underlying network type, they may include: random time-varying communication delays, packet dropout resulting in data unavailability, packet duplication or packet order reversal, nonsimultaneous delivery of packets originated from separate nodes, and many others. They can cause control quality deterioration and compromise stability conditions. To maintain acceptable level of robustness, controller designers have to resort to conservative tunings resulting in a further performance drop.

Many methods have been proposed in the literature to overcome or mitigate unfavourable phenomena incurred by network presence [4]. Some of them focus on control algorithms and try to adopt them to network-induced phenomena. That approach is called *control over network* [10]. Other adjust network protocols and algorithm in order to make the network more suitable for automation purposes. This attempt is referred to as *control of network*. There are also combined efforts described as *co-design* strategies [7,6].

Variance of the network-induced time delay can be reduced by a buffer or a queue implemented in a sensor, controller or actuator network node [1,14]. The buffer can also reduce packet dropping effects. State reconstruction and prediction techniques are widely employed to compensate for communication delays [13]. Smith predictor based solutions are often selected [1,15] to address network dead time problem. Controller parameters may also be adjusted according to gain scheduling scheme [5]. An interesting approach is presented in [17] where state observers implemented on sensor nodes in a MIMO system and connected to a shared network are used do control transmission rate and to reduce network traffic.

An important research area is stability analysis of distributed control systems [18]. In case of slowly varying network characteristics robust control framework can be used. Otherwise, if network properties change rapidly, switched systems or Markov chains theory may be applied [16]. Stability can be ensured with Lyapunov function and linear matrix inequalities (LMI) approach [16,9]. An interesting stability analysis for a system with packet dropping network is given in [11].

The solution proposed in this papers combines several methods originating from control and networking theory to improve quality of control in a networked control system. They include: packet buffering, clock adjustment, continual delay identification, a state observer, a state predictor, incorporation of a plant model into the control algorithm, and set point prefiltering. Resulting synergetic, adaptive closed-loop solution is intended for networked control systems following the structure shown in

Fig. 1. There is a control plant \mathbf{P} coupled tightly with a sensor \mathbf{S} and an actuator \mathbf{A} . The actuator is driven by a device constituting a single network node \mathbf{Na} where a control algorithm is implemented. A separate network node \mathbf{Ns} acts as a device measuring signal y from the sensor. Measurement results are transmitted via a computer network or a fieldbus \mathbf{N} . Thus a control loop is closed over the network.

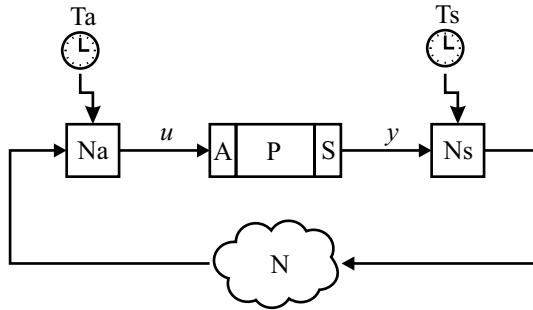


Fig. 1. Structure of networked control system: \mathbf{P} – plant, \mathbf{A} – actuator, \mathbf{S} – sensor, \mathbf{N} – fieldbus or computer network, \mathbf{Na} – actuator node, \mathbf{Ns} – sensor node, \mathbf{Ta} – actuator clock, \mathbf{Ts} – sensor clock

The paper is organised as follows. In section 2 a detailed structure of the system is presented and functions of all its components are explained. Section 3 presents a simulational study with the control system applied to a DC servomechanism. Final remarks are gathered in section 4 preceding a bibliography list.

2 Control System Structure and Operation Principle

The structure of the closed loop networked control system considered in the paper is shown in Fig. 2. It complies with the simplified block diagram depicted in Fig. 1 while presenting in details internals of sensor and actuator network nodes. A buffer \mathbf{B} collects data packets sent by a sensor node \mathbf{Ns} via network \mathbf{N} and passes them on to other blocks in an organised manner. It also adjusts timer \mathbf{Ta} governing the actuator node \mathbf{Na} operation. Communication delays are continually identified by \mathbf{I} block cooperating with a plant model \mathbf{M} . Identification results are used by estimator \mathbf{E} reconstructing an unknown plant state. The state estimate is fed to the controller \mathbf{C} of a 2DoF structure, cooperating with a set point prefilter \mathbf{F} forming reference and feedforward signals. The set point

is provided by generator **R**. Purposes of all individual modules and ways in which they cooperate are explained in following subsections.

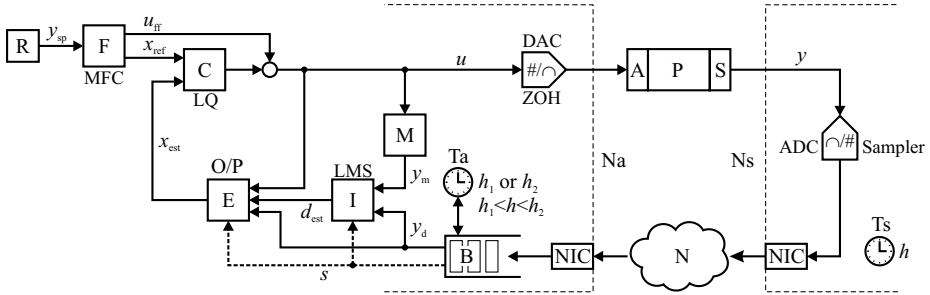


Fig. 2. Control system structure: **P** – plant, **A** – actuator, **S** – sensor, **N** – fieldbus or computer network, **Na** – actuator node, **Ns** – sensor node, **NIC** – network interface controller, **ADC** – analog to digital converter (sampler), **DAC** – digital to analog converter (**ZOH** – zero order hold), **Ta** – actuator node clock (interval timer), **Ts** – sensor node clock, **B** – packet buffer (queue), **M** – discrete-time plant model, **I** – time delay identification block, **E** – state estimator (one-step identity Luenberger observer, *d*-step state predictor), **C** – proportional state feedback controller, **F** – set point prefilter (reference state trajectory and dynamic command feedforward signal generator), **R** – set point generator.

2.1 Control Plant

A control plant **P** is coupled with actuator **A** and sensor **S**. Plant input signal $u(t) \in \mathbb{R}^p$ may be of arbitrary dimension $p > 0$ while plant output $y(t) \in \mathbb{R}$ is assumed to be scalar. A mathematical model is given as linear continuous time state space equations $\dot{x} = Ax(t) + Bu(t)$, $y(t) = Cx(t)$, $x(t) \in \mathbb{R}^n$ or a transfer function $G(s)$, which can be easily realised as the former model. As the control system works within a digital framework, equivalent discrete time model of the form $x_{k+1} = \Phi x_k + \Gamma u_k$, $y_k = Cx_k$ is employed, with state and input matrices $\Phi = e^{hA}$, $\Gamma = \int_0^h e^{tA} B dt$. Here $h > 0$ is a sampling period and k is an index corresponding to a discrete time instant $t_k = kh$. It is assumed that both continuous and discrete time models are observable, controllable and stable (a single integrating action is also allowed). Plant-model mismatch is considered and is assumed to be of a parametric type.

2.2 Sensor Node

A sensor node **Ns** samples plant output signal y using ADC converter with a constant sampling period h . It sends each y_k sample within a single packet using NIC controller via the network **N** the actuator node **Na**. These actions are triggered by the local clock **Ts**. All packets are supplemented with their serial numbers. Thus the receiver is able to remove duplicates and to rearrange packets should they arrive in an altered order.

2.3 Fieldbus or Computer Network

Data packets travel via network **N** possibly shared with other control loops and experience time-varying communication delay. It is assumed that the delay consists of two limited components: slowly varying, resulting mainly from changeable network load conditions and rapidly varying, caused for example by random, unscheduled medium access method. Sporadically the network may drop or duplicate a packet or change packets order. Time-varying delays and packet dropout make control task in NCS considerably harder than for conventional systems.

2.4 Packet Buffer and Actuator Node Clock

The main purpose of the buffer **B** is to reduce time delay variability making resultant delay almost constant or slowly varying [14]. That simplifies controller design and allows application of standard control theory methods. The buffer intercepts data packets incoming in random time instants and releases them in a more systematic manner. It accepts data packets from the network as soon as they come. It also sorts them according to their serial numbers and removes duplicates, if any. Data release process, on the other hand, is controlled by a clock **Ta** associated with the buffer. The clock should rather be referred to as an interval timer, since it does not keep a constant period. Instead the time interval is adjusted from step to step depending on data availability. If the buffer is empty, time interval is slightly extended compared to sensor sampling period h . The buffer does not wait for the missing data indefinitely, but rather releases a notice of data absence. If the buffer is not empty, the interval is slightly shortened. This process is governed by the following formula

$$\delta_k = h (1 + \alpha (r - s_k))$$

where δ_k is a time interval for the k -th step, s_k is a flag indicating availability ($s_k = 1$) or lack ($s_k = 0$) of data packet (signal s in Fig. 2),

$\alpha > 0$, $\alpha \approx 0$ is an interval length correction factor, and $0 < r < 1$ is a desired rate of data availability. Shortened and extended intervals are equal $h_1 = h(1 + \alpha(r - 1))$ and $h_2 = h(1 + \alpha r)$ respectively, $h_1 < h < h_2$. Adaptively adjusted step size δ_k allows the controller to keep track of the time-varying communication delay while reducing delay variation at the same time. Variation reduction is controlled mainly by the α factor, while r trades off packet dropout versus resultant filtered time delay. The timer **Ta** triggers not only the buffer action but also all other algorithms implemented on the actuator node. Thus, the buffer acts as a device synchronising sensor and actuator node clocks. Clocks frequencies are comparable (with an error dependent on α and r parameters) while phase shift follows a slowly varying component or an envelope (for $r \approx 1$) of network induced communication delays. Both parameters should be selected carefully based on communication time delay characteristics. A large value of α can make delay filtration ineffective while small may results in insufficient tracking capabilities. Parameter r value too close to either 0 or 1 makes the buffer respond sluggishly to increasing or decreasing time delay respectively. Figure 3 demonstrates an example of adaptive buffer operation.

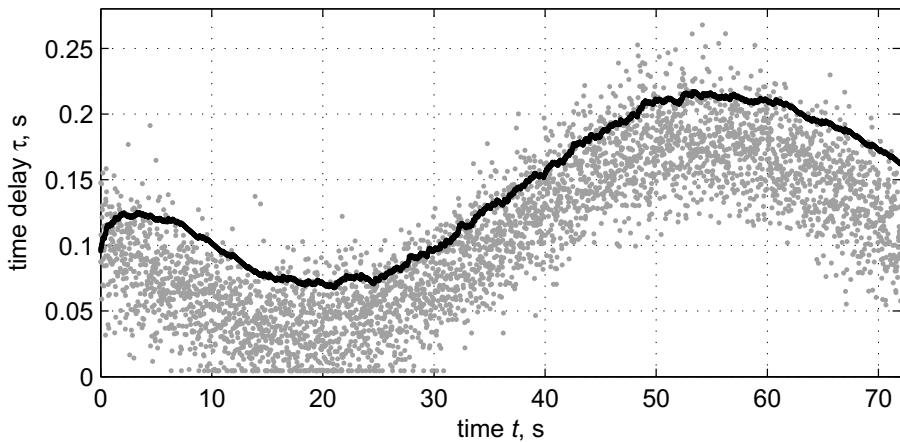


Fig. 3. Reduction of delay variation as a result of adaptive buffer introduction: time delays of packet entering (grey) and leaving (black) the buffer.

2.5 Time Delay Identification

To make the control algorithm able to compensate for the time delay it is necessary either to measure or estimate it. The first solution is possible only if both network nodes clocks are synchronised in frequency and phase and network packets are timestamped. That however involves troublesome

and resource-intensive implementation of time synchronisation protocols like NTP or PTP as well as application of stable oscillators. The solution proposed in the paper employs an alternative method with delay being continually identified with the **I** block implementing LMS FIR filter algorithm [2] and making use of a plant model **M**. Both delayed plant output measurement results y_d and present plant model output computation results y_m are fed to the **I** block. As both the plant **P** and its model **M** are fed with common control signal u (see Fig. 2), their outputs should only be time shifted, provided that initial conditions are identical and there are neither modelling errors nor external disturbances. The impulse response of a transport delay system has a form of a Dirac (continuous time) or Kronecker (discrete time) delta on a compact carrier. Hence, it can be approximated by a FIR filter. The **I** block estimates the time delay as a centre of gravity (CoG) of impulse response obtained with the LMS algorithm [15]. First differences with respect to time of discrete signals y_d and y_m rather than original signals themselves are fed to the LMS algorithm to avoid errors caused by the bias or offset (see Fig. 4). Thus it is possible to use the algorithm with plants exhibiting integrating characteristic or experiencing a constant load. An additional delay of $N/4$ steps has been introduced into y_d signal path where $N \cdot h$ is the FIR filter window width. It helps to avoid identification results lying near the starting point of the analysis window where computation result may be expected to be quite inaccurate. To make the identification possible, persistently exciting control signal u has to be secured [2]. In case of a periodic set point wave special care should be taken to make the LMS FIR filter window width smaller than the set point period to avoid spurious delays detection. On the other hand, N should be large enough to allow identification of the expected worst case time delay.

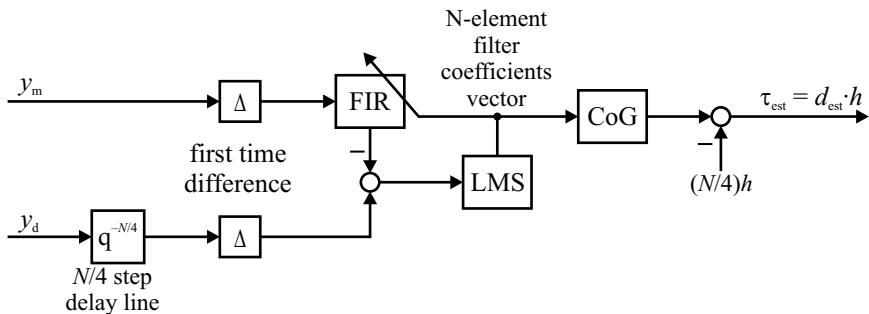


Fig. 4. Block diagram of time delay estimation algorithm.

2.6 State Estimator

A state estimator block **E** compensates for insufficient rank of the model output matrix C , missing data samples and the communication time delay [8].

If the rank of the output matrix C is lower than state space dimension, the state x cannot be calculated based on the instantaneous value of output signal y . It may be however reconstructed with an observer provided the system is observable or at least detectable. An identity Luenberger observer has been employed in the presented system. Modified version that introduces no additional delay has been chosen [3]: $\hat{x}_k = (I - LC)(\Phi \hat{x}_{k-1} + \Gamma u_{k-1}) + L y_k$. Observer gain matrix L has to be selected in such a way that the matrix $\Phi - LC\Phi$ appearing in an equation describing estimation error evolution is Schur stable.

If in a particular calculation step the y_k sample is missing a state predictor rather than the Luenberger observer has to be applied. The state predictor can be considered a “trivial” observer. The missing packet rate should be low enough not to compromise estimate convergence property.

To compensate partially for τ_k communication time delay d_k -step state predictor can be employed. Here $d_k = \lceil \tau_k/h \rceil$ where square brackets represent rounding operation.

Summing up, calculations performed by block **E** at each time step consist of exactly $\hat{d}_k + 1$ stages (see Fig. 5) where \hat{d}_k is the estimate of d_k . At the first stage $\hat{x}_{k-\hat{d}_k|k-\hat{d}_k}$ is computed from $\hat{x}_{k-\hat{d}_k-1|k-\hat{d}_k-1}$, $u_{k-\hat{d}_k-1}$ and $y_{k-\hat{d}_k}$, if available. Either identity Luenberger observer or state predictor algorithm is used depending on the availability of $y_{k-\hat{d}_k}$. All remaining stages make use of simple predictor formula $\hat{x}_{j|k-\hat{d}_k} = \Phi \hat{x}_{j-1|k-\hat{d}_k} + \Gamma u_{j-1}$ for $j \in \{k - \hat{d}_k + 1, \dots, k\}$.

It is necessary for block **E** to maintain memory for storing past values of control signal u and state estimates \hat{x} . At each time step some $\hat{x}_{i|j}$ values get overwritten by $\hat{x}_{j|j+1}$ calculated as explained earlier. Overwrites occur in an irregular manner as \hat{d}_k is not guaranteed to be constant and may change from step to step.

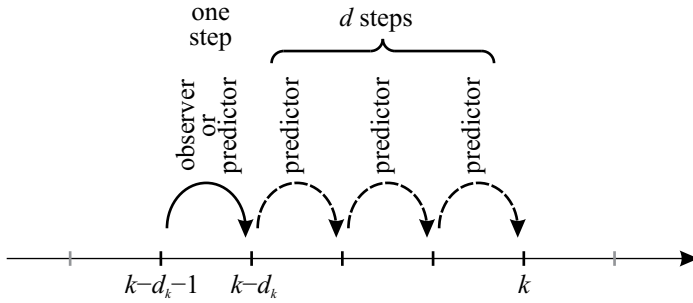


Fig. 5. A series of identity Luenberger observer and predictor stages constituting a single state estimation step.

2.7 Controller

A linear-quadratic algorithm has been selected for the controller **C** in the system. Proportional gain matrix K can be calculated based on the continuous plant model and an integral performance index of infinite horizon $J = \int_0^\infty (x(t)^T Q x(t) + u(t)^T R u(t)) dt$. Both model and performance index can be concurrently discretised with sampling period h [12] and the controller matrix can be computed from a solution of a discrete algebraic Riccati equation.

By itself the LQ controller is well suited for stabilization (regulatory) rather than tracking (servo) control. Hence, some modifications shown in Fig. 6 has been introduced resulting in 2DoF controller structure. Feed-forward control signal is added to the controller output ($u = u_{iq} + u_{ff}$) while its input is driven by the difference between reference trajectory and plant state estimate replacing an unknown plant state ($x_{err} = x_{ref} - x_{est}$).

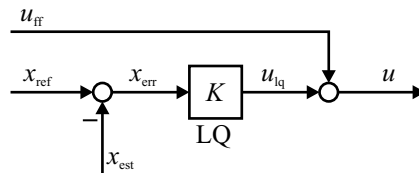


Fig. 6. 2DoF tracking LQ controller structure.

2.8 Set Point Prefilter

Both reference state trajectory x_{ref} and command feedforward signal u_{ff} are provided by a prefilter **F**. Its internal structure is shown in Fig. 7. It is inspired by the model following control (MFC) scheme and is build around the discrete time plant model **M**. Proportional state feedback controller gain K_f is selected to obtain a desired dynamical relationship between the set point y_{sp} and the reference signal y_{ref} using pole placement method. The relation is usually expressed in terms of a set of required eigenvalues or poles. Poles of discretised n -th order lag system of a given time constant T_f are often selected to that end. The scalar coefficient k_f in the set point path is chosen to obtain unit steady state gain between y_{sp} and y_{ref} . A model of actuator saturation is incorporated into the structure to ensure physical realizability of reference trajectories provided by the prefilter. Model parameters depend on the actuator type.

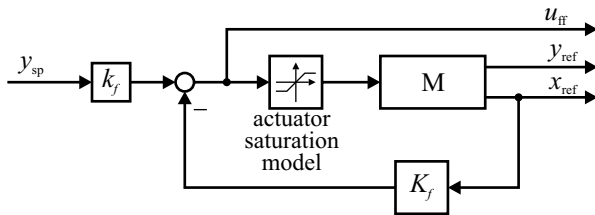


Fig. 7. Set point prefilter.

2.9 Set Point Generator

The set point source is represented by block **R** in Fig. 2. Since the proposed solution involves continual identification, the system must be persistently excited. If the original set point time series does not fulfil this requirement injection of a small amplitude pilot signal exhibiting that property should be considered. Square wave with a period adjusted to the plant dynamics is usually a good choice.

3 Simulational Case Study

The complete closed-loop control system behaviour has been modelled in MATLAB-Simulink environment. A DC servomotor has been selected as a control plant. It is modelled as a lag element of gain $K_m = 2.22$ and time

constant $T_m = 0.18$ s followed by an integrator. Motor angular position and velocity have been chosen as state variables ($x_1 = \varphi$, $x_2 = \omega$) with output signal equal to the position ($y = x_1$). The continuous time plant is represented by the following state space equations

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & -1/T_m \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ K_m/T_m \end{bmatrix} u(t), \quad y(t) = [1 \ 0] x(t)$$

Symmetric actuator saturation limits have been assumed with absolute value $U_m = 15$. To represent plant-model mismatch plant parameters has been deviated by 5% as follows: $K_p = 0.95 \cdot K_m$, $T_p = 1.05 \cdot T_m$. Symmetric square wave set point signal has been applied with amplitude $A_{sp} = 0.5$ and period $T_{sp} = 0.9$ s. Discrete-time prefilter has been tuned to mimic second order lag relationship between y_{sp} and y_{ref} with double time constant $T_f = T_m/5 = 36 \cdot 10^{-3}$ s. Resulting filter settings are: $K_f = [40.71 \ 2.902]$, $k_f = 40.71$ (see Fig. 7). Fixed sampling period of $h = 18 \cdot 10^{-3}$ s has been selected for the sensor node. It was also used for calculation of the discrete time equivalent of the plant model. Discrete-time LQ controller has been designed for infinite horizon integral performance index weight matrices $Q = \text{diag}(10, 0)$ and $R = 2 \cdot 10^{-5}$. The controller gain matrix is given by $K = [236.6 \ 5.866]$ (see Fig. 6). Identity Luenberger observer gain matrix has been calculated to make both eigenvalues of estimation error evolution equation matrix equal to $z_l = 0.9$. The result is $L = [0.1048 \ 0.0015]^T$. Window length of $N = 33$ has been chosen for LMS FIR filter as it is smaller than $T_{sp}/h = 50$. The additional delay in y_d signal path equals $h(N-1)/4 = 8$. A band-limited white noise with spectral power of $5 \cdot 10^{-3}$ and sample time equal to $h/10 = 1.8 \cdot 10^{-3}$ s has been injected into the control signal at the plant input to represent external stochastic disturbances acting on the system. Actuator clock period correction factor $\alpha = 0.075$ and desired packet delivery rate $r = 0.9$ has been chosen to govern buffer and actuator node clock operation. Network delays with slowly and fast varying components have been assumed. The fast component has been modelled using censored normal distribution with zero mean value and standard deviation $\sigma = 1.5 \cdot h = 27 \cdot 10^{-3}$ s. The slow component is represented by sum of constant value of $6 \cdot h$ and harmonic function with $4 \cdot h$ amplitude and 72 s period (see Fig. 3).

Several simulation tests with zero initial conditions have been conducted to verify effectiveness of the proposed solution and to compare it with a few alternative simplified structures. Figure 8 presents result of continual time delay identification for a selected simulation trial. It con-

firm's effectiveness of the employed identification method. Corresponding enlarged time series of plant input and output signals are shown in Fig. 9. They reveal relatively good tracking capabilities of the control system. Table 1 compares performance indices computed over a finite horizon of 75 s for several different variants of control systems. The first one represents the complete solution proposed in the article. Other correspond to simplified structures with one or more key elements removed. Considerable quality differences justify employment of the full solution rather than a simplified structure.

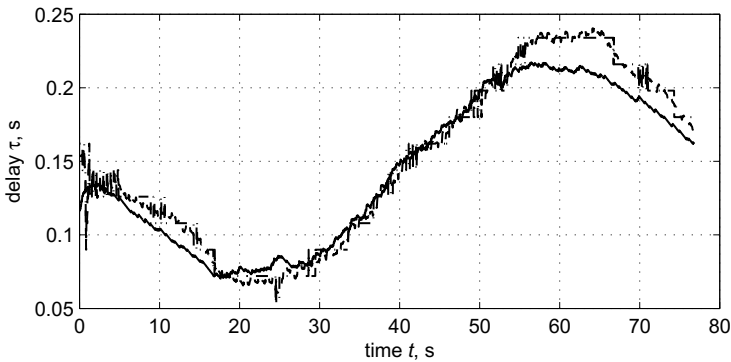
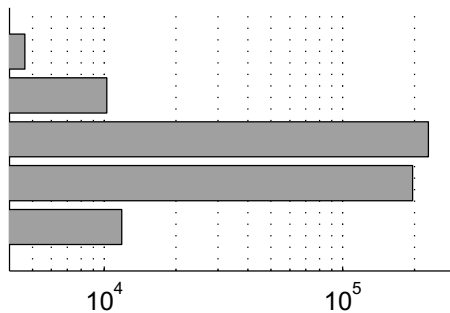


Fig. 8. Time delay identification results: actual time delay at the buffer output (solid), CoG calculation result (dash), CoG result rounded to integer value (dash-dot).

Table 1. Comparison of performance index values J for several control system variants.

Variant	J
complete solution with buffer, delay identification and state observer and predictor	4646
no delay identification, constant delay $\tau = 8 \cdot h$ assumed	10250
no delay identification, zero delay ($\tau = 0$) assumed	228700
no buffer, no delay identification, $\tau = 8 \cdot h$ assumed	196500
no buffer, no delay identification, $\tau = 0$ assumed	11840



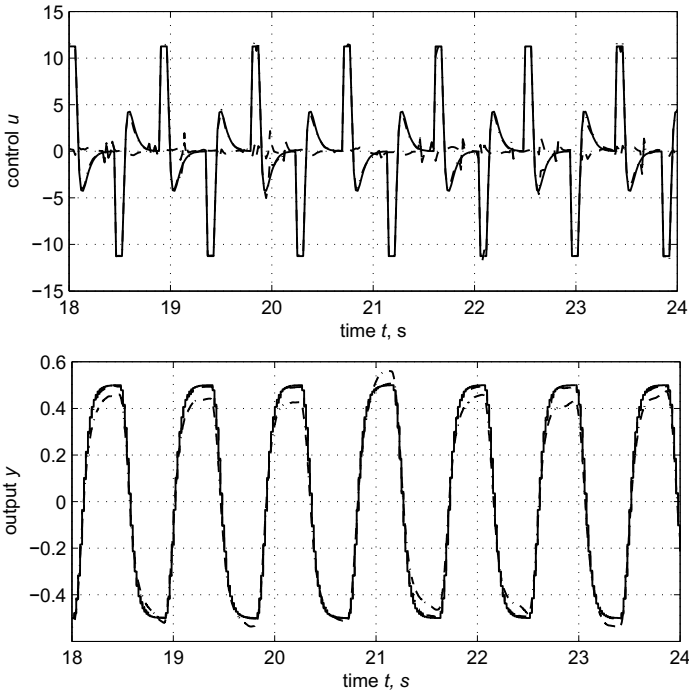


Fig. 9. Plant control and output signal trajectories. Control signal: feedforward component (solid), LQ controller component (dash), sum (dash-dot). Output signal: reference (solid), estimate (dash), actual (dash-dot).

4 Conclusions

In the solution presented in the paper packet buffering, state estimation, and delay identification are combined together to improve quality of control in a networked control system. The system is given a modular structure and relies on several elementary tasks that have been analysed and implement independently. That approach involves some simplifications but makes the problem more tractable.

Presence of the identification block makes the control system nonlinear while the network-induced delay causes it to be time-varying. Both these properties makes analytical stability analysis difficult. Simulation approach has been employed by the authors instead. As that method is by no means exhaustive, stability cannot be guaranteed. Hence in applications the presented algorithm should be supplemented with an emergency shut-down procedure activated as soon as an unstable behaviour is detected.

As an identification method is involved in the control algorithm, a persistently exciting stimulus must be provided and maintained. If a continual excitation is not acceptable an alternative possible solution is to activate the delay estimation algorithm intermittently and to use a burst test signal rather than an uninterrupted one.

The presented solution is formulated for single output (SISO and MISO) systems. However, it can relatively easily be extended to a MIMO or SIMO case, provided that all plant output signals are measured synchronously by a single sensor node and all samples corresponding to a given time instant are transmitted within a single network packet. All blocks involved in the control algorithms have to be accommodated to the multi-output framework with delay identification (**I**) and prefilter (**P**) blocks requiring arguably most attention. If individual output signal are measured by separate network nodes and transmitted asynchronously, a solution similar to that presented in [17] may be employed.

The assumption made in the paper that the controller is located on the actuator side is quite realistic one. Actuators often consume large power and are connected via individual lines directly to combined or coupled controller and power amplifier units located in a control cabinet or a control box. Sensor on the other hand usually exhibit low power demand and hence can use common power source and exchange data with the controller over a shared network or a fieldbus.

It should be also noted that the assumption on delay characteristics, i.e. superposition of slowly varying component and normal noise given in section 2.3, is important. If not fulfilled, the effectiveness of the controller could be lower.

References

1. Acrealã, A.M., Comnac, V., Boldișor, C.: Networked control systems: Network delay compensation with play-back buffers. In: 10th International Symposium on Signals, Circuits and Systems (ISSCS) (2011), 30 June-1 July 2011
2. Åström, K.J., Wittenmark, B.: Adaptive control. Dover Publications, Mineola, New York, second edn. (2008)
3. Åström, K.J., Wittenmark, B.: Computer-controlled systems. Theory and design. Dover Publications, Mineola, New York, third edn. (2011)
4. Bemporad, A., Heemels, M., Johansson, M. (eds.): Networked Control Systems. Lecture Notes in Control and Information Sciences, Springer (2010)
5. Chow, M.Y., Tipsuwan, Y.: Gain adaptation of networked DC motor controllers based on QoS variations. IEEE Transactions on Industrial Electronics 50, 936–943 (October 2003)

6. Dai, S.L., Lin, H., Ge, S.S.: Scheduling-and-control codesign for a collection of networked control systems with uncertain delays. *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY* 18(1), 66–78 (January 2010)
7. Daniel Simon, Ye-Qiong Song, C.A.: Co-design Approaches to Dependable Networked Control Systems. Wiley-ISTE (January 2010)
8. Grega, W., Tutaj, A.: Network traffic reduction by sample grouping for distributed control systems. In: *NeCST – 3rd international workshop on Networked control systems tolerant to faults*. pp. 1–8. Réseau de Formation des Jeunes Chercheuses et Chercheurs en Automatique et Productique, Nancy, France (June 2007)
9. Grega, W., Tutaj, A.: Improving quality of control in networked control systems by co-design of sampling period and buffer size. In: *MMAR 2014 – 19th international conference on Methods and Models in Automation and Robotics*. pp. 828–833. Międzyzdroje (September 2014)
10. Gupta, R.A., Chow, M.Y.: Networked control system: Overview and research trends. *IEEE Transactions of Industrial Electronics* 57, 2527–2535 (2010)
11. Hadjicostis, C.N., Touri, R.: Feedback control utilizing packet dropping network links. In: *Proceedings of the 41st IEEE Conference on Decision and Control*. vol. 2, pp. 1205–1210. Urbana, IL, USA (December 2002)
12. Loan, C.F.V.: Computing integrals involving the matrix exponential. *IEEE Transactions of automatic control* AC-23(3), 395–404 (June 1978)
13. Montestruque, L.A., Antsaklis, P.J.: On the model-based control of networked systems. *Automatica* 39, 1837–1843 (2003)
14. Tutaj, A.: Packets buffering in network traffic in distributed control systems. In: *MMAR 2006 – 12th IEEE international conference on Methods and Models in Automation and Robotics*. p. 571–578. Międzyzdroje (2006)
15. Tutaj, A.: Adaptacyjny układ regulacji z predyktorem Smitha z możliwością zastosowania w systemach rozproszonych — An adaptive Smith controller suitable for usage in distributed control systems. *Automatyka – Automatics – półrocznik Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie* 12(2), 211–231 (2008)
16. Xiao, L., Hassibi, A., How, J.P.: Control with random communication delays via a discrete-time jump system approach. In: *Proceedings of the American Control Conference*. Chicago, Illinois (June 2000)
17. Yook, J.K., Tilbury, D.M., Soparkar, N.: Trading computation for bandwidth: reducing communication in distributed control systems using state estimators. *IEEE Transactions on Control Systems Technology* 10, 503 – 518 (2002)
18. Zhang, W., Branicky, M.S., Phillips, S.M.: Stability of networked control systems. *IEEE Control Systems Magazine* 21(1), 84–97 (2001)

Part III
Autonomous Vehicles

Control Systems in Semi and Fully Automated Cars

Paweł Skruch

AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics,
Computer Science and Biomedical Engineering
Department of Automatics and Biomedical Engineering
al. A. Mickiewicza 30/B1, 30-059 Krakow, Poland
pawel.skruch@agh.edu.pl

Abstract. The paper focuses on essential elements of the control systems that are currently used in semi-automated cars and those that will be used in fully automated cars named as autonomous cars. Autonomous car is a vehicle that is capable of sensing its environment and navigating without human input. The highest level of automated driving assumes that all activities done by the driver can be replaced by a suitable control system. In this context, the autonomous car can be regarded as a control system working in a closed-loop setting. The desired value for the defined control system is a vehicle state specified in the destination point where the car should be placed starting a ride from a given initial state. The car in automated driving mode affects other cars in traffic, elements of road infrastructure and other road users, such as pedestrians, cyclists, animals, etc. Set of sensors provides data on the environment surrounding the car in the area defined by their field of view. These data after initial pre-processing are used for detecting objects surrounding the vehicle. Data from different types of sensors are combined with each other in order to increase the confidence level of the objects detected in the vehicles neighbourhood. Having the list of detected objects a vehicle environmental model is created, which is used to analyse the current situation and then plan the trajectory of the vehicle to the destination state. Besides the vehicle environmental model an important role in fully automated cars plays the driver model by means of which can be determined the characteristics for the actuators in such a way that the dynamics of the vehicle is as close as possible to the situation in which the driver manually guides the vehicle. Models of vehicle kinematics and dynamics are essential to tune controllers that are responsible for determining the trajectory of the vehicle and its stability properties. It should be emphasised that due to the nature of the automotive industry a fully automated driving will not happen overnight, but as technology develops. Control systems already on the market are gradually developed and their functionalities will work towards taking overall control of the vehicle.

Keywords: autonomous car, automated driving, control system

1 Introduction and Motivation

Autonomous vehicle can be defined as a vehicle capable of sensing its environment and navigating without human input. There are other names functioning in the literature: *driverless vehicle*, *uncrewed vehicle*, *self-driving vehicle*, *robotic vehicle*, *automated vehicle*. The latter name seems to be the most appropriate as the vehicles currently driving on roads, both prototypes and series ones, can be considered as automated vehicles at different levels of automation. Therefore, autonomous vehicle should be defined as a fully automated vehicle. The problem of building an autonomous vehicle can be then formulated as the problem of adding *senses* and *brain* to the vehicle so it is aware of its driving environment to first assist, and next substitute the driver. *Senses* in the vehicle are sensors with perception algorithms, they are sometimes called intelligent sensors; *brain* in the autonomous vehicle has the form of a distributed embedded software system where independent microprocessor systems communicate together using different communication networks.

As it may be difficult to determine in what percentage the vehicle moves in an automatic manner, the so-called levels of automation have been introduced. The Society of Automotive Engineers (SAE) identifies six levels of driving automation [21] from *No Automation* (level 0) to *Full Automation* (level 5). *No Automation* means that all aspects of the dynamic driving task, that is, execution of steering, acceleration, deceleration, monitoring of driving environment and fallback actions are performed by the human being. *Full Automation* stands for the situation when all aspects of the dynamic driving tasks under all roadway and environmental conditions are managed by an automated driving system. Newest vehicles, that are currently offered in dealer shops, are on level 3 (*Conditional Automation*). It means that the automated driving system monitors the driving environment and can perform itself selected dynamic driving tasks however with the expectation that the human driver will respond appropriately to a request to intervene. The National Highway Traffic and Safety Administration (NHTSA) classifies the automated driving modes in 5 levels' scale [23], where the lowest level 0 means *No Automation* and the highest level 4 means *Full Self-Driving Automation*.

According to the report of the World Health Organisation (WHO) concerning safety on roads [25], in 2013 the number of deaths caused by road accidents amounted approximately to 1.25 million per year. It is estimated that the annual costs related to treatment of road accident related injuries in European Union (EU) exceed 180 billion Euro. 3 571 persons died in road accidents on Polish roads in 2012 [3]. However, the number of victims has been decreasing every year which is due to the safety standards applied to vehicle construction. The most frequent reasons of accidents are due to the drivers fault and they are connected to failure to adapt speed to the conditions on the road, failure to give priority of passage, improper driving through pedestrian crossings, improper overtaking, and failure to keep a safe distance between vehicles. Accidents caused by pedestrians are typically due to careless entrance on the road (running across)

in front of a driving vehicle (including from behind another vehicle), on red lights or jaywalking.

Thanks to the enhanced development of passive safety systems, such as safety belts, air bags, etc., further improvement will be possible but not sufficient. These systems mainly protect the drivers and passengers but this protection is not absolute. The passive safety systems allow to minimise the accident effects but can neither eliminate nor prevent them. The problem of pedestrians' and bike and motor bike riders' safety in road traffic remains unsolved. Further improvement of the safety is possible thanks to the development of active safety systems, semi and finally fully automated vehicles. These systems can support the driver regardless of the part of the day and weather conditions. For instance, the function of autonomous emergency braking will significantly reduce the number of accidents involving pedestrians, bike and motor bike riders. Systems warning the driver against falling asleep or swerving off the lane will allow for avoiding accidents which cause great people, vehicle and road infrastructure damage. Autonomous vehicles will finally ensure comprehensive safety. Autonomous drive especially in long distances will allow for reducing of the drivers tiredness. In the cities the systems will provide for a smoother traffic which can translate into quantifiable economic benefits both for the state budget and drivers.

One of the purpose of the road traffic policy in EU is to promote mobility which is efficient, safe and environmentally friendly. The European Commission (EC) decided [6] that the road infrastructure is the third pillar of the road traffic safety policy which should to a great extent contribute to achieving the Community's goal that is reduction of the number of accidents. The EU action programme for road safety for the years 2011–2020 [8] sets a goal which assumes reduction of victims in road accidents in Europe by half within the next ten years that is by 2020. The programme includes ambitious proposals concerning increasing the vehicle safety, improvement of infrastructure and changing the road users behaviour. Autonomous vehicles which allow for significant reduction of human errors caused by for instance tiredness are very much within the goals.

EU directives [4, 5, 7] provide guidelines concerning fuel economy for new vehicles, both for passenger and truck vehicles, including guidelines concerning emissions of: nitric oxide (NO_x), hydrocarbon (NC), carbon monoxide (CO), carbon dioxide (CO₂) and particulate matter (PM). Application of the *eco-drive* rule will allow for fuel savings from 5 to 25%. The level of savings in every case depends on the drivers engagement and focus which is another source of tiredness. Automated driving in a natural manner allows for application of the *eco-drive* rule and savings in fuel consumption by approximately 15%.

The safety issue is also motivated by consumer's needs. Car manufacturers try to obtain best score results in the Euro NCAP (European New Car Assessment Programme) tests. Assessment criteria in these tests concern mainly the safety of passengers and other road users. It is worth mentioning that in a few years it will be possible to receive the highest score in those tests only if cars are equipped with active safety systems. So there is essentially no turning back from the introduction of active safety systems on the market.

The first systems in the automotive history containing automated driving features were simple adaptive cruise control and lane departure warning systems that were put into production around 2000. Currently, some serially produced cars from upper market segments contain features that enable automatic control over the vehicle in certain conditions. The driver can turn off the automated function at any moment of time and take back full control over the vehicle. According to automotive industry experts, cars that allow fully automated driving will appear on public roads around 2025. According to analysts from McKinsey Global Institute [16] autonomous cars are at the sixth position in the ranking of disruptive technologies that will change the world in the next 10 years. The same report features a calculation assessing the impact of this technology on the global economy in 2025 at the level of 0.2 – 1.9 trillion US dollars. The basis of those calculations is the vision of road traffic without any accidents that would allow to reduce road accidents in the interim period and eliminate them completely in the future. It is estimated that autonomous cars will entirely eliminate fatal accidents. This is strictly connected to the fact approximately 90% of accidents are because of the driver's fault [24]. The vision of road traffic without collisions and accidents is thus the main motivator and driving force behind actions taken by the largest automotive companies and corporations from other sectors.

2 Automated Car as a Control System

Automated car can be considered as a control system working in a closed-loop setting (Fig. 1). The endpoint for this control system is specified by the condition of the car in the destination where the car should be starting a ride from a given initial state. The car during driving influences other cars in traffic, elements of road infrastructure and other road users, that is, pedestrians, cyclists, animals, etc. Sensors provide data on the vehicle's surrounding in the area specified by their field of view. It is worth noticing that each car from the traffic block on Fig. 1 can work in the same way, so control systems for all cars located in a certain area are overlapping and influencing each other.

A vehicle software system is a distributed embedded software system where independent microprocessor systems, called Electronic Control Units (ECUs), communicate together using different communication networks. The list of wired protocols used in today's automotive industry includes: CAN (Controller Area Network) bus [13, 14], LIN (Local Interconnect Network) bus [15], MOST (Media Oriented System Transport) bus [17], FlexRay [10], and recently also Ethernet. Typical vehicle functionality is realized by several ECUs communicating with each other. The number of ECUs in vehicles, as well as their complexity, has been increasing from year to year. Nowadays, an average middle class vehicle is equipped with about 30 different co-operating microprocessor systems and the electronics comprises up to 40% of the total vehicle cost [2, 22]. A typical vehicle software system architecture is shown in Fig. 2.

An automotive ECU is a control system that processes continuously changing input signals and provides the appropriate output signals based on the inputs.

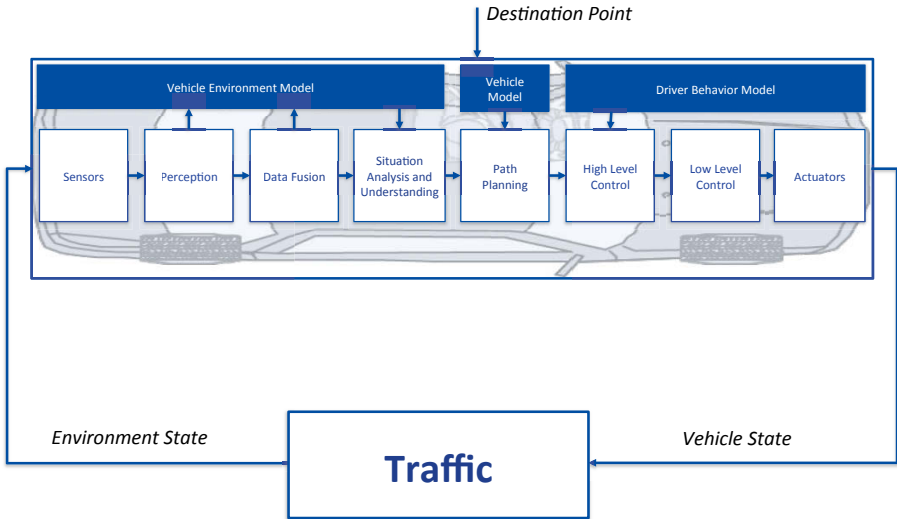


Fig. 1. Block diagram of a control system for automated cars

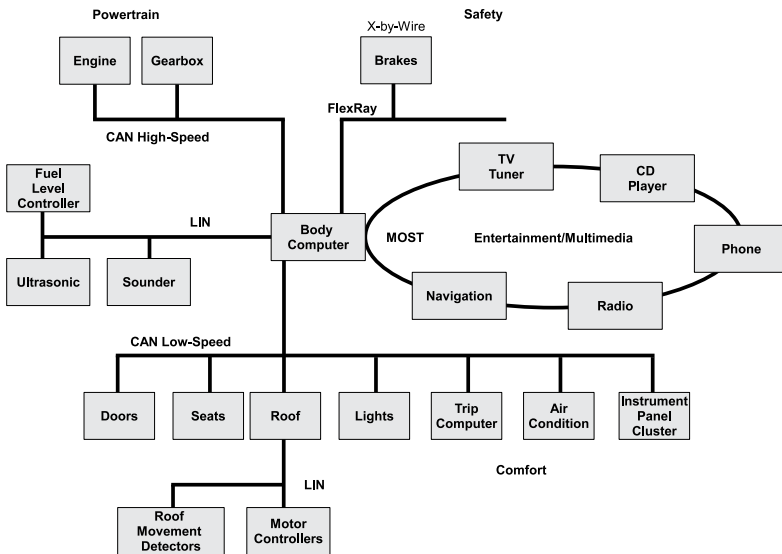


Fig. 2. Typical vehicle software system architecture

ECU behaviour can be categorised into three different types [9, 11]: discrete combinational (logical operation defined) behaviour, discrete sequential (state defined) behaviour, and continuous behaviour. Typically combinational behaviour

is modelled using Boolean algebra (in practice represented by predicates, logic gates, or graphs such as cause-effect graphs [1, 18]). State behaviour is modelled using graphs (usually state charts [12, 1, 9]). Continuous behaviour is modelled using differential equations (in practice represented by graphs of different Laplace transfer functions [19]). Automotive ECU behaviour is a combination of some or all of the above three basic behaviours.

Sensors are essential elements of automated driving. They are being used in perception algorithms to monitor and interpret the vehicle's surrounding. It looks that automated driving features will be based on the following set of sensors: cameras, radars, lidars, GPS (Global Positioning System), V2V (Vehicle-to-Vehicle) and V2I (Vehicle-to-Infrastructure) systems (Fig. 3). Camera takes



Fig. 3. Essential sensors for automated driving features: camera (left), radar (middle), lidar (right)

images of the road that are further interpreted in real-time by a computer program in order to distinguish and classify objects. The camera-based systems that are currently used in the automotive allow to distinguish and classify a variety of objects, such as: cars, trucks, pedestrians, small and large animals, road lanes, traffic lights, all types of traffic signs, barriers, and many others (Fig. 4). Radars provide a good measurement of the position and velocity of objects, both static and dynamic ones. Radar sends radio waves that are bounced from the objects and then received for interpretation (Fig. 5). Currently used radar systems are equipped with intelligent algorithms that besides high-quality range and velocity information are able also to classify some objects such as cars, pedestrians, barriers. Lidar works on the same principle as radar but instead of microwaves it uses light pulses that are sent out, reflected from the objects and then received for interpretation. Due to its high resolution lidar can be used to measure both the position and velocity of an object as well as classify several classes of objects. GPS receivers with detailed 3D maps are important to localise the vehicle and for path planning tasks. V2X (V2V + V2I) or V2E (Vehicle-to-Everything) are communication systems between other vehicles and elements of road infrastructure with intention to enhance reliability of perception algorithms.

Data from different types of sensors are combined with each other in order to increase reliability of the detected objects from the vehicle's surrounding. Due

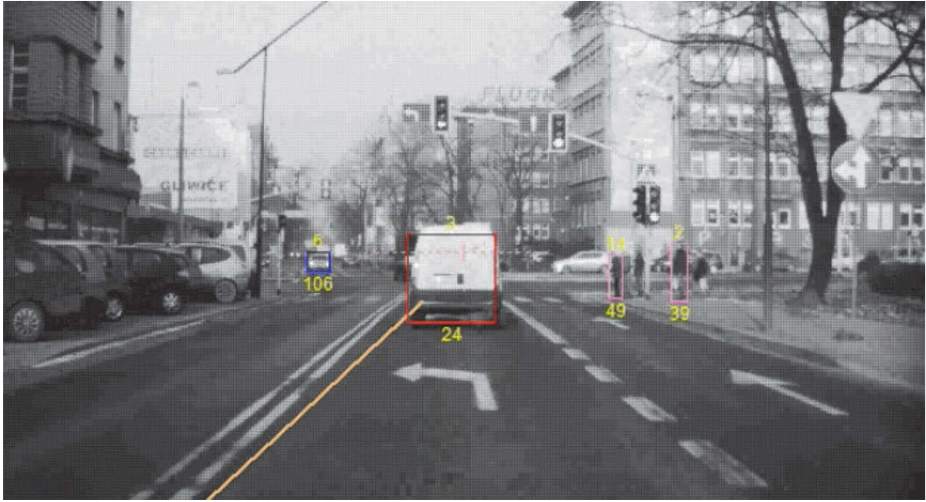


Fig. 4. Detection of objects using a vehicle camera

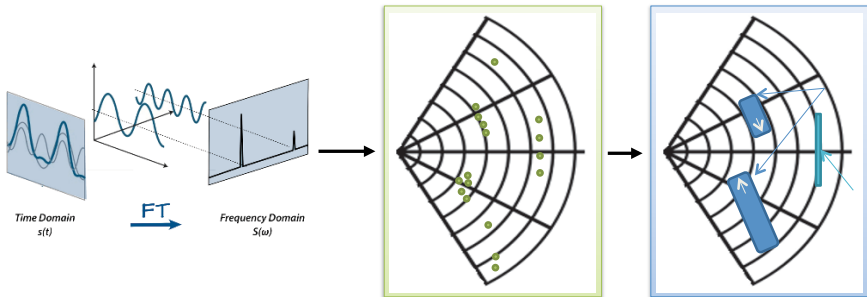


Fig. 5. Detection of objects using a vehicle short range radar

to the fact that the objects from the vehicle's surrounding can be detected with low confidence, the sensor data fusion allows to increase this confidence making the detected objects more reliable (Fig. 6). Using local objects provided by the sensing systems, objects identified by other cars and static objects from 3D maps a 3D model can be created for virtual representation and understanding of the vehicle's surrounding environment (Fig. 7). The model is dynamically changing over time during vehicle drive. This model shall be next used for implementation of the functionalities such as adaptive cruise control, autonomous emergency braking, lateral control, ad-hoc safety zones, control trajectories steering the vehicle from the starting point to the defined destination, etc. Besides the vehicle environmental model an important role in fully automated cars plays the driver model by means of which can be determined the characteristics for the actuators

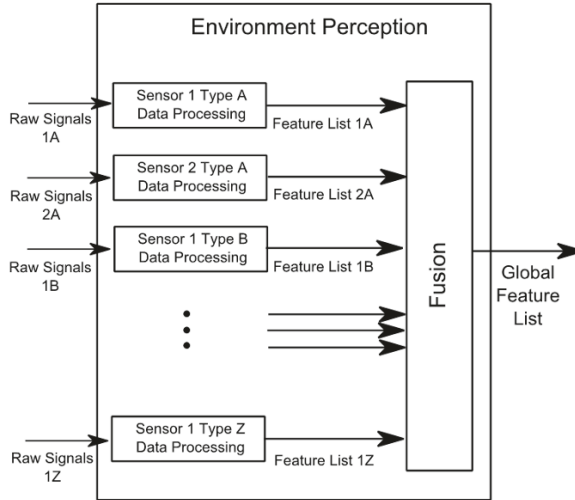


Fig. 6. General overview of sensor data fusion

in such a way that the dynamics of the vehicle is as close as possible to the situation in which the driver manually guides the vehicle. Models of vehicle kinematics and dynamics are essential to tune controllers that are responsible for determining the trajectory of the vehicle and its stability properties. Sets of objects detected by the sensors must be analysed always in the context of the road situation which takes place at every moment of time when the control decision should be made. Situational context on the road is changing rapidly and the same set of detected objects may cause other decisions relating to vehicle control. For instance, other decisions should be taken in a situation where a pedestrian intends to cross the street, and the other when he walks down the sidewalk next to the street. Other decisions are also taken in the event of changing traffic lights. Planning the trajectory of the vehicle is also extremely difficult and complex task. This is a multi-criteria optimisation problem that should consider such performance indices as travel distance, driving time, fuel consumption, etc. Path planning is updated usually every 50 ms (i.e., at a frequency of 20 Hz) and should take into account such situations as changing lanes, overtaking, turning to traffic, etc.

Due to the nature of the automotive industry a fully automated driving will not happen overnight, but as technology develops. Control systems already on the market are gradually developed and their functionalities will work towards taking overall control of the vehicle. The list presented below [20] may be considered as an overview of common active safety and advanced driver assistance features which are already available in serially produced cars and which will

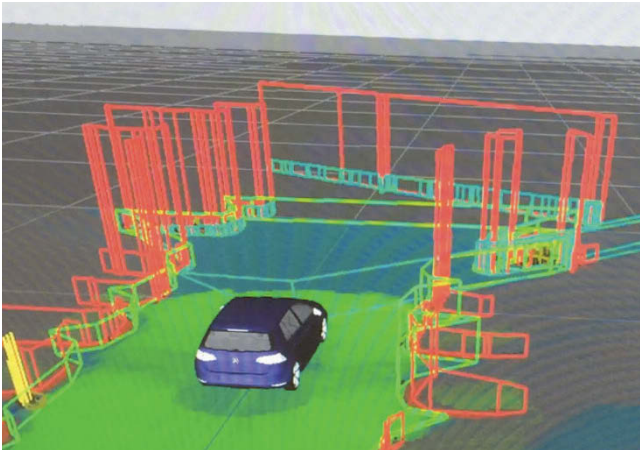


Fig. 7. Model of the vehicle's environment using radar detections only

be further developed and incorporated into semi and fully automated driving modes.

- Adaptive Cruise Control (ACC) — a cruise control of a new generation that increases the comfort of driving and enables speed regulation, not only on a set level but also in case of appearance of a vehicle that precedes the host. The speed and distance between vehicles is adjusted to safe values.
- Queue Assist (QA) – this may be an additional option to ACC that, unlike the ACC, works in the range of low speed together with full stop. QA is able to make the car start to move, keep the distance, and stop, which is a typical situation for a traffic jam.
- Traffic Jam Assist (TJA) — this is the QA feature with additional ability to control the steering wheel autonomously. Moreover, it can be considered as fully automated driving feature in restricted working conditions.
- Autonomous Emergency Braking (AEB) — this feature is responsible for braking if the system predicts unavoidable collision with an object in front of the host car. Since rear-end crashes without applying any braking action by the driver constitute significant percentage of accidents, this is a very promising feature in the case of rising safety level.
- Collision Warning (CW) — it alerts the driver about the possible danger of an accident. This exact signal implies the last chance of avoiding the collision. If the driver does not react, AEB is going to activate on its own.
- Collision Mitigation Support (CMS) — other name for the system which mitigates or avoids the collision; it consists of CW and AEB.
- Lane Departure Warning (LDW) — the purpose of this feature is to warn the driver in case a vehicle is crossing the lane marker while the turn indicator is not activated. The system interprets such situation as an effect of distraction or drowsiness, which may lead to an accident. After a few warnings the

system sends information that is displayed on the instrument panel cluster advising the driver to make a break.

- Lane Keep Aid (LKA) — it works in a similar way to the LDW. The system recognises lane markers on the road but tries to keep the vehicle on the proper lane by applying a force to the steering wheel.
- Curve Speed Warning (CSW) — this feature provides warning to the driver if the speed of a vehicle is not adapted to the road curvature.
- Road Friction Information (RFI) — information about road friction may come from wireless network or may be based on autonomous tests. This information adds to the increase of performance of ACC, AEB, CW and CSW.
- Intelligent Speed Adaptation (ISA) — it is based on traffic signs captured by camera or information provided by any external source. The system monitors current speed limit and reacts when the vehicle is moving too fast. A reaction device either alerts the driver or performs an automatic intervention concerning the speed of a vehicle.
- Adaptive Highbeam Assist (AHA) — adaptation of headlight beam distance depending on the position of vehicles on the road, in order not to dazzle other drivers.
- Blind Spot Information System (BLIS) — it is based on information from cameras located in the side mirrors. This feature warns the driver when the vehicle is in a blind spot. Warning is more intense if host car cuts in other vehicle path while changing the lane.

3 From Prototype to Series Production

Designing an embedded control system for automotive applications is a complex and error prone task. Embedded systems intended for automated driving applications are becoming increasingly sophisticated and their software content is growing rapidly. Although some prototypes with automated driving features are already available and work pretty well in a controlled environment, the way to serially produced cars seems to be a long journey. This is because every fault in such safety critical system may cost in worst case human life and can slow down or even stop the development of such systems. Therefore the main problem on the way to industrialise existing prototype solutions is related to the proof of robustness, safety and reliability of the system. In this context, advanced and automated development and testing methodologies will play a crucial role.

Exhaustive testing is impossible what means that testing everything (all combinations of inputs and preconditions) is not feasible except for trivial cases. This is valid in particular for software systems developed for automated driving applications. The number of important road scenarios for such system is actually infinite. Currently, all prototypes of active safety, advanced driver assistance and automated driving control systems must be verified in real conditions. This means that the system must be verified on a data set collected from about 1 million kilometres as this is a very basic car manufacturer's requirement. Based

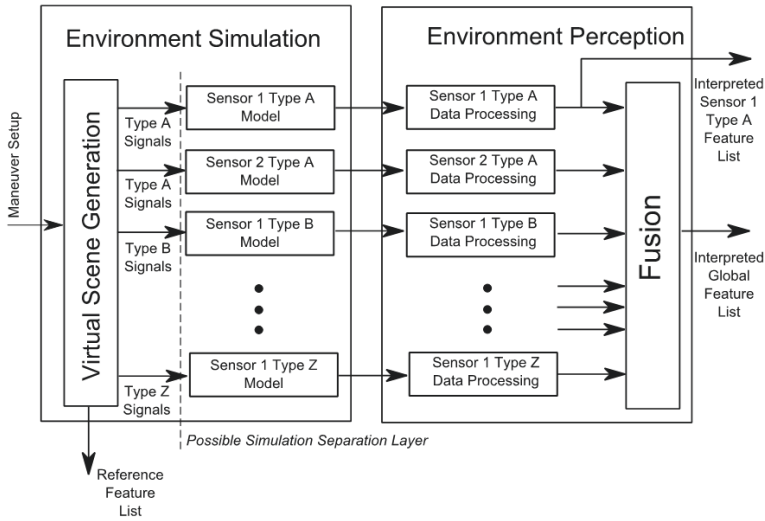


Fig. 8. Verification of automated driving perception algorithms using virtual scene generation

on the data collected, performance and reliability metrics are calculated. It is also estimated that for control systems implementing features of highest levels of automation according to SAE or NHTSA ranking, there will be a need to collect, store, re-process and analyse data from many millions of kilometres in order to proof the proper quality level of the system. A solution for this problem might be model-based testing and simulation approaches with artificially generated data. Then some of the tests can be done pretty fast in virtual environment without the need of having real vehicle drives (Fig. 8).

4 Conclusions

Elon Musk said that "Making rockets is hard, but making cars is really hard". Indeed, number of possible road scenarios that shall be properly handled by the perception systems in a vehicle is theoretically infinite. Much less demanding environment is valid for planes and rockets. So the straightforward conclusion would be that fully automated cars that can handle all possible road scenarios are by definition not possible. Nevertheless, the automotive industry will continue the development of automated vehicle towards this unreachable (at least theoretically) goal. Many car manufacturers and automotive suppliers have announced fully automated vehicle available in next years. However, at current stage of the development, we can say that there are nice prototypes available but there is far way in the front of us in order to industrialise these solutions.

There are still a lot of aspects that should be taken into account until all issues will be resolved. More sceptics predict these car available in 2035. We should expect this date would be rather close to 2035 than 2020.

References

1. Binder, R.: Testing Object-Oriented Systems: Models, Patterns, and Tools. Addison-Wesley, Boston, USA (1999)
2. Buchholz, K.: EETimes Europe: Model-based software development in the automotive industry. <http://www.electronics-eetimes.com/en/model-based-development.html> [16 April 2012] (2011)
3. Central Statistical Office: Road transport in Poland in the years 2012, 2013. <http://stat.gov.pl> [25 February 2017] (2015)
4. Commission of the European Communities: Directive 98/69/EC of the European parliament and of the Council of 13 October 1998 relating to measures to be taken against air pollution by emissions from motor vehicles and amending Council Directive 70/220/EEC. <http://europa.eu> [25 February 2017] (1998)
5. Commission of the European Communities: Commision Directive 2002/80/EC of 3 October 2002 adapting to technical progress Council Directive 70/220/EEC relating to measures to be taken against air pollution by emissions from motor vehicles. <http://europa.eu> [25 February 2017] (2002)
6. Commission of the European Communities: Communication from the commission. European road safety action programme. Halving the number of road accident victims in the European Union by 2010: a shared responsibility. <http://eur-lex.europa.eu> [25 February 2017] (2003)
7. Commission of the European Communities: Regulation (EC) No 715/2007 of the European parliament and of the Council of 20 June 2007 on type approval of motor vehicles with respect to emissions from light passenger and commercial vehicles (Euro 5 and Euro 6) and on access to vehicle repair and maintenance information. <http://eur-lex.europa.eu> [25 February 2017] (2007)
8. Commission of the European Communities: Road safety programme 2011-2020: detailed measures. MEMO/10/343. <http://europa.eu> [25 February 2017] (2010)
9. Douglass, B.: Uml statecharts. Embedded Systems Programming 12(1), 22–42 (1999)
10. FlexRay Spec.: FlexRay communications system protocol specification, ver. 2.1, rev. A. <http://www.flexray> [7 May 2010] (2005)
11. Hahn, G., Philipps, J., Pretschner, A., Stauner, T.: Technical report TUM-I0301: Tests for mixed discrete-continuous systems. Institute for Computer Science, Technical University of Munich (2003)
12. Harel, D.: Statecharts: a visual formalism for complex systems. Science of Computer Programming 8(3), 231–274 (1987)
13. ISO 11898-1:2003: International Standard ISO 11898-1:2003: Road vehicles – Controller area network (CAN) – Part 1: Data link layer and physical signalling. <http://www.iso.org> [16 April 2012] (2003)
14. ISO 11898-2:2003: International Standard ISO 11898-2:2003: Road vehicles – Controller area network (CAN) – Part 2: High-speed medium access unit. <http://www.iso.org> [16 April 2012] (2003)
15. LIN Spec.: LIN specification package, rev. 2.1. <http://www.lin-subbus.org> [16 April 2012] (2006)

16. McKinsey Global Institute: Disruptive technologies: advances that will transform life, business, and the global economy. <http://www.mckinsey.com> [25 February 2017] (2014)
17. MOST Spec.: MOST dynamic specification, rev. 3.0, 05/2008. <http://www.mostcorporation.com> [7 May 2010] (2008)
18. Myers, G.: The Art of Software Testing, 2nd ed. John Wiley & Sons, New York, USA (2004)
19. Papoulis, A.: Circuits and Systems: A Modern Approach. Holt, Rinehart, and Winston, New York, USA (1980)
20. Skruch, P., Dlugosz, R., Kogut, K., Markiewicz, P., Sasin, D., Rozewicz, M.: The simulation strategy and its realization in the development process of active safety and advanced driver assistance systems. SAE Technical Paper 2015-01-1401 (2015)
21. The Society of Automotive Engineers (SAE): J3016: Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles. <http://standards.sae.org> [25 February 2017] (2016)
22. Tung, J.: EE Times. Using model-based design to test and verify automotive embedded software. <http://www.eetimes.com/showArticle.jhtml?articleID=202100792> [04 May 2012] (2007)
23. U.S. Department of Transportation, National Highway Traffic Safety Administration (NHTSA): Human factors evaluation of level 2 and level 3 automated driving concepts (DOT HS 812 044). <https://www.nhtsa.gov> [25 February 2017] (2014)
24. Waldrop, M.: Autonomous vehicles: no drivers required. *Nature* 518(7537), 20–23 (2015)
25. World Health Organization: Global status report on road safety 2013: supporting a decade of action. <http://www.who.int> [25 February 2017] (2013)

Stability Analysis of a Series of Cars Driving in Adaptive Cruise Control Mode

Paweł Skruch, Marek Długosz, Wojciech Mitkowski

AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics,
Computer Science and Biomedical Engineering
Department of Automatics and Biomedical Engineering
al. A. Mickiewicza 30/B1, 30-059 Krakow, Poland
pawel.skruch@agh.edu.pl, mdlugosz@agh.edu.pl,
wojciech.mitkowski@agh.edu.pl

Abstract. The paper analysis the stability property of a series of cars following each other and functioning in adaptive cruise control mode. The adaptive cruise control mode controls the acceleration and deceleration of the car in order to maintain a set speed or to avoid a crash. Such series of cars can be mathematically represented by an equivalent system consisting of a set of masses, springs and dampers. Using this representation, the dynamic behaviour of the cars in a chain can be described by a matrix-vector differential equation of second-order. The paper analyses this model and formulates stability conditions.

Keywords: stability, adaptive cruise control, car

1 Introduction

Adaptive cruise control (ACC) is a control system which function is to keep safe distance from the vehicle ahead. Driver sets desired speed and time interval to the car ahead. When system detects slower vehicle the speed is automatically adapted so the vehicle ahead is followed with a setup distance between. Once road is clear again the car returns to the selected speed. The ACC functionality is based mostly on vision and radar sensor systems. Radar provides high quality range and velocity information (see Fig. 1) and camera provides high quality object detection. Fusion algorithms are used to combine both radar and camera data for reliable target detection. The detected target vehicle is used by the ACC application as the vehicle to be followed (vehicle marked in red rectangle on Fig. 2).

ACC systems are based on various control theory methods. A review of the applied methods can be found in [8, 33]. Proportional-Integral (PI) and Linear Quadratic (LQ) controllers [31], Balance-Based Adaptive Control (B-BA) algorithm methods [30], quadratic optimal control [1], Proportional-Integral-Derivative (PID) controllers [6], Sliding Mode Control (SLM) [9, 14] can be enumerated as examples of possible to implement control algorithms. Model

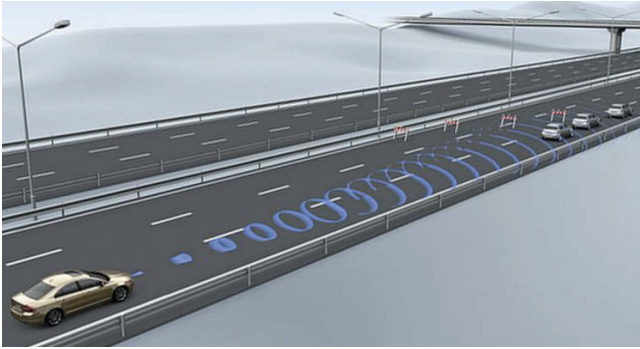


Fig. 1. Radar provides range and velocity information for ACC functionality



Fig. 2. Data fusion combines radar and camera information for reliable target detection

predictive control is also popular control method of ACC systems [2, 24, 29] as well as neural [4, 25] and fuzzy controllers [5, 28]. The majority of ACC control systems can be considered as hybrid control systems as those systems contain usually a logical part responsible for appropriate classification of an occurrence and a relevant controller for the classified situation [10, 22, 34].

In recent years there has been a growing attention to studies on ladder networks because they are strictly correlated to integrated interconnection problems, coupled mechanical systems, analog neural nets, distributed amplifiers, and so on. Ladder networks may be described as networks formed by numerous repetitions of an elementary cell. In case of a mechanical ladder network, the elementary cell may consist of masses, springs, and dampers connected in series or in parallel. Electrical ladder networks have similar dynamics as mechanical ladder networks and they consist of resistors, inductance coils, and capacitors. If all the elementary cells are identical, the ladder network is said to be homo-

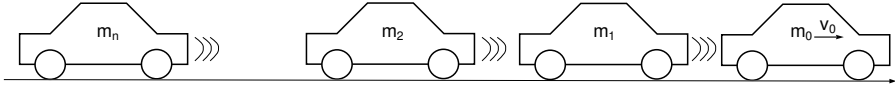


Fig. 3. Series of n -cars following each other in adaptive cruise control mode



Fig. 4. Mechanical equivalent model of a series of n -cars driving in adaptive cruise control mode

geneous; if the elementary cells are not identical, the ladder network is called inhomogeneous. The properties of ladder networks, especially electrical ladder networks, have been already studied in the past. Control problems for linear RL, RC, LC, and RLC electrical circuits are widely discussed in [3, 16, 18, 21, 20]. The dynamics and detailed characteristics of nonlinear electrical circuits are considered in [7, 15]. Control problems for nonlinear RLC circuits are discussed in [11, 12, 23, 26].

In this paper we investigate stability of a chain of cars following each other and functioning in adaptive cruise control mode. The car indicated as m_0 is chosen to lead the chain (see Fig. 3). It is assumed that the system is in an equilibrium state. This means that all cars in the chain have desired speed set to at least the same value as the leading car. Thus, in the equilibrium state each car in the chain shall maintain the safe distance to the car ahead only. The goal is to investigate dynamic behavior of the cars in the situation when the leading car will do an unexpected manoeuvre as acceleration or braking.

2 Mathematical Models

A series of cars following each other and functioning in adaptive cruise control mode can be represented by an equivalent system consisting of a set of masses and springs as depicted on Fig. 4. The masses m_i , where $i = 1, 2, \dots, n$, correspond to the masses of the cars, m_0 is the mass of the leading car. The coefficients k_i , where $i = 1, 2, \dots, n$, are related to gains of the controllers responsible to keep a set safe distance to the vehicle ahead.

The dynamic behaviour of n -cars driving in adaptive cruise control mode can be represented by a matrix-vector linear differential equation of second-order of the following form

$$\mathbf{E}\ddot{\mathbf{x}}(t) + \mathbf{A}\mathbf{x}(t) = \mathbf{0}, \quad \mathbf{x}(0) = \mathbf{x}_0, \quad \dot{\mathbf{x}}(0) = \mathbf{x}_{d0}, \quad (1)$$

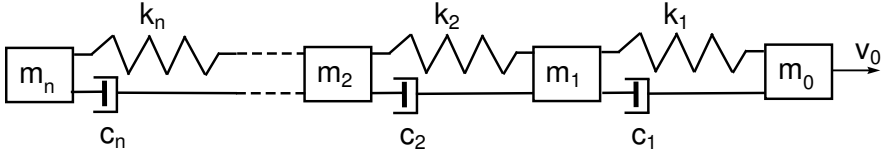


Fig. 5. Mechanical equivalent model with damping elements of a series of n -cars driving in adaptive cruise control mode

where $\mathbf{x}(t) = [x_1(t) \ x_2(t) \ \dots \ x_n(t)]^T \in \mathbb{R}^n$ is a vector representing displacements of the masses from the equilibrium state, $\mathbf{x}_0 \in \mathbb{R}^n$, $\mathbf{x}_{d0} \in \mathbb{R}^n$ are given initial conditions, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{E} \in \mathbb{R}^{n \times n}$ are matrices of the order $n \times n$ and the following form

$$\mathbf{A} = \begin{bmatrix} k_1 + k_2 & -k_2 & 0 & \dots & 0 & 0 & 0 \\ -k_2 & k_2 + k_3 & -k_3 & \dots & 0 & 0 & 0 \\ 0 & -k_3 & k_3 + k_4 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & k_{n-2} + k_{n-1} & -k_{n-1} & 0 \\ 0 & 0 & 0 & \dots & -k_{n-1} & k_{n-1} + k_n & -k_n \\ 0 & 0 & 0 & \dots & 0 & -k_n & k_n \end{bmatrix}_{n \times n}, \quad (2)$$

$$\mathbf{E} = \text{diag} (m_1, m_2, m_3, \dots, m_{n-2}, m_{n-1}, m_n). \quad (3)$$

By adding damping elements to the model from Fig. 4 we can include in the investigation the ACC controller that takes into consideration the derivative of a vehicle position. Then the dynamic equation will have an extended form, that is,

$$\mathbf{E}\ddot{\mathbf{x}}(t) + \mathbf{F}\dot{\mathbf{x}}(t) + \mathbf{A}\mathbf{x}(t) = \mathbf{0}, \quad \mathbf{x}(0) = \mathbf{x}_0, \quad \dot{\mathbf{x}}(0) = \mathbf{x}_{d0}, \quad (4)$$

where $\mathbf{F} \in \mathbb{R}^{n \times n}$ is a matrix of the order $n \times n$ and the form

$$\mathbf{F} = \begin{bmatrix} c_1 + c_2 & -c_2 & 0 & \dots & 0 & 0 & 0 \\ -c_2 & c_2 + c_3 & -c_3 & \dots & 0 & 0 & 0 \\ 0 & -c_3 & c_3 + c_4 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & c_{n-2} + c_{n-1} & -c_{n-1} & 0 \\ 0 & 0 & 0 & \dots & -c_{n-1} & c_{n-1} + c_n & -c_n \\ 0 & 0 & 0 & \dots & 0 & -c_n & c_n \end{bmatrix}_{n \times n}, \quad (5)$$

with the parameters $c_i > 0$ for $i = 1, 2, \dots, n$. The matrices \mathbf{A} and \mathbf{B} have the same meaning as in Eqs. (2) and (3).

In applications, values of the coefficients k_i , $i = 1, 2, \dots, n$ can depend on the vehicle positions as during braking and acceleration usually various settings

of the control algorithms are used. Also the matrix \mathbf{F} should be considered in general as a matrix with nonlinear elements if we would like to include into the model aerodynamic or other types of friction forces. In such situation, the dynamic behaviour of the vehicle chain can be described by a nonlinear differential equation of the following form

$$\mathbf{E}\ddot{\mathbf{x}}(t) + \mathbf{F}(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}}(t) + \mathbf{A}(\mathbf{x})\mathbf{x}(t) = \mathbf{0}, \quad \mathbf{x}(0) = \mathbf{x}_0, \quad \dot{\mathbf{x}}(0) = \dot{\mathbf{x}}_{d0}. \quad (6)$$

Here, $\mathbf{E} \in \mathbb{R}^{n \times n}$ is a square matrix with real entries as described in Eq. (3), $\mathbf{F} : \mathbb{R}^n \times \mathbb{R}^n \supset \Omega \times \Omega \rightarrow \mathbb{R}^{n \times n}$ and $\mathbf{A} : \mathbb{R}^n \supset \Omega \rightarrow \mathbb{R}^{n \times n}$ are matrices whose elements are nonlinear functions, that is, $\mathbf{F}(\boldsymbol{\xi}, \boldsymbol{\eta}) = [f_{ij}(\boldsymbol{\xi}, \boldsymbol{\eta})]_{n \times n}$, $\mathbf{A}(\boldsymbol{\xi}) = [a_{ij}(\boldsymbol{\xi})]_{n \times n}$, $i, j = 1, 2, \dots, n$, $\Omega \subset \mathbb{R}^n$ is a neighborhood of zero ($\mathbf{0} \in \mathbb{R}^n$). The matrices \mathbf{A} and \mathbf{F} have the same symmetric tridiagonal structure as in Eqs. (2) and (5) except that in place of the constant coefficients k_i and c_i are nonlinear functions $k_i(\mathbf{x})$ and $c_i(\mathbf{x}, \dot{\mathbf{x}})$ which are continuous with continuous derivatives with respect to each variable in the set Ω .

3 Stability Analysis

In this section we investigate stability properties of the systems described by Eqs. (1), (4) and (6).

Lemma 1. *The matrix \mathbf{A} is positive definite.*

Proof. As the matrix \mathbf{A} is symmetric with real values, all its eigenvalues are real numbers. According to the Gershgorin circle theorem [32] every eigenvalue of the matrix \mathbf{A} lies within at least one of the Gershgorin discs $D(a_{ii}, R_i)$, where

$$a_{ii} = k_i + k_{i+1} \text{ for } i = 1, 2, \dots, n-1 \text{ and } a_{nn} = k_n, \quad (7)$$

is the centred point of the disc that is equal to the i -th diagonal element of the matrix \mathbf{A} ,

$$R_1 = k_2 \quad R_i = k_i + k_{i+1} \text{ for } i = 2, 3, \dots, n-1 \text{ and } R_n = k_n, \quad (8)$$

is the radius of the disc that is calculated as the sum of the absolute values of the non-diagonal entries in the i -th row of the matrix \mathbf{A} . It is easy to notice that all Gershgorin discs $D(a_{ii}, R_i)$ lie in the right-half of complex plane including zero. Thus all eigenvalues of the matrix \mathbf{A} must be non-negative. It should be also noticed that the matrix $-\mathbf{A}$ is the Metzler matrix (see [19]) as its all off-diagonal entries are nonnegative. Following the analysis presented in [17] it can be concluded that $\det \mathbf{A} \neq 0$. Consequently, the matrix \mathbf{A} is positive definite.

In a similar way we can prove the following lemma.

Lemma 2. *The matrix \mathbf{F} is positive definite.*

Theorem 1. *The system (1) is oscillatory.*

Proof. Stability of the system (1) is determined by its eigenvalues, that is, roots of the characteristic equation

$$(\lambda_i \mathbf{E} + \mathbf{A}) v_i = 0, \quad i = 1, 2, \dots, n, \quad (9)$$

where $v_i = v_{R_i} + jv_{I_i}$ is a eigenvector corresponding to the eigenvalue λ_i , v_{R_i} , v_{I_i} stand for the real and imaginary parts of the vector v_i . Multiplying left the equation (9) by v_i^* we get

$$\lambda_i^2 v_i^* \mathbf{E} v_i + v_i^* \mathbf{A} v_i = 0, \quad i = 1, 2, \dots, n. \quad (10)$$

Because the matrices \mathbf{E} and \mathbf{A} are symmetric, thus

$$v_{R_i}^T \mathbf{E} v_{I_i} = v_{I_i}^T \mathbf{E} v_{R_i}, \quad (11)$$

$$v_{R_i}^T \mathbf{A} v_{I_i} = v_{I_i}^T \mathbf{A} v_{R_i}. \quad (12)$$

From Eq. (10) we can obtain

$$\alpha_{ei} \lambda_i^2 + \alpha_{ai} = 0, \quad i = 1, 2, \dots, n, \quad (13)$$

where

$$\alpha_{ei} = v_{R_i}^T \mathbf{E} v_{R_i} + v_{I_i}^T \mathbf{E} v_{I_i}, \quad (14)$$

$$\alpha_{ai} = v_{R_i}^T \mathbf{A} v_{R_i} + v_{I_i}^T \mathbf{A} v_{I_i}. \quad (15)$$

For positive definite matrices \mathbf{E} and \mathbf{A} the parameters α_{ei} and α_{ai} are positive. In such case, the eigenvalues of the system (9) are placed on imaginary axis, that is,

$$\lambda_i = \pm j \sqrt{\frac{\alpha_{ai}}{\alpha_{ei}}}, \quad j^2 = -1, \quad i = 1, 2, \dots, n. \quad (16)$$

This proves that the system (1) is oscillatory.

Theorem 2. *The system (4) is asymptotically stable.*

Proof. The characteristic equation of the system (4) has the form

$$(\lambda_i^2 \mathbf{E} + \lambda_i \mathbf{F} + \mathbf{A}) v_i = 0, \quad i = 1, 2, \dots, n, \quad (17)$$

where $v_i = v_{R_i} + jv_{I_i}$ is a eigenvector corresponding to the eigenvalue λ_i , v_{R_i} , v_{I_i} are the real and imaginary parts of the vector v_i . Let multiply left the equation (17) by v_i^*

$$\lambda_i^2 v_i^* \mathbf{E} v_i + \lambda_i v_i^* \mathbf{F} v_i + v_i^* \mathbf{A} v_i = 0, \quad i = 1, 2, \dots, n. \quad (18)$$

Because the matrices \mathbf{E} , \mathbf{F} and \mathbf{A} are symmetric thus

$$v_{R_i}^T \mathbf{E} v_{I_i} = v_{I_i}^T \mathbf{E} v_{R_i}, \quad (19)$$

$$v_{R_i}^T \mathbf{F} v_{I_i} = v_{I_i}^T \mathbf{F} v_{R_i}, \quad (20)$$

$$v_{R_i}^T \mathbf{A} v_{I_i} = v_{I_i}^T \mathbf{A} v_{R_i}. \quad (21)$$

Using (19), (20) and (21) into (18) we have

$$\alpha_{ei} \lambda_i^2 + \alpha_{fi} \lambda_i + \alpha_{ai} = 0, \quad i = 1, 2, \dots, n, \quad (22)$$

where

$$\alpha_{ei} = v_{R_i}^T \mathbf{E} v_{R_i} + v_{I_i}^T \mathbf{E} v_{I_i}, \quad (23)$$

$$\alpha_{fi} = v_{R_i}^T \mathbf{F} v_{R_i} + v_{I_i}^T \mathbf{F} v_{I_i}, \quad (24)$$

$$\alpha_{ai} = v_{R_i}^T \mathbf{A} v_{R_i} + v_{I_i}^T \mathbf{A} v_{I_i}. \quad (25)$$

For positive definite matrices \mathbf{E} , \mathbf{F} and \mathbf{A} the parameters α_{ei} , α_{fi} and α_{ai} are positive. In such case, the eigenvalues of the system (17) are located in the open left half plane. This means that the system (4) is asymptotically stable

Lemma 3. *If $\langle \mathbf{x}, \mathbf{A}(\mathbf{x})\mathbf{x} \rangle > 0$ for $\mathbf{x} \in \Omega \setminus \{\mathbf{0}\}$, then the line integral $\int_0^{\mathbf{x}} \mathbf{x}^T \mathbf{A}(\boldsymbol{\xi}) d\boldsymbol{\xi}$ along the straight line in the space \mathbb{R}^n from the beginning point $\mathbf{0}$ to the ending point \mathbf{x} ($\mathbf{x} \neq \mathbf{0}$) is positive.*

Proof. The proof can be conducted in the same manner as in the proof of similar lemma of [27].

Theorem 3. *The zero equilibrium point of the system (6) is locally asymptotically stable (in the Lyapunov sense).*

Proof. Consider the following Lyapunov functional

$$V(\tilde{\mathbf{x}}) = \frac{1}{2} \dot{\tilde{\mathbf{x}}}(t)^T \mathbf{E} \dot{\tilde{\mathbf{x}}}(t) + \int_0^{\mathbf{x}(t)} \boldsymbol{\xi}^T \mathbf{A}(\boldsymbol{\xi}) d\boldsymbol{\xi}, \quad (26)$$

where $\tilde{\mathbf{x}}(t) = \text{col}(\dot{\mathbf{x}}(t), \mathbf{x}(t))$. It can be concluded with the help of Lemma 3 that $V(\tilde{\mathbf{x}}) > 0$ for $\tilde{\mathbf{x}} \neq \mathbf{0}$ and $V(\tilde{\mathbf{x}}) = 0$ for $\tilde{\mathbf{x}} = \mathbf{0}$.

The derivative of V with respect to time t can be described by the following equation

$$\dot{V}(\tilde{\mathbf{x}}) = \dot{\tilde{\mathbf{x}}}(t)^T \mathbf{E} \ddot{\tilde{\mathbf{x}}}(t) + \nabla_{\mathbf{x}} \left(\int_0^{\mathbf{x}(t)} \boldsymbol{\xi}^T \mathbf{A}(\boldsymbol{\xi}) d\boldsymbol{\xi} \right) \dot{\tilde{\mathbf{x}}}(t), \quad (27)$$

and next

$$\dot{V}(\tilde{\mathbf{x}}) = \dot{\tilde{\mathbf{x}}}(t)^T \mathbf{E} \ddot{\tilde{\mathbf{x}}}(t) + \mathbf{x}(t)^T \mathbf{A}(\mathbf{x}) \dot{\tilde{\mathbf{x}}}(t). \quad (28)$$

Along the solutions of the system (6) it holds that

$$\dot{V}(\tilde{\mathbf{x}}) = \dot{\tilde{\mathbf{x}}}(t)^T (-\mathbf{F}(\dot{\mathbf{x}}, \mathbf{x}) \dot{\tilde{\mathbf{x}}}(t) - \mathbf{A}(\mathbf{x}) \mathbf{x}(t)) + \mathbf{x}(t)^T \mathbf{A}(\mathbf{x}) \dot{\tilde{\mathbf{x}}}(t), \quad (29)$$

what is equivalent to

$$\dot{V}(\tilde{\mathbf{x}}) = -\dot{\tilde{\mathbf{x}}}(t)^T \mathbf{F}(\dot{\mathbf{x}}, \mathbf{x}) \dot{\tilde{\mathbf{x}}}(t) \leq 0. \quad (30)$$

According to LaSalle's theorem [13], the trajectories of the system (6) enter asymptotically the largest invariant set in S , where

$$S = \left\{ \tilde{\mathbf{x}} \in \Omega_c : \dot{V}(\tilde{\mathbf{x}}) = 0 \right\}, \quad (31)$$

and Ω_c for $c > 0$ is a compact set defined as follows

$$\Omega_c = \left\{ \tilde{\mathbf{x}} \in \Omega \times \Omega \subset \mathbb{R}^{2n} : V(\tilde{\mathbf{x}}) < c \right\}. \quad (32)$$

It should be noted that $V(\tilde{\mathbf{x}}) > 0$ for $\tilde{\mathbf{x}} \in \Omega_c \setminus \{\mathbf{0}\}$, $V(\mathbf{0}) = 0$ and $\dot{V}(\tilde{\mathbf{x}}) \leq 0$ for $\tilde{\mathbf{x}} \in \Omega_c$. From the condition $\dot{V}(\tilde{\mathbf{x}}) = 0$ it follows that $\dot{\mathbf{x}}(t) = \mathbf{0}$ and based on (6) $\mathbf{x}(t) = \mathbf{0}$. This means that S contains zero equilibrium point of the system (6) only, thus $S = \{\mathbf{0}\}$. In result, the origin $\mathbf{0} \in \mathbb{R}^{2n}$ is asymptotically stable (in the Lyapunov sense).

4 Conclusions

The paper presents investigations on the stability of a series of cars following each other and functioning in adaptive cruise control mode. The main conclusion is that dynamics of such system is mathematically described by differential equations of second-order. This implies that in the system undamped or damped oscillations might occur. To avoid or minimise these oscillations the controller responsible for keeping the safe distance to the vehicle ahead shall take into consideration both position and velocity information.

References

1. Ames, A., Xu, X., Grizzle, J., Tabuada, P.: Control barrier function based quadratic programs with application to adaptive cruise control. In: Proceedings of the IEEE Conference on Decision and Control. pp. 6271–6278 (2015)
2. Bageshwar, V., Garrard, W., Rajamani, R.: Model predictive control of transitional maneuvers for adaptive cruise control vehicles. IEEE Transactions on Vehicular Technology 53(5), 1573–1585 (2004)
3. Baranowski, J., Mitkowski, W.: Stabilisation of lc ladder network with the help of delayed output feedback. Control and Cybernetics 41(1), 13–34 (2012)
4. Bengtsson, J.: Adaptive Cruise Control and Driver Modeling. Department of Automatic Control, Lund Institute of Technology, Lund, Sweden (2001)
5. Chakroborty, P., Kikuchi, S.: Evaluation of the general motors based car-following models and a proposed fuzzy inference model. Transportation Research Part C: Emerging Technologies 7(4), 209–235 (1999)
6. Chien, C., Ioannou, P., Lai, M.: Entrainment and vehicle following controllers design for autonomous intelligent vehicles. In: Proceedings of the American Control Conference. pp. 6–10 (1994)
7. Dabrowski, M.: Selected ideas of the theory of nonlinear electrical circuits. COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering 18(2), 204–214 (1999)

8. Dongbin, Z., Zhongpu, X.: Adaptive optimal control for the uncertain driving habit problem in adaptive cruise control system. In: 2013 IEEE International Conference on Vehicular Electronics and Safety (ICVES). pp. 159–164 (2013)
9. Ganji, B., Kouzani, A., Khoo, S., Nasir, M.: A sliding-mode-control-based adaptive cruise controller. In: Proceedings of the 11th IEEE International Conference on Control Automation (ICCA). pp. 394–397 (2014)
10. Ioannou, P., Chien, C.: Autonomous intelligent cruise control. *IEEE Transactions on Vehicular Technology* 42(4), 657–672 (1993)
11. J., J., Ortega, R., Scherpen, J.: On passivity and power-balance inequalities of nonlinear RLC circuits. *IEEE Transactions on Circuits and Systems Part I: Fundamental Theory and Applications* 50(9), 1174–1179 (2003)
12. J., J., Ortega, R., Scherpen, J.: Power shaping: a new paradigm for stabilization of nonlinear RLC circuits. *IEEE Transactions on Automatic Control* (Special Issue on New Directions in Nonlinear Control) 48(10), 1162–1167 (2003)
13. LaSalle, J., Lefschetz, S.: *Stability by Liapunov's Direct Method with Applications*. Academic Press, New York, London (1961)
14. Lingyun, X., Gao, F.: Practical string stability of platoon of adaptive cruise control vehicles. *IEEE Transactions on Intelligent Transportation Systems* 12(4), 1184–1194 (2011)
15. Mitkowski, S.: *Nonlinear Electric Circuits*. Wydawnictwa AGH, Cracow, Poland (1999)
16. Mitkowski, W.: *Stabilization of Dynamic Systems*. WNT, Warsaw, Poland (1991)
17. Mitkowski, W.: Remarks on stability of positive linear systems. *Control and Cybernetics* 29(1), 295–304 (2000)
18. Mitkowski, W.: Dynamic feedback in LC ladder network. *Bulletin of the Polish Academy of Sciences: Technical Sciences* 51(2), 173–180 (2003)
19. Mitkowski, W.: Dynamical properties of metzler systems. *Bulletin of the Polish Academy of Sciences: Technical Sciences* 56(4), 309–312 (2003)
20. Mitkowski, W.: Analysis of undamped second order systems with dynamic feedback. *Control and Cybernetics* 33(4), 563–572 (2004)
21. Mitkowski, W.: Stabilisation of LC ladder network. *Bulletin of the Polish Academy of Sciences: Technical Sciences* 52(2), 109–114 (2004)
22. Moon, S., Moon, I., Yi, K.: Design, tuning, and evaluation of a full-range adaptive cruise control system with collision avoidance. *Control Engineering Practice* 17(4), 442–455 (2009)
23. Oksasoglu, A., Vavriv, D.: Interaction of low- and high-frequency oscillations in a nonlinear RLC circuit. *IEEE Transactions on Circuits and Systems Part I: Fundamental Theory and Applications* 41(10), 669–672 (1994)
24. Shengbo, L., Keqiang, L., Rajamani, R., Wang, J.: Model predictive multi-objective vehicular adaptive cruise control. *IEEE Transactions on Control Systems Technology* 19(3), 556–566 (2011)
25. Simonelli, F., Nicola, B., Martinis, V.D., Punzo, V.: Human-like adaptive cruise control systems through a learning machine approach. In: *Applications of Soft Computing*. pp. 240–249 (2009)
26. Skruch, P.: Stabilization of nonlinear RLC ladder network. In: *Proceedings of the 7th Conference on Computer Methods and Systems*, 26–27.11.2009, Krakow, Poland. pp. 259–264 (2009)
27. Skruch, P.: Stabilization of a class of siso nonlinear systems by dynamic feedback. *Automatyka* 14(2), 197–209 (2010)
28. Spooner, J., Passino, K.: Stable adaptive control using fuzzy systems and neural networks. *IEEE Transactions on Fuzzy Systems* 4(3), 339–359 (1996)

29. Stanger, T., del Re, L.: A model predictive cooperative adaptive cruise control approach. In: Proceedings of the American Control Conference (ACC). pp. 1374–1379 (2013)
30. Syakouri, P., Czebot, J., Ordys, A.: Adaptive cruise control system using balance-based adaptive control technique. In: Proceedings of the 17th International Conference on Methods and Models in Automation and Robotics (MMAR). pp. 510–515 (2012)
31. Syakouri, P., Ordys, A., Laila, D., Askari, M.: Adaptive cruise control system: comparing gain-scheduling pi and lq controllers. In: Proceedings of the 18th IFAC World Congress. pp. 12964–12969 (2011)
32. Turowicz, A.: Geometry of zeros of polynomials. PWN, Warsaw, Poland (1967)
33. Vahidi, A., Eskandarian, A.: Research advances in intelligent collision avoidance and adaptive cruise control. *IEEE Transactions on Intelligent Transportation Systems* 4(3), 143–153 (2003)
34. Zhao, D., Hu, Z.: Supervised adaptive dynamic programming based adaptive cruise control. In: Proceedings of the 2011 IEEE Symposium on Adaptive Dynamic Programming And Reinforcement Learning (ADPRL). pp. 318–323 (2011)

A Formal Approach for the Verification of Control Systems in Autonomous Driving Applications

Paweł Skruch, Marek Długosz, Paweł Markiewicz

¹ AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics,
Computer Science and Biomedical Engineering
Department of Automatics and Biomedical Engineering
al. A. Mickiewicza 30/B1, 30-059 Krakow, Poland

² Delphi Technical Center Krakow

ul. Podgorki Tynieckie 2, 30-399 Krakow, Poland

pawel.skruch@agh.edu.pl, mdlugosz@agh.edu.pl, pawel.markiewicz@delphi.com

Abstract. Control systems in autonomous vehicles can be considered as distributed embedded software systems where independent microprocessor systems communicate together using different communication protocols. Typical autonomous driving functionality is then realised by several microprocessors communicating with each other. Quality assurance and safety standards combined with increasing complexity and reliability demands make the development of such systems challenging. In order to assure the required quality and compliance with safety standards, a formal and methodical approach for testing and verification is required. The paper presents a proposal of such approach for verification and testing of control systems in the automotive applications covering active safety, advanced driver assistance and autonomous driving systems. The main focus of this approach is black-box testing and includes test design, implementation and execution.

Keywords: embedded system, autonomous vehicle, testing, verification

1 Introduction

Autonomous driving is topic of the day. Autonomous vehicles (sometimes called automated vehicles) are vehicles that can navigate and operate with reduced or no human intervention. The majority of automotive manufacturers are highlighting right now the benefits of this technology which definitely will enable in the near future a revolution in ground transportation. The potential benefits of autonomous vehicles include increased safety with the vision of zero road accidents, reduced carbon dioxide emissions, increased flow of cars in urban environments and increased productivity in the trucking industry.

A vehicle control system is a distributed embedded software system where independent microprocessor (very often multicore) systems, called Electronic

Control Units (ECUs), communicate with each other using communication networks. Each ECU in this distributed architecture can be considered as a sub-control system that processes continuously changing input signals and provides appropriate output signals based on the inputs and internal states. As those inputs and outputs include a mix of electrical, video, radar, light and communication signals, the development of such systems becomes especially challenging (see e.g., [18]).

Designing an embedded software system for automotive applications is a complex and error prone task. Within the last decades embedded systems have become increasingly sophisticated and their software content has grown rapidly. The increasing miniaturisation of embedded software systems on the one hand and rising functional demands on the other hand require advanced and automated development and testing methodologies [5, 13, 15, 19].

It is worth to mention the difference between testing and verification. Testing is the process of trying to discover every conceivable fault or weakness in a work product. Testing can show that defects are present, but cannot prove that there are no defects [10]. Testing reduces the probability of undiscovered defects remaining in the software but, even if no defects are found, it is not a proof of correctness. Verification is the process of evaluating a system to determine whether the product satisfies the requirements [6] what is a very basic manufacturers' demand. Poorly tested systems may cost producers billions of dollars annually especially when defects are found by end users in production environments [8, 9, 14]. Barry Boehm's research analysis [4] indicates that the cost of removing a software defect grows exponentially for each stage of the development life cycle in which it remains undiscovered. Boris Beizer [2] estimates that 30 up to 90 percentage of the effort is put into testing. Another research project conducted by the United States Department of Commerce, National Institute of Standards and Technology [11] estimated that software defects cost the U.S. economy \$60 billion per year.

Exhaustive testing is impossible what means that testing everything (all combinations of inputs and preconditions) is not feasible except for trivial cases. This is valid in particular for software systems developed for autonomous driving applications. The number of important road scenarios for such system is actually infinite. Testing dynamic aspects of autonomous driving requires tests that utilise time continuous input signals and time continuous output signals (even when the system is digitally processed). The process of selecting just a few of the many possible scenarios to be tested is a difficult and challenging task and currently is most often based on qualitative best engineering judgment [17].

The paper presents a formal approach for testing embedded software systems developed for autonomous driving applications. In the next section, test automation framework is presented. After that, the concept of testing with a model as an oracle is explained. In the following section, the system under test is specified and represented in the form of input/state/output, allowing tests to be described independently of test methodology, implementation and execution. The key idea is to describe all signals being part of a test case as one-dimensional continuous-

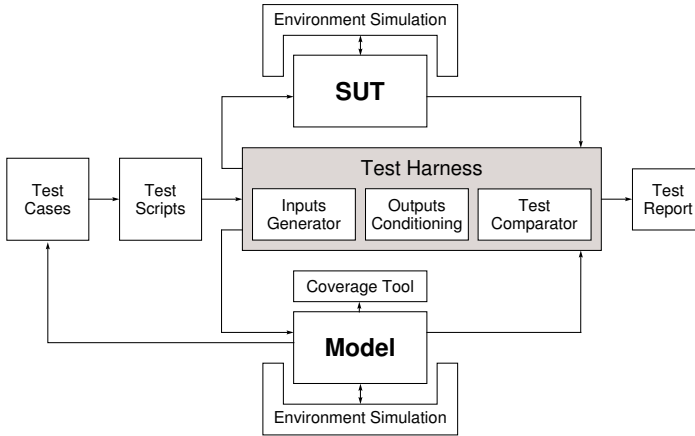


Fig. 1. Architecture of test automation framework.

time signals, with each one part of the test case mathematically described by a set of discrete time markers with linear interpolation. The details are provided in the next section. After that, test comparator mechanism is presented that is used to judge whether a test case passes or fails. The approach presented in the paper can be realised in a modelling framework to ensure efficient test automation and real-time execution as described in the last section before conclusions.

2 Testing Framework

Fig. 1 presents the main elements of the test automation framework used to implement a model-based, real-time testing approach dedicated for embedded control systems. In the presented framework models play an important role, as they are used to model the system under test (SUT) and possible behaviour of the SUT environment, automatically generate test cases, develop a test oracle mechanism, implement a test harness, calculate test coverage and report test results. Arrows in the diagram indicate the direction of the information flow between the elements.

3 Testing Concept

The term *test oracle* describes a source used to determine expected results to compare with the actual result of the SUT [1]. The role of such a source in the model-based approach is often played by the model assuming that it fully represents the requirements. Fig. 2 illustrates a possible test scenario where the same inputs (from a logical point of view) are applied to both the physical SUT

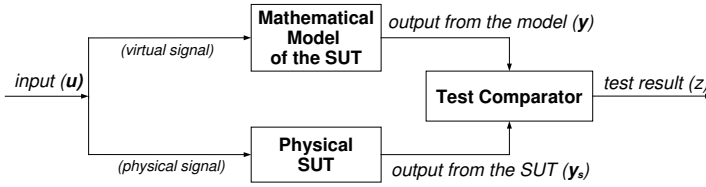


Fig. 2. Concept of testing with a model as an oracle

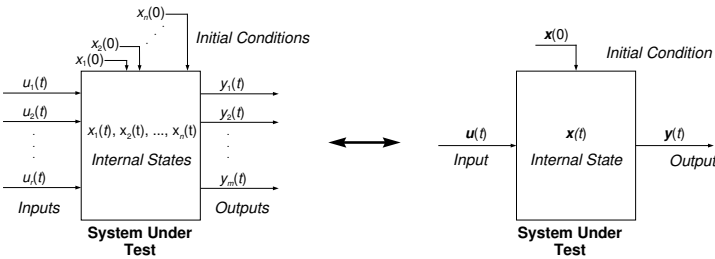


Fig. 3. State space modelling concept of the SUT

and the model. In this scenario, the signals are physical in the case of the SUT and virtual in the case of the model. The judgment of whether a test result conforms with the model is delegated to a test comparator, which is a tool that compares the actual output produced by the SUT with the expected output produced by the model.

4 Representation of the System Under Test

The modelling concept called *state space representation* (or input/state/output representation) provides a convenient way to model and analyse systems with multiple inputs and outputs. The state space model (Fig. 3) can represent the function, unit, module, system etc. that is being tested. It is constructed at a certain level of abstraction and describes the functionality of the system at that level. The state space model consists of a set of input $\{u_1(t), u_2(t), \dots, u_r(t)\}$, output $\{y_1(t), y_2(t), \dots, y_m(t)\}$, and internal state $\{x_1(t), x_2(t), \dots, x_n(t)\}$ variables that can be expressed as vectors, that is $\mathbf{u}(t)$, $\mathbf{y}(t)$ and $\mathbf{x}(t)$ respectively. The values of input, output and internal state variables may change over time t . The relationship between those variables is determined by the system requirements. For a distributed control system, each sub-system can be considered separately or in combination with other sub-systems as illustrated in Fig. 4.

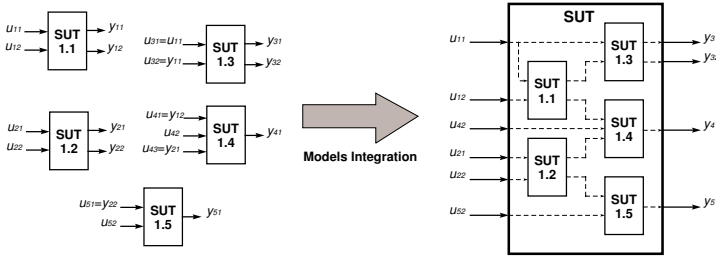


Fig. 4. System integration concept

5 The Proposed Test Notation

The state space modelling concept of the SUT illustrated in Fig. 3 implicates the notion of a single test case in the following form:

$$T_{\text{case}}^{(j)} = \left\{ T^{(j)}, \mathbf{x}_0^{(j)}, \mathbf{u}^{(j)}(\cdot), \mathbf{y}^{(j)}(\cdot) \right\}, \tag{1}$$

in case of black-box testing [3], or

$$T_{\text{case}}^{(j)} = \left\{ T^{(j)}, \mathbf{x}_0^{(j)}, \mathbf{u}^{(j)}(\cdot), \mathbf{x}^{(j)}(\cdot), \mathbf{y}^{(j)}(\cdot) \right\}, \tag{2}$$

in case of gray-box testing [12]. Here, $\mathbf{u}^{(j)} : [0, T^{(j)}] \rightarrow \mathbb{R}^r$ is an input vector function represented signals applied to the SUT, $\mathbf{x}^{(j)} : [0, T^{(j)}] \rightarrow \mathbb{R}^n$ is an expected vector state function of internal signals, and $\mathbf{y}^{(j)} : [0, T^{(j)}] \rightarrow \mathbb{R}^m$ is an expected vector output function represented measured signals on the SUT when the system starts from an initial condition $\mathbf{x}_0^{(j)}$, $j = 1, 2, \dots, N$ is a label to indicate different test cases. A collection of one or more test cases forms a test suite $T_{\text{suite}} = \left\{ T_{\text{case}}^{(1)}, T_{\text{case}}^{(2)}, \dots, T_{\text{case}}^{(N)} \right\}$.

In the implementation every one-dimensional continuous-time signal being part of the test case, (1) or (2), can be approximated by a set of discrete points with linear interpolation [16]. Such an approximated signal can then be characterized by a pair $(s_{\text{time}}, s_{\text{values}})$, where $s_{\text{time}} = [t_1 t_2 \dots t_k]^T$ stands for the time coordinate and $s_{\text{values}} = [v_1 v_2 \dots v_k]^T$ stands for the value coordinate of the signal. Fig. 5 illustrates this approximation approach for digital and analogue signals, typical in practical applications.

The essential element of the autonomous driving constitutes a 360 degree sensing system that enables object detections and interpretation of sensor information. The objects are provided by the sensing system, identified by other cars and delivered from 3D maps. Next, multi-domain fusion algorithms allow combining information coming from the sensors into reliable object detections. The list of objects is next used for implementation of the functionalities such as adaptive cruise control, autonomous emergency braking, lateral control, ad-hoc

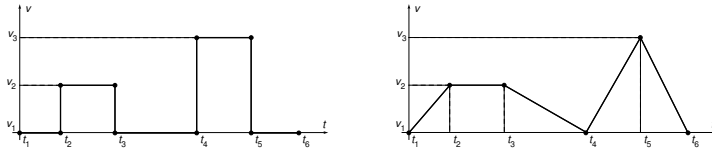


Fig. 5. Representation of digital (left) and analogue signals (right)

safety zones, control trajectories steering the vehicle from the starting point to the defined destination, etc. As those objects are coming from sensors of different nature (camera, radar, lidar, GPS) it might be difficult to describe them mathematically in unique way. Therefore, the key idea is to characterise every object (i.e., input to SUT) from the vehicle’s surrounding by a set of one-dimensional continuous time waveforms. This is nothing new as such process is performed by the environment perception algorithms which include data segmentation, clustering, labelling and classification. Figs. 6 and 7 present a typical test scenario for autonomous emergency braking functionality. The ego vehicle (marked as a red rectangle on Fig. 7) shall detect a pedestrian that is going to cross the street behind other car and then enable emergency braking action. Fig. 8 and 9 illustrate how to describe such test scenario using the notation proposed. First waveform represents the information about the presence of the object detected, the others represent position and velocity characteristics.

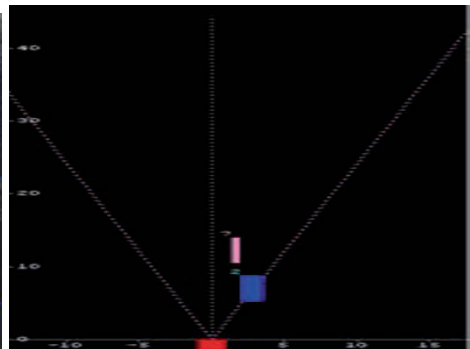


Fig. 6. A test scenario for autonomous emergency braking functionality (3D view) **Fig. 7.** A test scenario for autonomous emergency braking functionality (2D view)

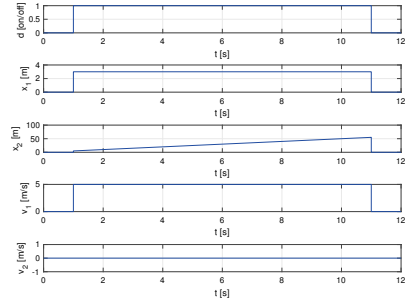
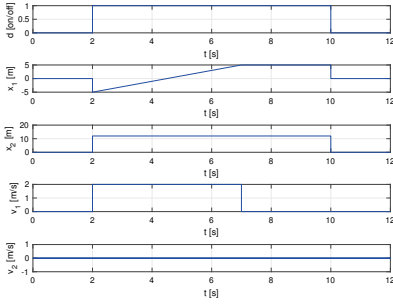


Fig. 8. 1D characteristics of a 'pedestrian' object in the test scenario for autonomous emergency braking functionality **Fig. 9.** 1D characteristics of a 'vehicle' type object in the test scenario for autonomous emergency braking functionality

6 Test Comparator Mechanism

The test comparator implements a mechanism for determining whether a test passes or fails [7]. The comparison mechanism for a given test case $T_{\text{case}}^{(j)}$ can be defined by tolerance range and expressed mathematically as follows:

$$z(T_{\text{case}}^{(j)}) = \begin{cases} 0 & \text{if } \forall_{t \in [0, T^{(j)}]} \forall_{i \in \{1, 2, \dots, m\}} \epsilon_{\text{low}}(t) \leq y_{si}^{(j)}(t) - y_i^{(j)}(t) \leq \epsilon_{\text{up}}(t), \\ 1 & \text{otherwise.} \end{cases} \quad (3)$$

The formula (3) means the executed test case is qualified as *pass* ($z = 0$) if every one-dimensional output produced by the SUT is within predefined tolerance ranges: lower tolerance ϵ_{low} and upper tolerance ϵ_{up} relative to the expected output, otherwise the test is qualified as *fail* ($z = 1$).

The implementation of a comparison mechanism for digital signals requires two additional tolerances [16]: left and right tolerances (see Fig. 10). This is caused by the fact that a digital signal is not a continuous function. The left tolerance means the maximum allowed difference in time if the step on the actual output signal appears before the step on the expected one. The right tolerance stands for the maximum allowed difference in time if the step on the actual output signal appears after the step on the reference signal.

7 Test Execution Harness

Fig. 11 presents a block diagram of the environment that can be used to invoke the SUT, provide test inputs, control and monitor execution and report test results. Such an environment is called a *test harness* or *test driver* [6] and can be implemented in most of the modelling frameworks. *Signal Builder* and

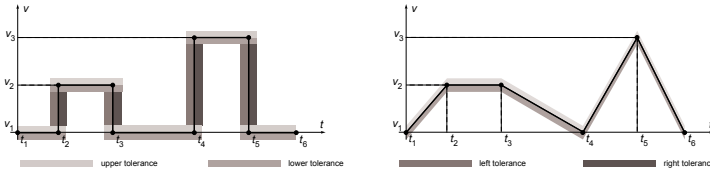


Fig. 10. Representation of expected digital (left) and analogue signals (right) with tolerance ranges

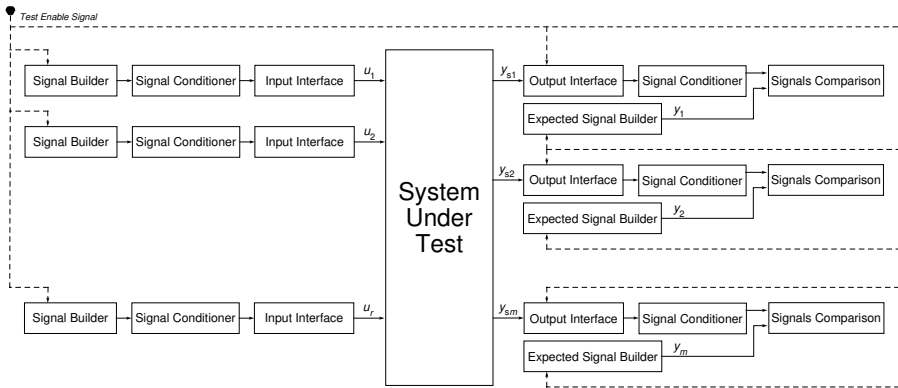


Fig. 11. Block diagram of the test harness

Expected Signal Builder blocks are used to create the waveform of the input signals applied to the SUT and the output signals expected from the SUT. *Signal Conditioner* blocks convert one type of the signal into another that is usually required and imposed by the test instrumentation. *Input Interface* and *Output Interface* blocks are used to communicate with external hardware, sensors and actuators. *Signal Comparison* blocks implement the comparison mechanism. All blocks are triggered and synchronised by a *Test Enable* signal.

The following part of this section describes the configuration of the test harness that has been designed using Simulink[®] tool in the MATLAB[®] environment. Besides standard Simulink[®] blocks dSPACE[™] Real-Time Interface (RTI) library has been used to link test application software with test system hardware.

Figs. 12 and 13 present examples of signal flow graphs generated by the test system which are then directed through the RTI interface to the corresponding inputs of the system being tested. Three types of flow graphs have been shown in these figures which correspond to digital, analogue and resistance signals (these signals are inputs for the tested system and outputs for the test system) and can be freely multiplied. When *IN_Signal_X_Enable* flag is unset, the corresponding

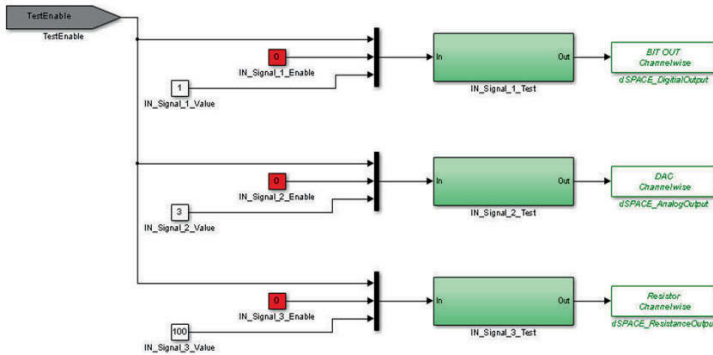


Fig. 12. Flow graph created in Simulink[®] for the input signals applied to the SUT

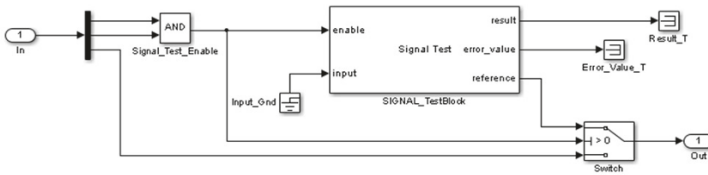


Fig. 13. Flow graph created in Simulink[®] for the input signals applied to the SUT (view of *IN_Signal_X_Test* subsystem)

signal is excluded from the experiment which is started when *TestEnable* flag is set. In this case, the signal can only be changed in manual mode. When *TestEnable* flag is set, then a waveform defined in *SIGNAL_TestBlock* will be applied to the corresponding input of the SUT.

Figs. 14 and 15 present examples of signal flow graphs measured by the test system and available through the RTI interface in the Simulink[®] model. When *OUT.Signal_X.Enable* flag is unset, this signal is excluded from the experiment when the *TestEnable* flag is set. When the *TestEnable* flag is set, the measured signal is compared with the reference signal waveform defined by *SIGNAL_TestBlock*. The *TestEnable* flag is then a synchronization trigger for all enabled input and output flow graphs.

SIGNAL_TestBlock (Fig. 16) is a Simulink[®] block that compares two signals named as *input* and *reference* in every time step, each with a given tolerance. The *input* signal shall be connected to the output of the SUT, while the *reference* is defined by time and value vectors delivered to the block as parameters. Fig. 16 shows the *Time* and *Reference* vectors. The user must define four kinds of tolerances: upper tolerance (maximum difference if the input signal is higher than the reference), lower tolerance (maximum difference if the input signal is below the reference), left tolerance (maximum difference if the step on the input signal

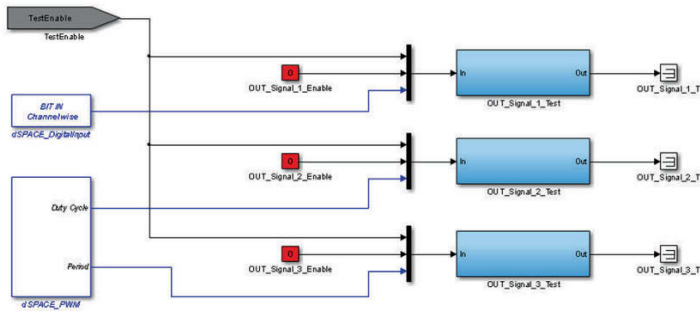


Fig. 14. Flow graph created in Simulink[®] for the output signals measured by the test system from the SUT

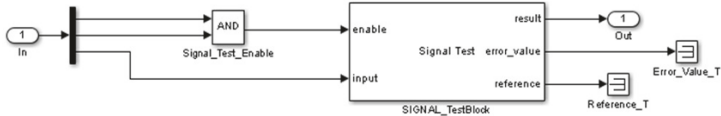


Fig. 15. Flow graph created in Simulink[®] for the output signals measured by the test system from the SUT (view of *OUT_Signal_X_Test* subsystem)

appears before the step on the reference signal) and right tolerance (maximum difference if the step on the input signal appears after the step on the reference signal). *SIGNAL_TestBlock* expects values for each time step. If the number of elements in *Reference vector* is smaller than the number of time steps (defined in *Time vector*), then the last value from the *Reference vector* is used. The left and right tolerances are not defined for time steps but for each step (trigger) in the reference signal. This means the block will trigger on the reference signal and check if the same trigger appears on input signal with the defined tolerance. *SIGNAL_TestBlock* can be used in a number of scenarios. In the first, two signals are compared with fixed tolerance values. In this case, all tolerances are defined as one element vector and set up from the *Block Parameters* window. Those values will be used during the whole experiment, because they are the last elements of the parameter vectors. In the second, two signals are compared with dynamic tolerance values. In this case, the user can define tolerances with the *Block Parameters* window as a vector with different values for each time step. The last value will be used for each time step of the experiment if the number of time steps exceeds the number of the vector elements. The *SIGNAL_TestBlock* values can be defined by *Signal Builder*, remembering that the signal configuration can be also imported to *Signal Builder* from MATLAB[®] Workspace from external tools such as csv files or MS Excel.

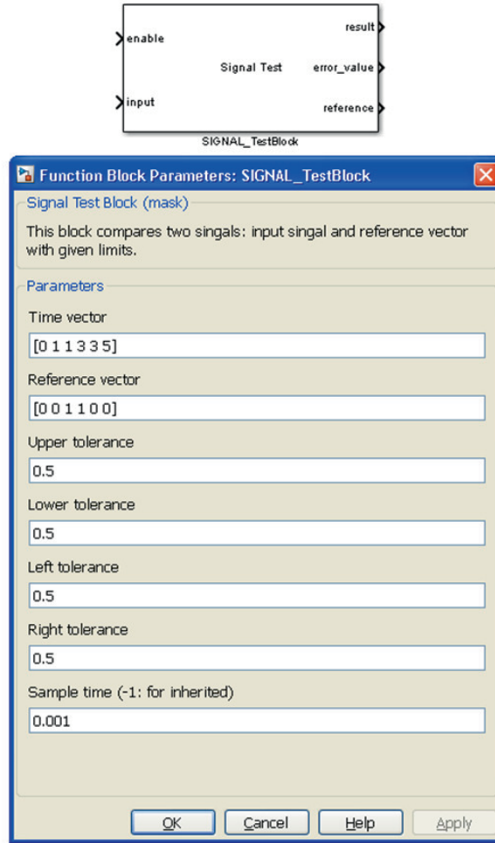


Fig. 16. `SIGNAL_TestBlock` and its parameters

8 Conclusions

The testing and verification methodology presented in this paper can be successfully used in safety critical applications. Test cases include continuous time signals what allows checking a system not only in discrete moments of time. Moreover, the approach can be easily implemented and executed on real-time platforms what means that test evaluation is done online and the results are provided immediately with the added benefits of test engineers not having to spend additional time and effort on offline analysis of test logs.

References

1. Adrion, W., Brandstad, J., Cherniabsky, J.: Validation, verification and testing of computer software. *Computing Surveys* 14(2), 159–192 (1982)

2. Beizer, B.: *Software Testing Techniques*. 2nd ed. Van Nostrand Reinhold, Boston, USA (1990)
3. Beizer, B.: *Black-Box Testing. Techniques for Functional Testing of Software and Systems*. John Wiley & Sons, New York, USA (1995)
4. Boehm, B.: *Software Engineering Economics*. Prentice Hall, Englewood Cliffs, USA (1981)
5. Buchholz, K.: *EETimes Europe: Model-based software development in the automotive industry*. <http://www.electronics-eetimes.com/en/model-based-development.html> [16 April 2012] (2011)
6. IEEE Std 610.12-1990: *IEEE standard glossary of software engineering terminology*. <http://www.standards.ieee.org> [16 April 2012] (1990)
7. ISTQB: *Standard glossary of terms used in software testing, Version 2.1*. <http://www.astqb.org> [16 April 2012] (2010)
8. Leveson, N., Turner, C.: *An investigation of the therac-25 accidents*. *IEEE Computer* 27(7), 18–41 (1993)
9. Lions, J.: *ARIANE 5. Flight 501 failure. Ariane 501 inquiry board report*. Paris, France (1996)
10. Myers, G.: *The Art of Software Testing*, 2nd ed. John Wiley & Sons, New York, USA (2004)
11. NIST: *National Institute of Standards & Technology, U.S. Department of Commerce: The economic impacts of inadequate infrastructure for software testing. Final report*. North Carolina, USA (2002)
12. Patton, R.: *Software Testing*, 2nd ed. Sams, Indianapolis, USA (2005)
13. Short, M., Pont, M.: *Assessment of high-integrity embedded automotive control systems using hardware-in-the-loop simulation*. *Journal of Systems and Software* 81(7), 1163–1183 (2008)
14. Skeel, R.: *Roundoff error and the patriot missile*. *Society for Industrial and Applied Mathematics (SIAM) News* 25(4), 11 (1992)
15. Skruch, P.: *A complete deployment of model-based and real-time approaches in verification of production automotive embedded systems*. In: *Proceedings of the 5th AutoTest Technical Conference on 'Test of Hardware and Software in Automotive Development'*, 15-16.10.2014, Stuttgart, Germany. pp. 145–152 (2014)
16. Skruch, P., Buchala, G.: *Model-based real-time testing of embedded automotive systems*. *SAE International Journal of Passenger Cars – Electronic and Electrical Systems* 17(2) (2014)
17. Skruch, P., Dlugosz, R., Kogut, K., Markiewicz, P., Sasin, D., Rozewicz, M.: *The simulation strategy and its realization in the development process of active safety and advanced driver assistance systems*. *SAE Technical Paper* 2015-01-1401 (2015)
18. Skruch, P., Panek, M., Kowalczyk, B.: *Model-based testing in embedded automotive systems*. In: *Zander-Nowicka, J., Schieferdecker, I., Mosterman, P. (eds.) Model-Based Testing for Embedded Systems*, pp. 293–308. CRC Press, Boca Raton, London, New York (2011)
19. Zander-Nowicka, J.: *Model-based testing of embedded systems in the automotive domain*. Ph.D. thesis, Technical University Berlin (2009)

A new current based slip controller for ABS

Krzysztof Kogut, Krzysztof Kołek, Maciej Rosół, and Andrzej Turnau

Institute of Automatics and Biomedical Engineering, Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering, AGH University of Science and Technology

Abstract. A laboratory Anti-Lock Brake System (ABS) is examined. The architecture of the ABS system is shown. The real-time experiments related to a control action to stabilize the slip at a certain level are taken into consideration. A new slip control algorithm differs from the previously used approaches. It is based on the measured current of the braking DC motor. The experimental results collected in the real-time for an old (relay) and new (current based) slip control are compared.

1 Introduction

All modern automotive vehicles are equipped with an anti-lock brake system. It is used to prevent the lockup of the wheels during the braking action. Avoiding the car wheel lock is crucial from a safety point of view. In general, when a wheel slip occurs (wheels stop because of the brakes) the car controllability is constrained and the braking distance is extended. This might happen on the snowy or wet surfaces. ABS optimizes braking effectiveness to keep wheels rolling on the road and reduce braking distance [1] [2]. Several algorithms have been developed in recent works. The most common approach is to use fuzzy-logic controllers, learning neural networks, sliding modes and genetic algorithms [3] [4] [5] [6]. However, even the most complex algorithmic method will not help if there is no measurement of the braking force applied to the wheel. In this study, a new controller based on measuring of the DC motor current responsible for the braking action is considered. The laboratory ABS system manufactured by the INTECO company, depicted in Fig. 1, has been equipped with the extra measurement. It is just the DC motor current proportional to the braking torque. The paper is organized as follows. In Sect. 2 the ABS device and its parameters are introduced. The slip control based on the measured current of the DC brake motor is described in Sect. 3. Experimental results are shown in Sect. 4. Final remarks are given in Sect. 5.

2 System Overview and Its Physical Parameters

The ABS laboratory system used in experiments consists of two rolling wheels: the car wheel and the car road wheel animating the relative road motion. The upper car wheel remains permanently in a rolling contact with the lower wheel.

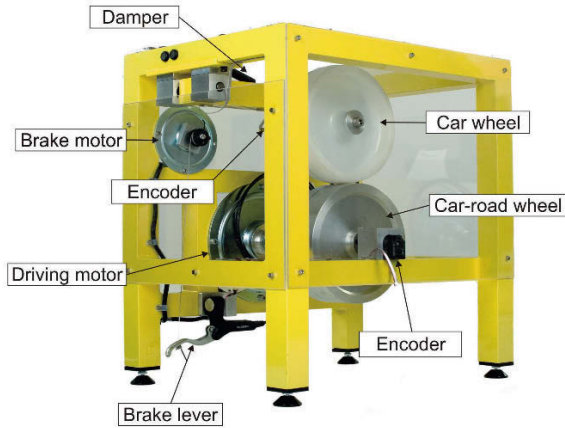


Fig. 1. Photography of the ABS (source: <http://www.inteco.com.pl/>)

The car road wheel has a smooth surface which can be covered by a given material to simulate a surface of the road. It is also used to accelerate both wheels to the desired initial angular velocity before the braking action begins. The car road wheel is driven by a powerful flat GPN12LR DC motor. GPN12LR is supplied with a voltage of +24 VDC and a maximum current equal to 11 A. The car wheel is rigidly connected to a disk brake system. This brake system is linked via hydraulic coupling to the brake lever which is driven by the small flat GPN9 DC motor by the tight side and tightening pulley. This DC motor is supplied with a voltage of +12 VDC and a maximum current equal to 6 A. Both DC motors are controlled with a Pulse-Width Modulation (PWM) signals with a frequency of 10 kHz to 20 kHz. To prevent unexpected vibrations, the car wheel has a damper attached to the rigid frame. The angular position of two wheels are measured by two identical incremental encoders of HEDM-5055 type. The encoders resolution is 4096 pulses per revolution (quadrature mode), giving the accuracy equal to 0.001534 rad. The corresponding angular velocities are reconstructed from these two positions using the simple Euler formula.

The architecture of a control system is shown in Fig. 2. The laboratory rig is directly connected to a power interface. It amplifies the PWM signals and separates them from a PC. Measurements of the DC motors current levels are taken with an integrated LEM CAS 6-NP (brake, the nominal current equal to 6 A) and LEM CAS 15-NP (driving, the nominal current equal to 16 A) current transducers, utilizing the Hall effect. An analog electronic circuit allows to condition output voltage signal from LEM by changing gain and offset.

All the examined digital and analog I/O signals from the power interface unit are connected to the RT-DAC/USB multipurpose I/O board. The whole logic necessary to activate and read the encoder signals, to generate PWM signals and

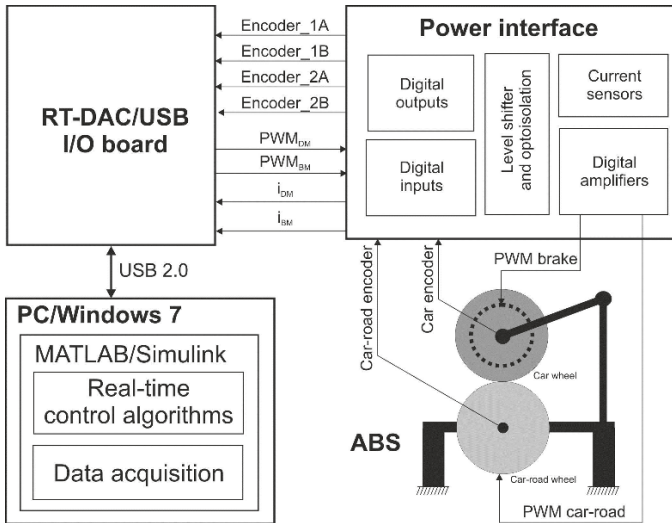


Fig. 2. Architecture of the control system

to handle ADC converter is configured in the Xilinx chip of the RT-DAC/USB board. The system sampling period is set to 0.01 s.

Real-time control algorithms and data acquisition are implemented in MATLAB/Simulink environment installed on PC. Rapid prototyping technique with automatic code generation is used. MS Windows 7 operating system is adapted to a soft-real time platform, using RT-CON software. RT-CON provides a real-time engine for executing Simulink models on MS Windows OS.

The process of obtaining a model of the laboratory ABS is presented in [7]. The grey-box method and data fitting technique are used in the identification process. The ABS brake torque is modelled with the first order differential equation with a brake control delay. The further research revealed that the braking torque control depending on the brake motor voltage is insufficient on account of brake force moment fluctuations and the brake current stabilization shall be applied. In fact, for the stabilization of the braking torque the measurement of the current flowing into the DC motor is required. The braking torque is proportional to the current. The controller is not based on the dynamical model from [7]. Therefore, the dynamical model is not used in this work.

3 The new Slip Controller Based on Brake DC Current Measurement

The new ABS slip controller is a combination of hybrid slip controller and brake DC motor current controller. First, the DC current measurement filtering is presented. Then the current controller is introduced and tuned. Finally, the obtained controller is connected to the output of the root slip controller.

3.1 The Brake DC Current Measurement Filtering

The Hall effect sensors tend to introduce noise to output voltage signal measured on USB RT-DAC board. Additionally, due to the fact that DC motor is controlled by PWM signal, the resulting current is affected by high frequency voltage changes. No filtering is applied to the sensor signal in the measurement board. The signal is filtered only in MATLAB/Simulink control and measurement model.

The measured sensor voltage signal obtained from the A/D converter is in the range $[0.375, 4.625]$ V and the value 2.5 V denotes 0 A current value. Before the voltage signal is converted into corresponding current signal the filter is applied. In order to smooth the data the moving-average filter is introduced. The window size is set to 10. The filtered voltage signal is utilized to calculate the current using linear transformation. The bias is set to 2.493 V measured by the A/D converted when the DC motor control is set to 0 V. The gain is set to 104.2 mV/A. It is based on the sensor technical specification.

Figure 3 presents the current sensor output filter response to the DC motor voltage control change. One can notice a noise contained in the raw signal and the filtering effect which enhances the current signal quality.

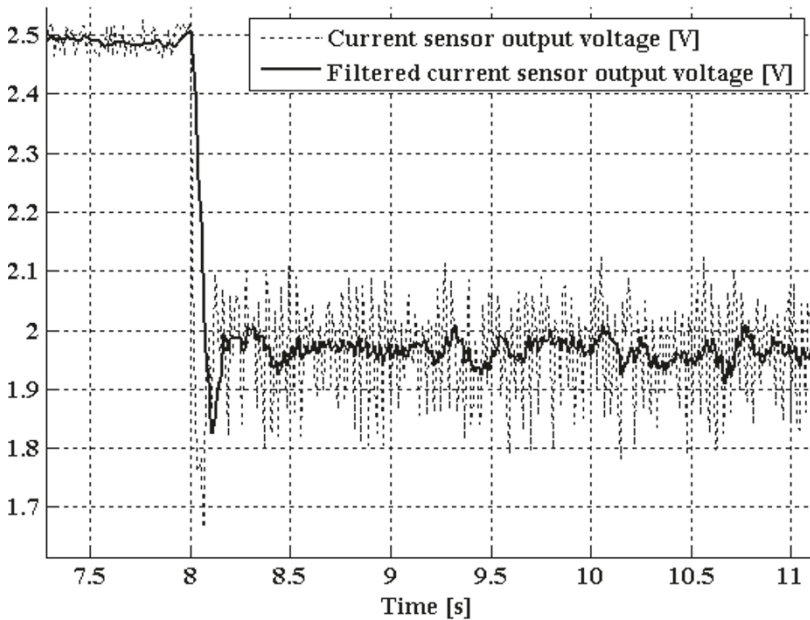


Fig. 3. The current sensor output filter performance

3.2 The Brake DC Current Controller

The DC motor current signal has a direct impact on the braking force moment acting on car wheel therefore it is a key requirement to control the current at the desired level during the braking maneuver. In order to realize such requirement the current controller must be employed.

In this work PI controller is designed and tuned for DC motor current control. The input to the controller consists of the current error calculated as the difference between the desired current (designated by the root slip controller see Sect. 3.3) and the actual current. The output is the DC motor PWM voltage control limited in the range [30, 50] %. The limitation of the controller output is applied in order to prevent high control values which cause rapid wheel velocity decrease which leads to wheel lock-up.

The tuning procedure of the proportional gain k_p^b and integral gain k_i^b was conducted using trial and error method. The goal was to find parameters which ensure fast reaching of the desired current and accurate tracking of the set point. The obtained parameters are $k_p^b = 0.1$ and $k_i^b = 0.5$.

The controller performance is depicted in Fig. 4. The slope of the desired current is driving the controller. The upper plot shows the actual and desired current and the lower plot presents the PI controller output signal. The DC motor current is properly tracking the monotonically increasing reference signal.

3.3 The Slip Controller Based on Current Control

The objective of the ABS is to keep the wheel slip at the required level during braking. In the examined system the control of a slip is performed by steering the DC motor PWM voltage. Originally, when there is no DC motor current measurement, the control strategy might only be based on wheels velocities. With the addition of the mentioned feedback signal from the brake the control algorithms gain extra input which might be utilized to improve control laws and their performance.

For the purpose of this work a daisy chain of system controllers was prepared. The block diagram of the hybrid controller is presented in Fig. 5. The PI current controller examined in previous Sect. 3.2 is governed by the slip root controller. The role of the master controller (slip root controller) is to provide the desired DC motor current to the slave controller (current controller) in such a way that the desired slip ratio is maintained.

The slip controller is a heuristic controller realized as a combination of PD controller and two-states controller. The control law is as follows:

$$i_d = \begin{cases} i_d^{\max}, & : \lambda_e \geq 0.1 \\ \text{PD}(\lambda_e), & : |\lambda_e| < 0.1 \\ i_d^{\min}, & : \lambda_e \leq -0.1 \end{cases} \quad (1)$$

where i_d and λ_e denote respectively the desired current of the brake DC motor and slip error values. The limits of the current (i_d^{\max} and i_d^{\min}) are set to 9 A and 6 A. The lower limit corresponds to the end of brake dead zone (current level

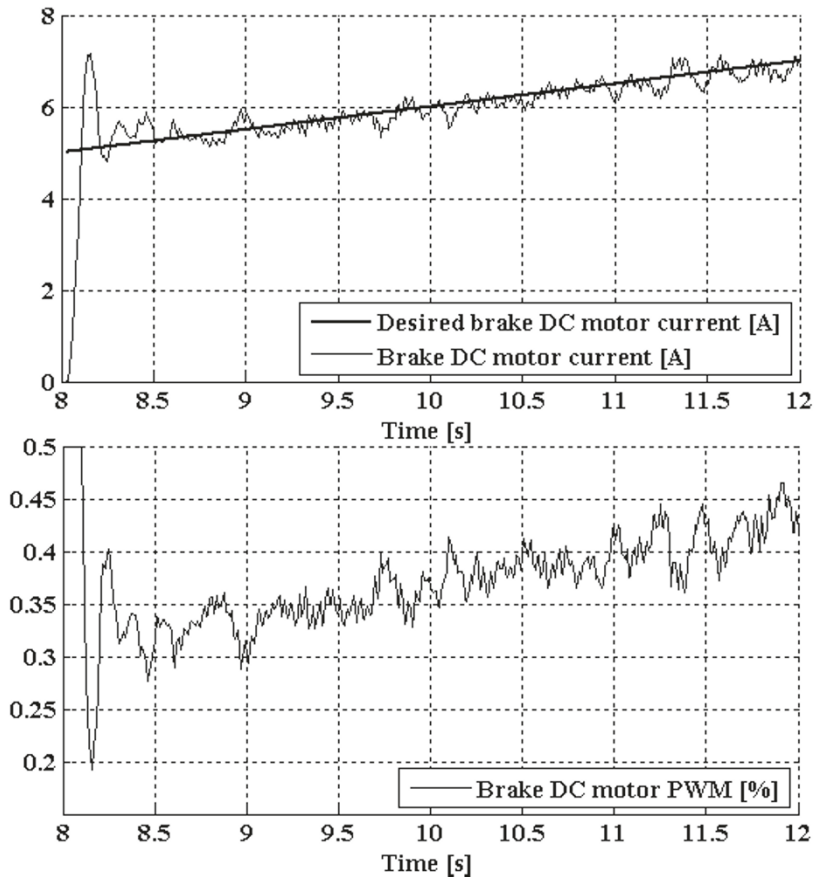


Fig. 4. The brake DC motor current controller performance

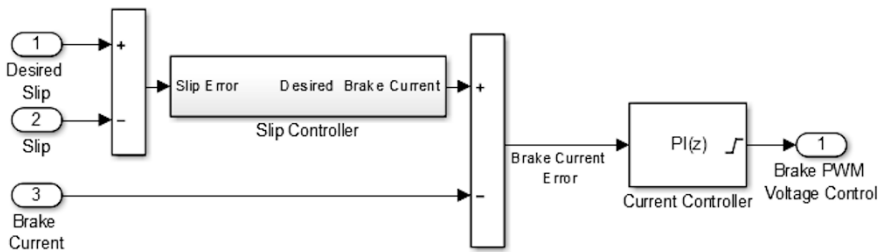


Fig. 5. The new slip controller block diagram

the application of which does not generate braking force). The PD controllers tunable parameters were obtained based on an experimental observation and the trial and error approach. The proportional gain is set to 20 and derivative gain is set to 1.

4 Experimental Results

The new ABS slip controller performance is compared with classic two-states relay controller. This controllers input is also λ_e although the control variable is the brake DC motor PWM voltage u_b . The control law is as follows:

$$u_b = \begin{cases} u_b^{\max}, & : \lambda_e \geq 0 \\ u_b^{\min}, & : \lambda_e < 0 \end{cases} \quad (2)$$

The limits of PWM voltage (u_b^{\max} and u_b^{\min}) are set to 30 and 50 [%] in order to be compliant with the limits applied to the PI current controller.

The performance assessment of the new ABS slip controller is presented in Fig. 6 and the behavior of relay controller is depicted in Fig. 7. The desired slip ratio is set to 0.5 in both cases. The plots time range is limited to a span of approximately 1s. During the time before the depicted period the wheels are accelerated to the speed of about 2200 RPM. When this happens the controllers are enabled and the braking maneuver begins (the first jumps of control denote controllers start). After the presented one second period, the wheels speed decreases to the level when slip stabilization is not applicable. One can notice that during the control period the new ABS slip controller outperforms the classic one in the desired slip tracking (the maintained slip is closer to the reference and the oscillations are smaller).

5 Conclusions

No information about the braking force in the ABS during its operation certainly was an astonishment to the authors. Therefore this additional measurement signal based on the current supplied to the braking DC motor which is now installed in the ABS, facilitates, and even makes it possible to stabilize the slip in the car. A reconstruction of the physical quantities which cannot be measured is a challenging task for the engineer. Less ambitious but recommendable is to benefit from the new measurement.

References

1. Saveresi, S.M., Tanelli, M.: Active Braking Control Systems Design for Vehicles. Springer-Verlag London Limited (2010)
2. Kiencke, U., Nielsen, L.: Automotive Control Systems for Engine, Driveline, and Vehicle. Springer-Verlag Berlin Heidelberg (2005)

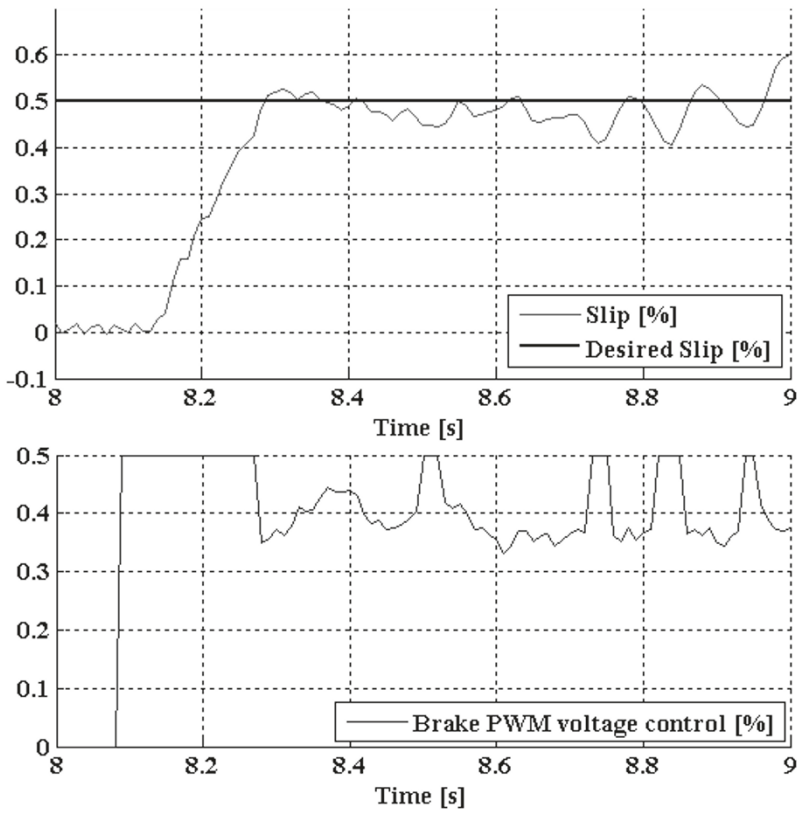


Fig. 6. The performance of the new ABS slip controller

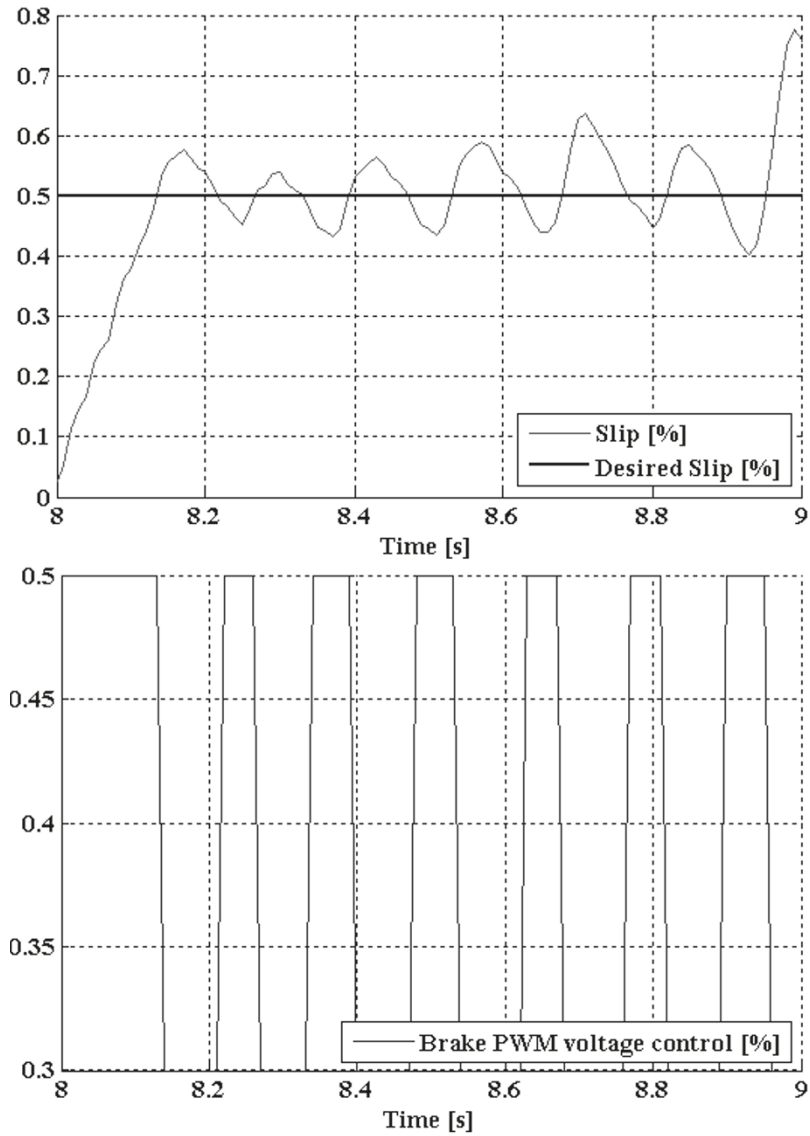


Fig. 7. The performance of the relay slip controller

3. Precup, R.E., Spataru, S.V., Radac, M.B., Petriu, E.M., Preitl, S., Dragos, C.A., David, R.C.: Experimental Results of Model-Based Fuzzy Control Solutions for a Laboratory Antilock Braking System Hippe, Z.S., et al. (Eds.): Human Computer Systems Interaction: Backgrounds and Applications 2: Part 2, pp. 223-234, Springer-Verlag Berlin Heidelberg (2012)
4. Sharkawy, A.B.: Genetic fuzzy self-tuning PID controllers for antilock braking systems Engineering Applications of Artificial Intelligence 23, pp. 1041-1052 (2010)
5. Kayacan, E., Oniz, Y., Kaynak, O.: A Grey System Modeling Approach for Sliding-Mode Control of Antilock Braking System IEEE Transactions On Industrial Electronics, Vol. 56, No. 8 (2009)
6. Poursamad, A.: Adaptive feedback linearization control of antilock braking systems using neural networks Mechatronics 19, pp. 767-773 (2009)
7. Kogut, K.: Anti-lock braking system modelling and parameters identification 19th International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 342-346 (2014)

System for tracking multiple trains on a test railway track

Zdzisław Kowalczyk and Sylwester Frączek

Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Narutowicza 11/12, 80-233 Gdańsk, Poland
kova@pg.gda.pl, s.fraczek@gmail.com

Abstract. Several problems may arise when multiple trains are to be tracked using two IP camera streams. In this work, real-life conditions are simulated using a railway track model based on the Pomeranian Metropolitan Railway (PKM). Application of automatic clustering of optical flow is investigated. A complete tracking solution is developed using background subtraction, blob analysis, Kalman filtering, and a Hungarian algorithm. In total, six morphological, convolutional and non-linear filtering methods are compared in sixty-three combinations. Accuracy and performance of the system are evaluated, and the obtained results are analysed and commented on.

Keywords: Computer vision, Cameras, Object tracking, Signal processing, Motion detection, OpenCV

1 Introduction

Although the problem of object tracking in video streams has been approached by many in the past three decades, it continues to be a great challenge. Recent increase in affordable computational power and active development of open source frameworks make tackling the problem increasingly more appealing to companies as well as individual engineers.

As research in machine learning is currently sky-rocketing, its contribution to image and video analysis is immense. Object tracking can use trained detectors instead of hand-engineering new features or heuristic methods for detection of objects. Learning algorithms are fed with hundreds of thousands of samples and find the best features on their own. Machine learning has already surpassed classical methods in accuracies at object detection task but the problem of tracking is still relatively fresh. Using detectors leads to great results [1,12,11,10] and ConvNets have a potential for use in end-to-end tracking solutions [9]. Detectors require a lot of data for either tracking or detection even when only fine-tuning a network pre-trained for general image classification. Data acquisition, labelling, developing models and training of detectors by oneself can be time consuming even with the state-of-the-art hardware.

To the best of our knowledge, there is no publicly available research on any visual systems for train tracking. In [2] and [8] authors rely on background subtraction, discard detections that do not last for a certain minimum number of frames and use Mixture of Gaussians (MOG)-like background subtraction. The first method uses window correlation score and checks if motion superimposes with the detections, while the second one uses a Hungarian algorithm for matching and a Kalman filter for trajectory refinement. It is impossible to tell which method achieves higher accuracy based on the provided information.

In this paper, having no dataset for train detection, we propose to use a pure motion analysis approach. We review an attempt to clustering of similar motion regions and present a motion-based Pomeranian Metropolitan Railway (PKM) train tracking algorithm built up on [8]. The proposed system simultaneously tracks all visible trains in the Field Of View (FOV) of multiple independent Internet Protocol (IP) cameras as they move on a testing track (Fig. 1). It projects the position, ID and velocities of all trains detected and tracked on the image of the railway track on-line. The purpose is to deliver localization information about trains on the PKM railway testing track model for indoor use. The system filtering out objects which are not trains, is resistant to changes in lighting conditions and other possible sources of noise such as camera grain.

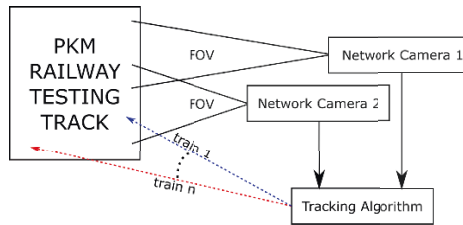


Fig. 1. General architecture of the tracking system for PKM railway [6].

This paper is organised as follows. Sect. 2 explains streaming and merging of frames. Sect. 3 reviews the approach of similar motion regions clustering. Sect. 4 describes background subtraction. In Sect. 5 the final performance of this tracking solution is described on a set of representative situations, highlighting the advantages and disadvantages of this specific implementation. Finally, we summarize our work in Sect. 6.

2 Streaming

Multiple IP camera feeds are streamed using a LibVLC media framework in separate threads. Streams can be manipulated (position and scaling) on the screen at any time. The detection and drawing is performed in the main thread. Received frames are updated to a single frame whenever a new update comes. If a stream is slower, its last frame is reused.

We first merge frames, then we apply processing. Otherwise, the edges of frames would require special attention when overlapping or adjoining. Most morphological operations and filtering based on blob dimensions may produce different results for both approaches as shown in Fig. 2. If the minimum preserved blob size is 7 or bigger, half of the motion information is lost, because the top frame contains only 6 pixels.

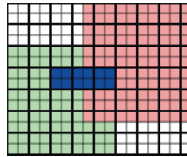


Fig. 2. Train (blue) traversing frames (red and green).

Calibration [3,14] was unnecessary with distortions within tolerance. Composition of two IP camera streams on the screen joins smoothly, resulting in a barely visible contact line where the overlapping streams meet.

3 Clustering of motion vectors

The proposed idea for a detection algorithm was to calculate optical flow on the input image and cluster the output using automatic k-means and a Calinski-Hrabsz criterion for evaluating the optimal number of clusters. Clustering in a Cartesian representation was hard, but trivial for polar coordinates, as shown in Fig. 3. Although trains are close to each other, their speed vectors are opposite and have different magnitudes.

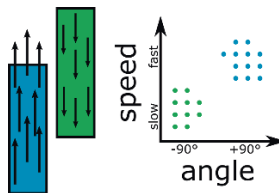


Fig. 3. Simplified model in the expected parameter space of the flow data.

Feasibility was verified in MATLAB simulation. Each circle in Figs. 4a and 4b has a random colour and represents spatial coordinates of a pixel. Randomly coloured dotted lines represent motion vectors of pixels to which they are attached. The colours of the rectangles represent clusters to which they were automatically assigned.

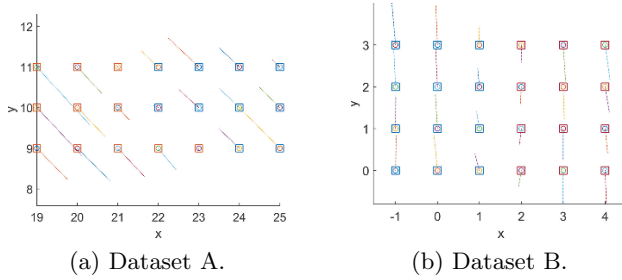


Fig. 4. Two exemplary datasets (A and B) and their clustering.

Motion vectors were additionally processed with a Gaussian filter. Datasets drawn in all orthogonal projections and coloured according to cluster membership, are shown in Fig. 5. A 4D plot is shown in Fig. 6. The lengths of motion vectors are in a feasible range for speed of SA136 series diesel multiple units owned by the PKM joint-stock company [7]. On the screen of a laptop at comfortable viewing settings the length of translation vector $\|v\| = 2\sqrt{2}$ px per frame at 30 frames per second (fps) represents roughly speed of $s = 2\sqrt{2} \frac{\text{px}}{\text{frames}} \times 30 \frac{\text{frames}}{\text{s}} \times \frac{2.883}{16} \frac{\text{m}}{\text{px}} \approx 55 \frac{\text{km}}{\text{h}}$ in the real world.

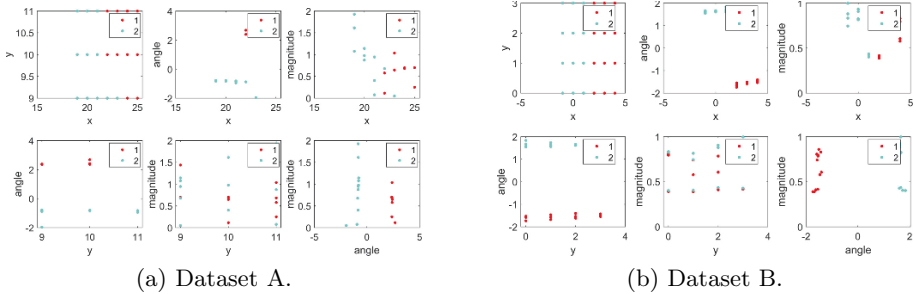


Fig. 5. 2D projections of the data.

The performed simulations have shown quite promising results. The algorithm has correctly found 2 clusters with the upper limit of 4 set. The encouraging outcome motivates further tests on real data.

3.1 Clustering of motion vectors on real data

With the Farneback’s [4] dense optical flow from OpenCV applied, the performance of the algorithm has dropped to 4.53 fps.

As illustrated in Figs. 7a to 8b, motion is detected only on small patches of the trains. Dots are the points for which the optical flow is drawn and lines

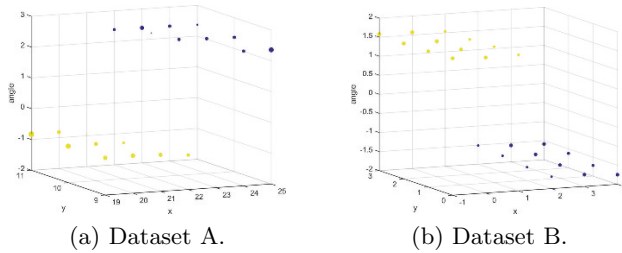


Fig. 6. Clustering in 4D space, where size represents magnitude.

represent their motion vectors. In Figs. 7b and 7c vector magnitudes are amplified by the factor of 5 to emphasize on noise in the surroundings, which might be caused by diffused reflections and shadows on the surface. In Figs. 8a and 8b colours represent magnitude and angle according to the HSV colour space, where H is the angle, while S and V are both 200 times the magnitude saturated at 255. Colouring indicates how the motion directions span across the trains. The red and the aquamarine, for instance, correspond to opposite directions of motion (on the HSV colour-wheel), and yet are very close to each other in the image.

This algorithm has poor performance in our environment both in the speed and quality of detection. Slow processing leaves no room for clustering and tracking on-line. With this quality of motion detection, it is not likely that clustering could effectively work. As the results were unsatisfactory, background subtraction has been considered.

4 Background subtraction based multiple train tracking

The second approach relied on background subtraction and blob analysis. Comparisons available in the Internet claim superiority of MOG, which even supports shadow removal, over similar methods provided by the OpenCV library. We have confirmed the results as well.

Trains were frequently split into smaller blobs. In order to refine the foreground mask, a number of filters were tested. Tile-based discretisation of size 5×5 will be called *discretisation* for simplicity. In this algorithm, each tile is filled with either white or black colour depending on the count of white pixels inside of it; according to [5], where it was used to filter out all the small noisy pixels. If there are at least N non-black pixels in each square, the colour is white, and if there are less, it is black.

Sixty-three combinations of 6 filters were tested and the results are presented in Fig. 9 (all of the possible combinations, but not all permutations). Empty label means no filtering. The median and Gaussian filters were the only ones applied to the colour image before background subtraction. The remaining filters were applied after thresholding. The execution frame-rate of the main thread is presented in Tab. 1.

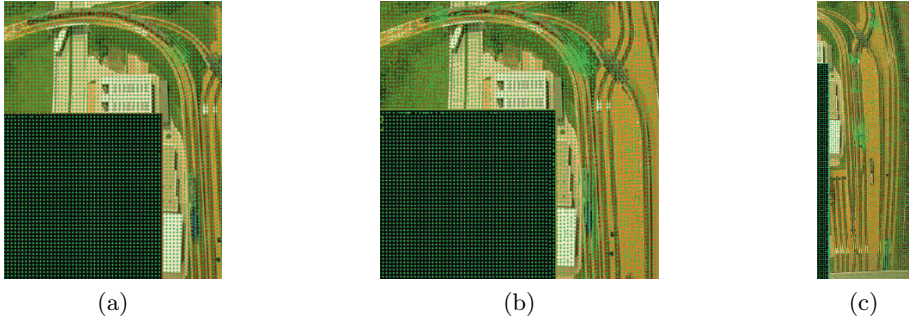


Fig. 7. Optical flow, where exemplary pictures (b) and (c) were five times amplified.

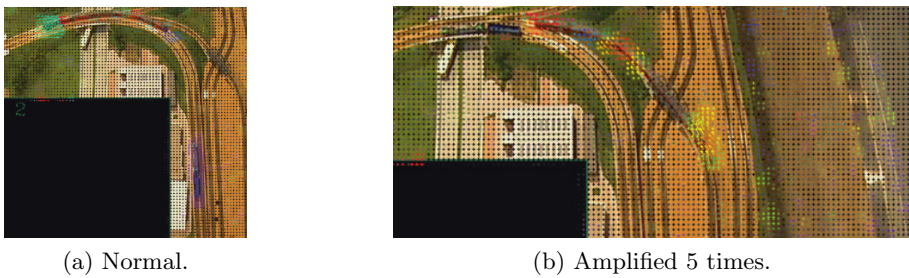


Fig. 8. Optical flow, where angle is represented by colour and magnitude by intensity.

Table 1: Filter combinations and their average frame-rate.

filter	fps	filter	fps	filter	fps	filter	fps
original	31.66	GS	21.52	MG	20.4	ML	21.13
B	29.93	GSB	21.15	MGB	20.58	MLB	21.03
C	28.93	GSC	21.15	MGC	20.74	MLC	20.98
CB	30.91	GSCB	22.25	MGCB	22.99	MLCB	21.11
G	27.39	L	30.33	MGL	20.79	MLS	19.27
GB	28.95	LB	28.98	MGLB	20.23	MLSB	19.54
GC	24.74	LC	27.88	MGLC	20.27	MLSC	19.48
GCB	27.57	LCB	28.12	MGLCB	21.13	MLSCB	19.86
GL	26.61	LS	22.95	MGLS	17.06	MS	19.29
GLB	25.95	LSB	24.23	MGLSB	18.49	MSB	19.95
GLC	23.81	LSC	22.45	MGLSC	17.51	MSC	19.85
GLCB	27.61	LSCB	22.05	MGLSCB	20.95	MSCB	20.66
GLS	21.41	M	21.14	MGS	18.24	S	25.57
GLSB	21.11	MB	20.98	MGSB	18.15	SB	24.64
GLSC	26.26	MC	21.34	MGSC	17.5	SC	23.27
GLSCB	21.49	MCB	21.34	MGSCB	18.36	SCB	23.72



Fig. 9. Original, B&W and 63 combinations of filters (5×5 kernel) applied in the following order: G – Gaussian filter, M – median filter, C – morphological closing, L – morphological dilation, S – discretisation (8 px threshold), B – logical sum with previous frame.

The logical sum with the previous frame (B) has only effect if in between consecutive main thread updates all of the streams were updated. Small differences can be seen in the case of MGLS vs. MGLSB, where the little blob on the bottom of the longer train was joined with the body of the train. Also in MLSC and MLSCB the split of a smaller train's blob was prevented. In the case of S vs. SB there is no result as the proceeding frame was identical, and in the C vs. CV the sum is actually worse, which need not to be deterministically-based. The frame-rate drop is above 5%.

The morphological closing (C) operation deals well with joining sparse pixels into a single blob. It is visible in the simplest case (no filter versus C) and in other cases as well. In the cases of MGS vs. MGSC, M vs. MC and G vs. GC improvement is apparent. The drop in the frame rate is almost 9%.

The morphological dilation (L) leaves no disconnected parts of a longer train, although it can also connect them with the shadow to the left. Even small trains can be well segmented. It looks more robust as compared to the closing operation, which has barely preserved the connection at the bottom of a longer train. The frame-rate drop caused by dilation is approximately only 4%.

The discretisation (S) filter makes blobs better defined, and is good for noise removal if not done in-place. In the L vs. LS dilated noise has amplified (as it exceeded a threshold). Shadow was merged with the blob. Our simple suboptimal implementation resulted in a significant drop, above 19%, in the frame rate.

Gaussian (G) filter shows no improvement over clean mask. In C vs. GC, the effect is good as longer train is thicker. In S vs. GS, the effect is positive. In combination with dilation (L) noise reduction improves. In total, the effect of the filter seems positive but at the expense of around 13% frame-rate drop.

The median filter (M) smooths out the area inside a longer train as compared to a clean mask, but at the same time, it can split it near the top. The bottom of a longer train is jagged, and a smaller train is split. In the C vs. MC, the influence of shadow is reduced with the median filter, but the long train is also shorter at the bottom. The small train without the median filter is split into 3 instead of 2 blobs. The median filter has negative influence on background subtraction and it also suffers from the substantial drop of the frame-rate, resulting in the most drastic loss of more than 33%.

To summarize, the median filter is definitely not improving the segmentation quality and has a high computational cost. The Gaussian filter improves the filtering results on average, but not always. The logical sum with previous frame is fast, but too often the frames are the same. The discretisation is effective but slow, and is not a competitor to the dilation or closing. The best two were the dilation and closing operations. Being computationally cheap, they also outperform others. Since the shadow is not interesting, only the closing operation was selected.

4.1 Blob analysis, detection matching and tracking

With a filtered mask, blobs with its area a in the range of $50 < a < 320 \times 240$ square pixels were found. The range was chosen empirically and the remain-

ing blobs were ignored. The contours of those areas were stored in an array as hypothesized detections of trains. Their centroids and lengths were extracted. These parameters were later used to match the current detections with the ones already being tracked. Tracking was done in two steps. First, the Hungarian algorithm was used with a quadratic distance metric:

$$d_{i,j} = (x_i - x'_j)^2 + (y_i - y'_j)^2 + (l_i - l'_j)^2 \quad (1)$$

where i is the index of a train, j is the index of a blob, $d_{i,j}$ is the distance value between them, (x_i, y_i) are the train's coordinates, (x'_j, y'_j) are the blob's coordinates, l_i is the train's length and l'_j is the blob's length. The applied vision system uses the following approximation:

$$d_{i,j} \approx (\overline{TPC}_i - \overline{BC}_j)^2 + (TL_i - BL_j)^2 \quad (2)$$

where \overline{TPC}_i is the centroid of the i -th train predicted by a Kalman filter, TL_i is its minimum area bounding rectangle's length, \overline{BC}_j is j -th blob's centroid and BL_j is its minimum area bounding rectangle's length. If the Hungarian algorithm [13] finds the best matches and the value of distance is less than empirically selected threshold $\tau = 400$, then it is assumed that the hypothesis of a blob being a train is correct and the model is updated with its coordinates. A Kalman filter is applied to each of the *tracks* to refine its trajectory. Next, a prediction of their next state is updated, and the cycle repeats.

Trains were modelled with a set of two second order linear differential equations of Newtonian dynamics. Let p_x and p_y be the coordinates, $v_x = \dot{p}_x$ and $v_y = \dot{p}_y$ be the corresponding velocities, $a_x = \ddot{p}_x$ and $a_y = \ddot{p}_y$ be the corresponding accelerations. Then

$$p_x(t) = p_x(0) + V_x(t)t + \frac{a_x t^2}{2} \quad (3)$$

$$p_y(t) = p_y(0) + V_y(t)t + \frac{a_y t^2}{2} \quad (4)$$

which in state-space representation can be written as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (5)$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad (6)$$

where the state and control vectors are

$$\mathbf{x} = [p_x \ p_y \ v_x \ v_y \ a_x \ a_y]^\top \quad \mathbf{u} = [0 \ 0 \ 0 \ 0 \ 0]^\top$$

along with the following matrices:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & t & 0 & \frac{t^2}{2} & 0 \\ 0 & 1 & 0 & t & 0 & \frac{t^2}{2} \\ 0 & 0 & 1 & 0 & t & 0 \\ 0 & 0 & 0 & 1 & 0 & t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

As the measured state variables of the trains represent the screen coordinates, the output vector is $\mathbf{y} = [p_x \ p_y]^\top$. Since there is no input to the system (\mathbf{u} is zero), both the input matrix B and the feed-through matrix D can be omitted. For the sampling rate $t = 1$ we fix $A(t = 1)$, the measurement matrix

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and the process noise covariance and the measurement noise covariance matrices as $\mathbf{Q} = 0.01\mathbf{I}_6$ and $\mathbf{R} = 10\mathbf{I}_2$. The process is thus expected to be trusted more than the measurement of the centre of the train. The noise of measurement is assumed to have the variance of $\sigma^2 = 10$ as opposed to the variance of the process noise, which is $\sigma^2 = 0.01$ for all the state variables. It is assumed that all the noisy signals in the process are orthogonal. The initial state vector assigned during the first appearance of a train is $\mathbf{x}(\mathbf{0}) = [p_x(0) \ p_y(0) \ 0 \ 0 \ 0 \ 0]^\top$, where $p_x(0)$ and $p_y(0)$ are the coordinates of the newly discovered blob's centroid. It is safe to initialize the velocity and acceleration to zero rather than to a value that might have the opposite signs.

Despite the efforts made, the system is not robust to occlusion. When a train approaches or comes out of the other side of a bridge, the contours change their sizes accordingly. The Kalman filter does not deal with this problem satisfactorily. Since an attempt to counter this by replacing the train's length with an Exponentially Weighted Moving Average (EWMA) of the length failed (providing worse overall accuracy), the algorithm has been reverted back to the simple assignment approach.

To stop generating many random train labels on the image, a *track* disposal condition from [8] has been augmented by additional conditions. The specifics of our trains was that they are significantly long in one dimension and do not move fast. Thus a train is deleted when any of the criterions used in [8] or the following four conditions is met:

- $age < 8$ and $\frac{totalVisibleCount}{age} < 0.6$
- $ratio < 2$
- $speed > 16$
- $consecutiveInvisibleCount \geq 16$

where *ratio* is the ratio of the minimum area rectangle's dimensions, *speed* is the average translation from up to 10 last iterations. A simplified diagram of the algorithm in its final state is shown in Fig. 10.

5 Tracking tests

Trains are assigned IDs based on their detection order. Velocity is measured in pixels per frame. The colours of the text and path are the same. A path is drawn by the centroid of the shape at the predicted position in each frame. The black irregular shapes are the contours of motion blobs that have passed through

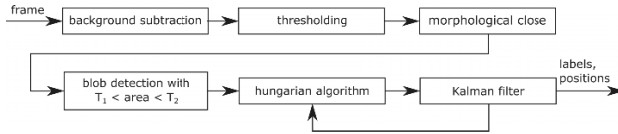


Fig. 10. Simplified block diagram of the final algorithm.

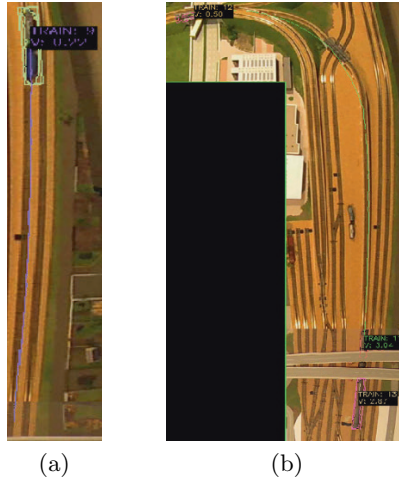


Fig. 11. Examples of successful tracking of short (a) and long (b) trains.

the filters. They are not yet classified as trains. One of those contours can be seen in Fig. 13b. The rectangle represents the last detected minimum area bounding rectangle of the contour. The last of the shapes is train's contour centred at the predicted position of the centroid. Colour of the contour is the same as that of rectangle. The fact that the contour is centred at the predicted position, while the rectangle stays at the last seen position allows us to see where the detections are lost but are kept being predicted.

The system performs well (Fig. 11) as long as the models are in motion, not occluded, and background is mostly distinguishable from trains. The paths travelled in Figs. 11a and 11b are consistent for both fast and slow trains. Lack of determinism can be observed by different label assignment in Figs. 12a and 12b due to poor synchronization between the threads.

The examples in Fig. 13 show lowlights of the algorithm. In Fig. 13c the longer train was last seen inside the white rectangle and the predicted position of the train is at the centre of the white contour. The small train to the right was tracked until it started vanishing under the bridge. A false positive appears close to the bridge. In Fig. 13b the train on the left is about to stop its downward motion while it is no longer seen as moving, but the background around its bottom end was detected. Looking at the foreground mask image, it became clear that the pixels occupied by the train are scattered, forming smaller blobs

that could not pass through the filters. This means that the modelled background around that spot was hardly distinguishable from train. Our algorithm does not handle stationary trains such as the one on the right. The problem of merging trains passing each other can be seen in Fig. 13a. The train on the right was successfully tracked for a number of frames going upwards, then the *track* is being intercepted by the train going downwards. Beside the successful part, there are also effects of occlusion in Fig. 11b. Successful green detection fades out right before the bridge and the new one emerges on the other side. Fig. 13d indicates the problem of adaptive background modelling. The MOG model has adapted to the colours of pixels of the train before it has started to move. The space which was previously occupied, started to differ from what was learned; and it was classified as a train. The smaller train approaching on the left merged together with the false positive into a bigger blob.

The system was tested with two 1280×1024 10 fps video streams from USB 3.0 HDD, on a laptop running Windows 10 Pro 64-bit with Intel Core i7-4720HQ @ 2.60 GHz, 8,0 GB RAM. The average frame-rate is approximately 32.78 fps with the lowest inter frame delay settings. The system required approximately 28 MB of RAM and engaged less than 25% of the CPU usage (Fig. 14).

6 Summary

A vision system for on-line tracking of PKM trains on a railway testing track was developed. It labels and tracks detected trains on multiple video streams such as IP cameras. The user interface provides a comfortable way of stream arrangement. Several ways of improving the accuracy were mostly unsuccessful. Accuracy is far from the state of the art, but it does well under favourable conditions.

Although the approach of clustering of similar motion regions using the Calinski-Hrabsz criterion seemed promising in simulation, it has turned out

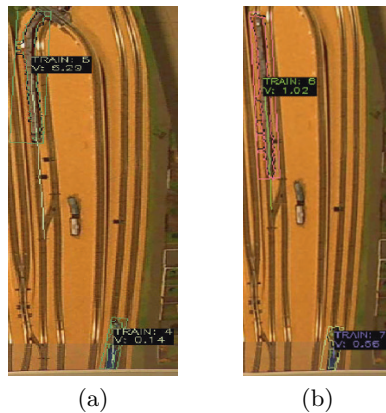


Fig. 12. Non-determinism. Trains labelled 5 and 4 (a) seen as 6 and 7 (b).

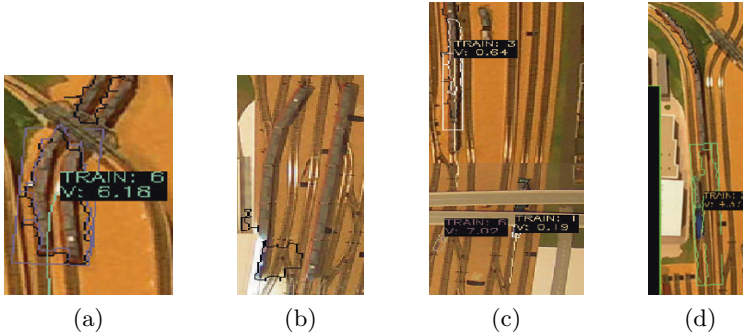


Fig. 13. Examples of tracking problems.

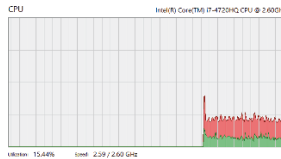


Fig. 14. CPU utilization over time, from launch to exit.

unreliable with the actual video streams due to the insufficient quality of optical flow estimation. After several tests on the actual camera streams, poor results led to a dead end. A better established approach based on the sample code from MathWorks tutorial was used to complete the solution. By a set of tests of filtering methods of the output foreground mask image we have found that morphological closing had the best effect on the correctness of blob segmentation. The system does not handle occlusion well, what could not be accomplished by detection of the change of train's size and pattern of motion on the other side of an obstacle.

The system could be improved by taking the actual frame rate into the Kalman filter (instead of $t = 1$), which could improve the accuracy of predictions. Correction of colour, or other pre-processing might also improve the background detection accuracy in places where the trains are confused with background. Overall performance could be improved by adapting a ConvNet for either detection or semantic image segmentation. In this way the stationary trains could be detected or segmentation be made more accurate.

References

1. Andriyenko, A., Schindler, K.: Multi-target tracking by continuous energy minimization. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. pp. 1265–1272. IEEE (2011)
2. Baldini, G., Campadelli, P., Cozzi, D., Lanzarotti, R.: A simple and robust method for moving target tracking. In: Proceedings of the International Conference Signal Processing Pattern Recognition and Applications (SPPRA2002) (2002)

3. Burrus, N.: Kinect calibration (May 2014), available at <http://nicolas.burrus.name/index.php/Research/KinectCalibration> (Last retrieved 20-02-2016)
4. Farneback, G.: Two-frame motion estimation based on polynomial expansion. In: 13th Scandinavian Conference on Image Analysis. pp. 363–370. Springer (2003)
5. Frączek, S.: Problem sterowania manipulatorem robota z wykorzystaniem obrazu z kamery w celu rozwiązania problemu teorii grafów na tablicy – Optymalizacyjny System Wycierania Tablicy. (Bsc Thesis no. PG/WETI/KSD/ZK182i/12/2013, Supervisor: prof. Z. Kowalczyk), Faculty of ETI, Gdańsk University of Technology, Gdańsk, Poland (2016)
6. Frączek, S.: Vision system for PKM railway testing track. (MSc Thesis no. PG/WETI/KSDiR/ZK238m/09/2016, Supervisor: prof. Z. Kowalczyk), Faculty of ETI, Gdańsk University of Technology, Gdańsk, Poland (2016)
7. Jędrzejewski, B.: Spalinowe zespoły trakcyjne serii SA136. Świat Kolei 7, 12–17 (2011)
8. Motion-based multiple object tracking (publication date unknown), available at <https://www.mathworks.com/help/vision/examples/motion-based-multiple-object-tracking.html> (Last retrieved 20-02-2016)
9. Ondruska, P., Posner, I.: Deep tracking: Seeing beyond seeing using recurrent neural networks. The 13th AAAI Conference on Artificial Intelligence (2016)
10. Face detection using haar cascades (publication date unknown), available at http://docs.opencv.org/trunk/doc/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html (Last retrieved 20-02-2016)
11. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. vol. 1, pp. I–511. IEEE (2001)
12. Xiang, Y., Alahi, A., Savarese, S.: Learning to track: Online multi-object tracking by decision making. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4705–4713 (2015)
13. Yiming, S.: Kuhn-munkres (hungarian) algorithm opencv implementation (2014), available at <https://github.com/soimy/munkres-opencv> (Last retrieved 20-02-2016)
14. Zhang, Z.: A flexible new technique for camera calibration. Transactions on Pattern Analysis and Machine Intelligence 22(11), 1330–1334 (2000)

The clipped LQ control oriented on driving safety of a half-car model with magnetorheological dampers

Jerzy Kasprzyk ^{*}, Piotr Krauze and Janusz Wyrwał

Institute of Automatic Control,
Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland

Abstract. The problem of improving driving safety for vehicles equipped with magnetorheological (MR) dampers is considered. It is proposed to control the MR dampers using the clipped LQ (Linear Quadratic) control which can be regarded as the two-dimensional Skyhook algorithm. The strategy is applied to a half-car model with four degrees of freedom, oriented on roll dynamics. Here, control algorithm optimised with respect to minimisation of roll vibrations is considered. Simulation experiments were performed assuming the model is subjected to impulse excitation generated as a torque applied in the centre of gravity, that is equivalent to a manoeuvre in the form of a sudden turning. The quality of the algorithm was validated using the RMS-based performance index and the results confirm the effectiveness of the proposed solution for the semi-active suspension.

Magnetorheological damper, driving safety, Skyhook control, clipped LQ control, half-car model

1 Introduction

The primary aim of the vehicle suspension is to isolate the vehicle body from uncomfortable vibrations generated by road roughness and transmitted through the tires. Concurrently, the control strategy should be designed in a way which provides driving safety. However, improving ride comfort usually deteriorates driving safety while the latter requires more attention. Different safety issues can be distinguished including road holding and ride handling. The road holding factor can commonly be improved with the cost of ride comfort, and inversely, improving the ride comfort deteriorates traction of the vehicle to the road surface resulting in a significant decrease in driving safety [6, 14].

Modelling and simulation of vehicle dynamics, especially in the field of driving safety, has become an important area of research in the recent years [8, 17, 19]. In general, roll and pitch motions of the vehicle body are caused by two types

^{*} Corresponding author. Email: jerzy.kasprzyk@polsl.pl. Tel.: +48-32-237-1046. Fax: +48-32-237-2127.

of disturbances that can occur while driving: kinematic excitation due to road irregularities and inertia forces caused by longitudinal and lateral vehicle accelerations. Presented studies are focused on cornering manoeuvres that are related to roll, as it is more important for safety problem. It is assumed that this body motion is caused by an inertial load acting directly on the body centre of gravity that causes torque roll with respect to the longitudinal axis of the body.

Numerous control strategies related to improvement of driving safety have been studied in the literature, e.g. PID control [13], Groundhook [7], H_∞ control [4], predictive control [1], fuzzy control [3], active anti-roll control [12] and others. In this paper the clipped LQ (Linear Quadratic) control which can be regarded as the two-dimensional Skyhook algorithm is considered in order to improve driving safety.

The paper is organized as follows. First, half-car model used to analyse the effect of rolling is derived. Subsequently, models of magnetorheological (MR) dampers are discussed and details related to the applied control strategy are described. Then, the results of the system performance for roll motion of the vehicle body are presented and concluding remarks are formulated.

2 Half-car model of roll dynamics

Rotation about the longitudinal axis of the vehicle (x -axis) is called *roll* and it will be denoted by angle φ , whereas rotation about the lateral axis (y -axis) is referred to as *pitch*. Fig. 1 presents the body diagram related to roll. It is assumed that the centre of gravity is located above the centre of roll motion.

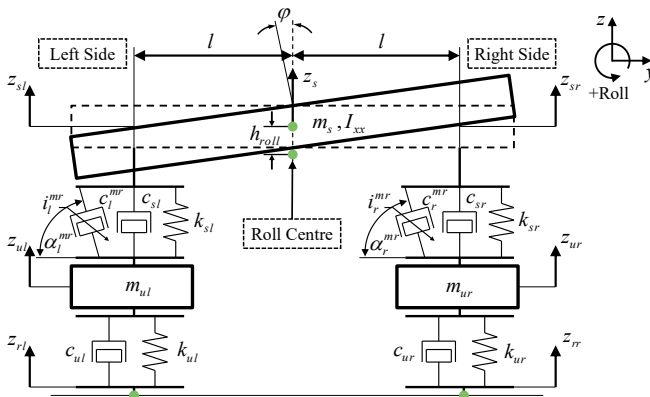


Fig. 1: Body diagram for roll motion

In the vehicle one can distinguish the sprung mass (vehicle body) and the unsprung mass (wheels). The forces acting in the vertical direction at the points

of support of the sprung mass are given by:

$$\begin{aligned} F_r(t) &= -c_{sr}(\dot{z}_{sr} - \dot{z}_{ur}) - k_{sr}(z_{sr} - z_{ur}) + F_r^{mr} \sin(\alpha_r^{mr}), \\ F_l(t) &= -c_{sl}(\dot{z}_{sl} - \dot{z}_{ul}) - k_{sl}(z_{sl} - z_{ul}) + F_l^{mr} \sin(\alpha_l^{mr}), \end{aligned} \quad (1)$$

where F_r^{mr} and F_l^{mr} represent forces generated by the MR dampers mounted in the right and left side of the suspension. Viscous damping coefficients c_r^{mr} and c_l^{mr} of the MR dampers can be changed by applying the control currents i_r^{mr} and i_l^{mr} , respectively. In general, the relationships $F_r^{mr}(i_r^{mr})$ and $F_l^{mr}(i_l^{mr})$ are non-linear and are discussed in details in Section 3. Angles α_r^{mr} and α_l^{mr} describe the configuration of the MR dampers in the suspension (Fig. 1). The sprung and unsprung parts of the vehicle are denoted by subscripts s and u , respectively.

Equations describing vertical displacement and rotation of the sprung mass take the following form:

$$\begin{aligned} m_s \ddot{z}_s(t) &= F_r(t) + F_l(t) - F_g, \\ I_{xx} \ddot{\varphi}(t) &= -lF_l \cos\varphi + lF_r \cos\varphi + h_{roll}(F_y \cos\varphi + F_g \sin\varphi), \end{aligned} \quad (2)$$

where F_y is the lateral (horizontal) force in the direction of y axis caused by the road irregularity or cornering manoeuvres, I_{xx} is the moment of inertia of the sprung mass with respect to the roll axis, h_{roll} is the distance between the centre of gravity of the sprung mass and the roll centre, φ is the roll angle of the sprung mass, l is the transverse distance between the centre of gravity of the sprung mass and the points of support.

Substituting (1) into (2) yields:

$$\begin{aligned} m_s \ddot{z}_s(t) &= -c_{sl}(\dot{z}_{sl} - \dot{z}_{ul}) - k_{sl}(z_{sl} - z_{ul}) + F_l^{mr} \sin(\alpha_l^{mr}) \\ &\quad - c_{sr}(\dot{z}_{sr} - \dot{z}_{ur}) - k_{sr}(z_{sr} - z_{ur}) + F_r^{mr} \sin(\alpha_r^{mr}) - m_s g, \\ I_{xx} \ddot{\varphi}(t) &= l c_{sl}(\dot{z}_{sl} - \dot{z}_{ul}) \cos\varphi + l k_{sl}(z_{sl} - z_{ul}) \cos\varphi - l F_l^{mr} \sin(\alpha_l^{mr}) \cos\varphi \\ &\quad - l c_{sr}(\dot{z}_{sr} - \dot{z}_{ur}) \cos\varphi - l k_{sr}(z_{sr} - z_{ur}) \cos\varphi + l F_r^{mr} \sin(\alpha_r^{mr}) \cos\varphi \\ &\quad + h_{roll}(F_y \cos\varphi + m_s g \sin\varphi). \end{aligned} \quad (3)$$

To obtain a half-car model, equations (3) are complemented by equations of motion of the unsprung masses:

$$\begin{aligned} m_{ur} \ddot{z}_{ur} &= c_{sr}(\dot{z}_{sr} - \dot{z}_{ur}) + k_{sr}(z_{sr} - z_{ur}) - c_{ur}(\dot{z}_{ur} - \dot{z}_{rr}) - k_{ur}(z_{ur} - z_{rr}) \\ &\quad - F_r^{mr} \sin(\alpha_r^{mr}) - m_{ur} g, \\ m_{ul} \ddot{z}_{ul} &= c_{sl}(\dot{z}_{sl} - \dot{z}_{ul}) + k_{sl}(z_{sl} - z_{ul}) - c_{ul}(\dot{z}_{ul} - \dot{z}_{rl}) - k_{ul}(z_{ul} - z_{rl}) \\ &\quad - F_l^{mr} \sin(\alpha_l^{mr}) - m_{ul} g. \end{aligned} \quad (4)$$

Displacements and velocities of road-induced excitations of the right and left wheels are denoted as z_{rr} , \dot{z}_{rr} and z_{rl} , \dot{z}_{rl} , respectively. Displacements and velocities of the sprung mass at the points of support are related to displacement and velocity of the centre of gravity of the sprung mass by the following relations:

$$\begin{aligned} z_{sr} &= z_s + l \sin \varphi \implies \dot{z}_{sr} = \dot{z}_s - l \dot{\varphi} \cos \varphi. \\ z_{sl} &= z_s - l \sin \varphi \implies \dot{z}_{sl} = \dot{z}_s - l \dot{\varphi} \cos \varphi. \end{aligned} \quad (5)$$

Parameters of the simulated half-car model can be found in [11].

Preliminary results of the precise half-car model showed that the roll angle varies in the range of a few degrees. Furthermore, synthesis of the proposed clipped LQ control needs the linear matrix form of the vehicle model. Thus, for the purpose of further analysis and synthesis of the control scheme the presented half-car model of roll dynamics was linearised assuming small roll angles, and the following replacement was proposed for the angular expressions: $\sin \varphi \Rightarrow \varphi$ and $\cos \varphi \Rightarrow 1$. Additionally, since all elements of the suspension model represent a linear stiffness relationship, a gravitational acceleration was neglected in the model. Thus, the matrix form of the linearised half-car model can be formulated as follows:

$$\dot{X} = A \cdot X + B \cdot U + T \cdot M_r, \quad (6)$$

where $X = [(z_{uf} - z_{rf}), \dot{z}_{uf}, (z_{ur} - z_{rr}), \dot{z}_{ur}, (z_{sf} - z_{uf}), (z_{sr} - z_{ur}), \dot{z}_s, \dot{\varphi}_s]$ represents a vector of state variables, $U = [F_r^{mr}, F_l^{mr}]$ denotes a vector of MR damper forces, and M_r is the additional roll torque of the body. Matrices A , B , and T can be derived directly from the equations (1) to (5).

3 MR damper models for simulation and control

The semi-active vehicle suspension system is equipped with MR dampers which are the energy-dissipating devices controlled by a magnetic field. They are filled with MR fluid whose viscosity changes in the external magnetic field, making it to behave as free-flowing in the absence of magnetic field and as a viscous liquid, or even semi-solid, with increasing magnetic field strength. Consequently, damping coefficient c^{mr} of the MR damper changes upon the applied control current i^{mr} . As a result, the force yield by the damper may change even by an order of magnitude with a very small energy consumption. Therefore, the MR dampers seem to be very attractive in semi-active suspension. However, a detailed analysis of their behaviour revealed that the force-velocity characteristics is non-linear with the hysteresis loop and saturation. There exist a number of references that propose different models of the MR damper behaviour. Among them, the Bouc-Wen model [5, 16] is considered to be one of the more accurate and this model is applied in this paper. It is described by the following equation:

$$F_{bw}^{mr} = f(z_{mr}, \dot{z}_{mr}) + \alpha_{bw} w, \quad (7)$$

where $f(z_{mr}, \dot{z}_{mr})$ is the non-hysteretic component being a function of the instantaneous relative displacement z_{mr} and velocity \dot{z}_{mr} of the damper piston, and w is an evolutionary variable representing the hysteretic component of the

model. Parameter α_{bw} is a scaling factor for the Bouc-Wen model. The non-hysteretic component is given by:

$$f(z_{mr}, \dot{z}_{mr}) = c_{bw}\dot{z}_{mr} + k_{bw}(z_{mr} - x_{bw}), \quad (8)$$

where c_{bw} , k_{bw} are the viscous damping and stiffness coefficients, respectively. An initial displacement of the spring x_{bw} is introduced to model the impact of the gas accumulator in the MR damper. The hysteretic component w is described by the non-linear first order ordinary differential equation:

$$\dot{w} = -\gamma_{bw} \cdot |\dot{z}_{rm}| \cdot w \cdot |w|^{n-1} - \beta_{bw} \cdot \dot{z}_{mr} \cdot |w|^n + A_{bw} \cdot \dot{z}_{mr}. \quad (9)$$

Parameters γ_{bw} , β_{bw} , A_{bw} and n are used to shape the hysteresis loop. The scale and general shape of the hysteresis loop are governed by γ_{bw} , β_{bw} and A_{bw} , whereas the smoothness of the force-velocity characteristic is controlled by n .

To obtain the model which is valid for the varying strength of the magnetic field, the model parameters α_{bw} and c_{bw} are assumed to be current-dependent. They can be approximated by the third order polynomial according to the recommendation given in [18]:

$$\alpha_{bw} = \sum_{i=0}^3 \alpha_i^{bw} (i^{mr})^i, \quad c_{bw} = \sum_{i=0}^3 c_i^{bw} (i^{mr})^i. \quad (10)$$

The other parameters are assumed to be constant. Values of the identified parameters of the Bouc-Wen model are listed in Tab. 1. The model was fitted to the measurement data obtained for the commercially available MR damper produced by Lord Corporation in identification experiments performed on the MTS (*Material Testing System*) machine.

It is observed in practice that the response time of the MR damper depends on the control current and kinematic excitation. In real-world applications this response time is approximately within the range from 20 ms to 40 ms [10]. Thus, in the suspension model a first order filter with the time constant $T_{mr} = 12$ ms was added at the output of the Bouc-Wen model.

Most of the strategies used to control semi-active suspension systems require the so-called inverse model of the MR damper. This model should provide calculation of the control current for a given (i.e. measured) piston velocity and for the damping force determined by the control algorithm. Here, it is proposed to approximate the MR damper behaviour by a model based on the hyperbolic tangent function as follows [9, 15]:

$$F_{ht}^{mr} = -\alpha_{ht} \tanh[\beta_{ht} \dot{z}_{mr} + \gamma_{ht} \text{sign}(z_{mr})] - c_{ht} \dot{z}_{mr} - k_{ht} z_{mr}, \quad (11)$$

where $\alpha_{ht} = \alpha_0 + \alpha_1 \sqrt{i^{mr}}$ is a factor defining the height of the hysteresis, β_{ht} is the scale factor of the damper velocity defining the slope of the hysteresis, γ_{ht} is the scale factor determining the width of the hysteresis, c_{ht} and k_{ht} contribute to the representation of a conventional damper without hysteresis. The inverse model can be determined explicitly as:

$$i_{ht}^{mr} = \frac{1}{\alpha_1^2} \left(\frac{-F_{ht}^{mr} - c_{ht} \dot{z}_{mr} - k_{ht} z_{mr}}{\tanh[\beta_{ht} \dot{z}_{mr} + \gamma_{ht} \text{sign}(z_{mr})]} - \alpha_0 \right)^2, \quad (12)$$

where i_{th}^{mr} denotes the value of control current that yields the desired force F_{mr}^{ht} for piston velocity \dot{z}_{mr} .

Such approach makes the simulation closer to reality, in this sense that the model used for control constitutes an approximation of the real plant (here simulated as the Bouc-Wen model).

Table 1: Parameters of the MR damper models

Bouc-Wen model of the MR damper		
$i_{mr} \in (0.0 ; 1.33)$ A		$T_{mr} = 12$ ms
$n = 2$	$k_{bw} = 0.001$	$x_{bw} = 1.5$
$[\alpha_0^{bw}, \alpha_1^{bw}, \alpha_2^{bw}, \alpha_3^{bw}] = [93506, 888021, 27374, -294583]$		
$[c_0^{bw}, c_1^{bw}, c_2^{bw}, c_3^{bw}] = [792, 4195, -6390, 2565]$		
$\gamma_{bw} = 987288$	$\beta_{bw} = 983237$	$A_{bw} = 7.979$
Tanh-based model of the MR damper		
$\alpha_0 = -23.05$	$\alpha_1 = 1215$	$\beta_{ht} = 36.47$
$\gamma_{ht} = 1.6$	$c_{ht} = 1203$	$k_{ht} = 1297$

4 Clipped LQ control dedicated to driving safety

The quarter Skyhook control related to the quarter-car model is generally applied for the semi-active devices due to its simplicity. This algorithm is used here as a reference to be compared with the clipped LQ control. Each quarter of the vehicle suspension denoted with index i is controlled separately using the quarter Skyhook according to the following formula:

$$F_{ht,i}^{mr}(n) = -g_{v_{s,i}} \cdot v_{s,i}(n), \quad (13)$$

where $v_{s,i}$ denotes velocity of the selected quarter of the vehicle body. The gain $g_{v_{s,i}}$ was obtained by trial and error method in order to improve the driving safety performance index defined by (19).

The quarter Skyhook algorithm can be generalised to the multi-dimensional clipped LQ control. Constraints denoted as S_X and S_U are assumed as the input parameters of the considered control scheme. They are set on the selected state and control variables, respectively, and are related to their maximum acceptable amplitudes. For the clipped LQ control the following time-infinite cost function is minimized:

$$J = \int_0^{\infty} [X(\tau)^T Q X(\tau) + U(\tau)^T R U(\tau)] d\tau, \quad (14)$$

where Q and R denote cost matrices related to the state and control variables denoted as X and U , respectively. The minimized cost function (14) is valid for vibration control applied for roll motion. Matrices Q and R are obtained according to the Bryson's rule [2], as follows:

$$Q = S_X^{-2} \quad , \quad R = S_U^{-2}. \quad (15)$$

The clipped LQ problem can be evaluated by solving the continuous-time Algebraic Riccati Equation (ARE):

$$A^T P + PA + BPR^{-1}B^T P + Q = 0, \quad (16)$$

where P denotes the solution of ARE used for determination of the control gains G according to the following formula:

$$G = R^{-1}B^T P. \quad (17)$$

Matrices A and B were defined in (6) obtained by reformulation of the differential equations (3 - 5) into the matrix form.

Because in the real measurement system a limited number of sensors is available, so it was also assumed that the vector of output variables is limited. Only heave and pitch velocity of the body denoted as v_s and ω_s , respectively, were selected since they have decisive influence on vehicle handling. Thus, the desired control forces for the analysed pairs of the front and rear, or right and left MR dampers are expressed as follows:

$$F_{ht,i}^{mr}(n) = -g_{v_s,i} \cdot v_s(n) - g_{\omega_s,i} \cdot \omega_s(n), \quad (18)$$

where the selected control gains $g_{v_s,i}$ and $g_{\omega_s,i}$ are included in the vector G and are evaluated according to equations (16) and (17), subject to constraints assumed in matrices S_X and S_U . Finally, the desired values of the control current i_{ht}^{mr} are determined based on the inverse MR damper model (12) using control forces $F_{ht,i}^{mr}$ obtained for quarter Skyhook or clipped LQ control.

5 Results of simulation

The algorithms were validated for an impulse excitation generated as a torque which influences roll motion of the vehicle body. Simulation environment consists of two modules: one related to the vehicle dynamics simulation and the other corresponding with the control algorithm execution. The differential equations related to the half-car model and to the Bouc-Wen model were solved numerically using the Runge-Kutta method with variable integration step. The control algorithms were executed with the sampling period equal to 2 ms. Such approach corresponds to the real-world application, where a continuous plant is controlled by a digital controller with the sampling period limited by the hardware.

The quality of vibration control was validated using the following RMS-based performance index related to vehicle handling (VH) and calculated in the time domain:

$$J_{VH} = \sqrt{\frac{1}{N} \sum_{n=1}^N x^2(n)}, \quad (19)$$

where $x(n)$ is the roll angle and N is the number of simulated samples. In order to achieve comparable operating conditions for both control strategies, their parameters were optimised with respect to the vehicle dynamics. Results of the algorithms optimised for sudden turning manoeuvres are presented in Fig. 2. It can be noticed that the clipped LQ control allows for obtaining the better value of the performance index than the quarter Skyhook. However, it is more sensitive to the proper tuning of the algorithm contrary to the quarter Skyhook.

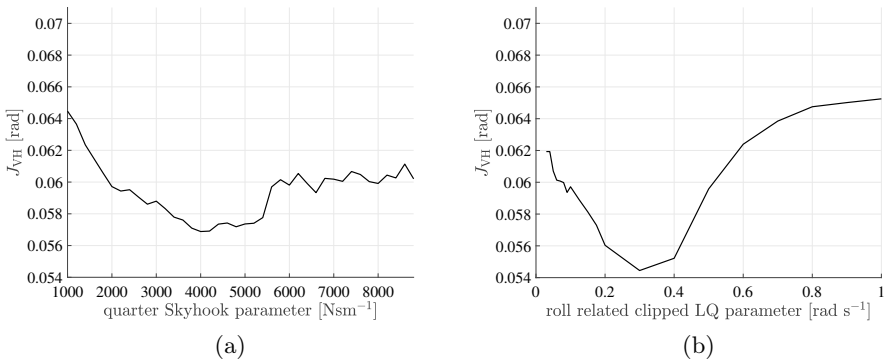


Fig. 2: Quality of safety control in the case of sudden turning: a) for different parameters of the quarter Skyhook control, b) for different parameters of the clipped LQ control

A similar analysis was performed for the passive suspension with the control current varying from 0 to 0.9 A. Generally, the results show that the higher level of control current means the greater value of the performance index, and consequently, the worse driving safety.

Torque impulse responses are compared for different control strategies in Fig. 3. It was shown that the semi-active control of the vehicle suspension is favoured over the passive suspension for both soft and hard types. Furthermore, it is recommended to extend the well-known quarter Skyhook approach to the clipped-LQ strategy if the sufficient number of measurement signals is available.

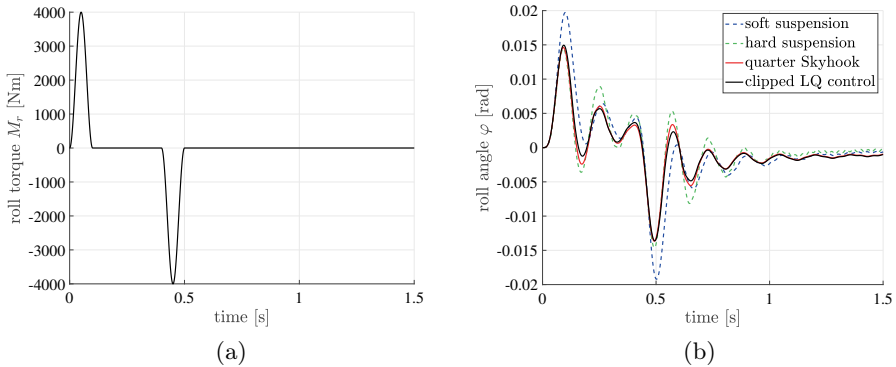


Fig. 3: Results of sudden turning simulation for passive suspension, quarter Skyhook and clipped LQ control: a) roll torque excitation, b) angle of the vehicle body roll

6 Conclusions

The problem of driving safety plays a crucial role in the exploitation of the road vehicles. The clipped LQ control was modified in order to improve driving safety of an off-road vehicle model equipped with the automotive MR dampers. The presented studies are based on simulations performed for the 4-DOFs half-car model and the Bouc-Wen model of the MR damper. Different control strategies were compared including the passive suspension as well as the semi-active control like the quarter Skyhook and the simplified clipped LQ control. The behaviour of the vehicle was analysed for situation related to turning manoeuvres. Similar considerations can be also made for pitch motion related, e.g., to sudden braking. Presented simulation results show the usefulness of the clipped LQ for improvement of driving safety.

Acknowledgement

The partial financial support of this research by Polish Ministry of Science and Higher Education is gratefully acknowledged.

References

1. Anwar, S.: A predictive control algorithm for a yaw stability management system. SAE paper doi:10.4271/2003-01-1284, 639—657 (2003)
2. Bryson, A.E.J., Ho, Y.C.: Applied optimal control: optimization, estimation and control. Hemisphere Publishing Corporation, USA (1975)

3. Caponetto, R., Dimanante, O., Fargione, G., Risitano, A., Tringali, D.: A soft computing approach to fuzzy skyhook control of semiactive suspension. *IEEE Transactions on Control Systems Technology* 2(6), 786–798 (November 2003)
4. Choi, S.B., Han, S.S.: H_∞ control of electrorheological suspension system subjected to parameter uncertainties. *Mechatronics* 13, 639–657 (2003)
5. Dyke, S.J., Spencer, B.F., Sain, M.K., Carlson, J.D.: Modeling and control of magnetorheological dampers for seismic response reduction. *Smart Materials and Structures* 5, 565–575 (1996)
6. Guglielmino, E., Sireteanu, T., Stammers, C.W., Ghita, G., Giuclea, M.: Semi-active suspension control, improved vehicle ride and road friendliness. Springer-Verlag London Limited (2008)
7. Hong, K.S., Sohn, H.C., Hedrick, K.: Modified Skyhook control of semi-active suspensions: a new model, gain scheduling, and hardware-in-the-loop tuning. *Journal of Dynamic systems, Measurement, and Control* 124, 158–167 (2002)
8. Hrovat, D.: Survey of advanced suspension developments and related optimal control applications. *Automatica* 33, 1781–1817 (1997)
9. Kasprzyk, J., Wyrwał, J., Krauze, P.: Automotive MR damper modeling for semi-active vibration control. *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM2014* pp. 500–505 (8–11 July Besancon, France, 2014)
10. Koo, J.H., Goncalves, F.D., Ahmadian, M.: A comprehensive analysis of the response time of MR dampers. *Journal of Smart Materials and Structures* 15, 351–358 (2006)
11. Krauze, P.: Comparison of control strategies in a semi-active suspension system of the experimental ATV. *Journal of Low Frequency Noise, Vibration and Active Control* 32(1-2), 67–80 (2013)
12. Krid, M., Benamar, F.: Design and control of an active anti-roll system for a fast rover. *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems* (25–30 September San Francisco, CA, USA, 2011)
13. Long, C., Li-min, N., Jing-bo, Z., Hao-bin, J.: Application of AMESim and MATLAB simulation on vehicle chassis system dynamics. *Proceedings of the Workshop on Intelligent Information Technology Application* (2007)
14. Savaresi, S.M., Poussot-Vassal, C., Spelta, C., Sename, O., Dugard, L.: Semi-active suspension control design for vehicles. Butterworth-Heinemann, Elsevier (2010)
15. Song, X., Ahmadian, M., Southward, S.C.: Modeling magnetorheological dampers with application of nonparametric approach. *Journal of Intelligent Material Systems and Structures* 16, 421–432 (May 2005)
16. Spencer, B.F., Dyke, S.J., Sain, M.K., Carlson, J.D.: Phenomenological model of a magnetorheological damper. *ASCE Journal of Engineering Mechanics* 123, 230–238 (1997)
17. Stone, M., Demetriou, A.: Modelling and simulation of vehicle ride and handling performance. *Proceedings of the 15th IEEE International Symposium on Intelligent Control* pp. 85–90 (17–19 June 2000)
18. Yang, G., Spencer, B.F., Carlson, J.D., Sain, M.K.: Large-scale MR fluid dampers: modelling and dynamic performance considerations. *Engineering Structures* 24, 309–323 (2002)
19. Zehsaz, M., Tahami, F., Paykani, A.: Investigation on the effect of stiffness and damping coefficients of the suspension system of a vehicle on the ride and handling performance. *U.P.B Scientific Bulletin, Series D: Mechanical Engineering* 76, 55–70 (2014)

Review of tracking and object detection systems for advanced driver assistance and autonomous driving applications with focus on vulnerable road users sensing

Paweł Markiewicz¹, Marek Długosz², Paweł Skruch^{1,2}

¹ Delphi Technical Center Krakow

ul. Podgorki Tynieckie 2, 30-399 Krakow, Poland

² AGH University of Science and Technology

Faculty of Electrical Engineering, Automatics,

Computer Science and Biomedical Engineering

Department of Automatics and Biomedical Engineering

al. A. Mickiewicza 30/B1, 30-059 Krakow, Poland

pawel.markiewicz@delphi.com, mdlugosz@agh.edu.pl, pawel.skruch@agh.edu.pl

Abstract. The paper presents review of key aspects associated with object detection and tracking algorithms in sensing systems for automotive applications covering active safety, advanced driver assistance and autonomous driving systems. The role of discussed systems is to establish location and monitor relative movement of objects and environment with main focus on identification of vulnerable road users. The object detection algorithms allow to classify and differentiate both static and dynamic objects in the vehicle's surrounding. The analysis will be focused on selection of most suitable platform for development of optimized and safe vulnerable road user detection system.

1 Introduction

As a result of recently publicized guidelines from vehicle safety organization such as [N1] Euro NCAP or [NT1] NHTSA the major scope of improvement of safety conditions in upcoming years is to focus the development and implementation of ADAS systems on detection of Vulnerable Road Users. According to the statement provided by Euro NCAP half of the fatalities in accidents involve VRUs, listed as most frequent scenarios are cyclist crossing, turning or traveling down the road. Along with increased urbanization those data are considered as the

driving element towards popularization of advanced driver assistance systems focused on counteracting hazards for pedestrian, cyclist and other vulnerable occupants of the road. Popularization which is indicated by organizations is directly linked to deployment of ADAS systems to lower segments, economy vehicles, thus imposing cost boundaries for such solutions.

1.1 Explanation of challenges and needs for VRU detection

To understand the challenges behind the perception systems used in Advanced Driver Assistance Systems and Autonomous Driving, we need to have an outline of what exactly the vehicle has to "see". From the point of view of the end user we only see features like Lane Keeping Aid, Highway Pilot or Intersection Assist or a system providing some level of autonomous driving. All the discussed features require information about surrounding environment with precise distinction between object types and parameters describing the surrounding environment. As the current state of technology allows us to differentiate between stationary and moving object with property assignment in abstract type definitions covering most of types of obstacles that could be encountered in traffic (vehicles - trucks, passenger cars, lanes, road signs, barriers, pedestrians, motorcyclists ect.) a special group has been defined mainly as a result of EURO NCAP organization guidelines, called Vulnerable Road Users. Scenarios in which detection of a pedestrian is obligatory to engage breaking are already implemented and exercised during rating test. Similar scenario for which [N2]detection of bicyclist model will be required is expected for 2018

From the point of view of automotive standards one of the most critical aspects considered in designing any automotive system is safety. [I1]This area of expertise over time evolved in to its own domain called Functional Safety, which is an independent factor driving requirements and architecture of implementations towards compliance with ISO 26262 and its derivatives. Second most significant challenge that is encountered in automotive design is the cost efficiency, which imposes certain limitations directly impacting hardware design, this influence on hardware design on the other hand, indirectly impacts the design of software and algorithms. Major points in which cost efficiency has an impact on algorithmic and software solutions is in available computational resources which can be split into two major factors: computational power and available mem-

ory, thus requiring thoughtful design as well as optimization of implementations.

As a result conditions mentioned above and advancing regulations, a demand for low cost reliable system has been created as more and more segments of vehicles will require capability for detecting VRU's. As well as increased demand in sensor performance (accuracy and resolution) is required for higher segments with focus on Autonomous Driving. In further chapters a review of existing systems will be provided with a review of technical principles standing behind various types of sensors common for both automated driving and ADAS. As a result of the review, a comparison of different methods will be presented with focus on detection capability, and safety levels allowed by each of the systems.

2 Presentation of currently existing systems

As a result of presented constrains several different approaches have been formed to tackle the problem of detection and tracking of traffic participants and environment. On a course of history first implementations of detection and ranging systems were not providing any abstract classification of objects, those systems were just used to determine the distance to whatever obstacle or object was receding it, without angular discrimination.[UTK1] First type of object that has been classified in detection methods for commercially available systems was in general form a "vehicle". Early stage systems mainly based on ultrasonic sensors (further called PDCs) and radars - evolving from single to dual beam and further to scanning radars. In parallel a development of vision based systems was ongoing, in the early stages the use of vision systems was mainly limited to lane detection to precisely locate the host inside the driving lane. Along with regular vision systems more specialized implementations have been presented to the market such as Near Infrared and Far Infrared cameras.

2.1 Radar Based

One most widespread type of sensors that is currently in use for commercial systems is a microwave radar, the principle of operation of this devices mainly relay on phased array transducers capable of generating high frequency electromagnetic modulated pulses. [UTK1]In typical solutions the frequencies of those pulses are either in 24 or 77GHz. A systemic diversification has been introduced

for separation into 3 major groups of such sensors, with respect on range of operation. Most of the automotive radars are continuous wave electronic scanning radars, that emit the series of frequency swept pulses, and after performing multiple FFTs extract the range, range rate and azimuth information which further is processed to extract the target information.

Long Range Typically Long range radars operate in ranges up to 250m in distance, [UTK1] the usual field of view of this type of device is narrow 10° to 15° . This type of devices are mainly used for features such as ACC for identification of large targets mainly vehicles - at a greater distance. In many implementation cases long range radars are integrated with medium range radars with switchable mode of operation allowing performing different types of scanning.

Medium Range This type of devices as mentioned before are usually fused with long range radars and used for front sensing - this means mounted on vehicles front end, usually the front grill or behind front bumper. Medium range radars are "workhorses" of the ADAS systems, providing capability of detection in much more diverse types of objects, such as vehicles, barriers road edges, pedestrians and motorcyclist, but for the last mentioned types classified as VRU a specific type has to be used that allows ASIL B or D level discrimination with use of Micro-Doppler principle to distinguish VRUs, for those objects to be used in consumer functions with such safety level (such as Autonomous Emergency Braking or Collision Avoidance with vehicle Path Change. Those devices provide the field of view of ranging from 65° to 70° .



Fig. 1. Example of Medium Range Radar - Delphi MRR.

Short Range Most recently systems of such type gained more popularity in autonomous driving applications mainly for environment sensing, output of the devices is applied to support of vehicle position and orientation estimation for autonomous driving, but is not only limited to that task, they provide sensible information on objects such similarly to medium range radar (i.e. vehicles, VRUs - with Micro Doppler, barriers ect.). They are in most use cases used in pairs mounted on corners of the host vehicle - providing view to the sides of the car. with some areas overlapping in front of the vehicles. Pairs are usually mounted in front and rear of the vehicle []. This devices in many cases are providing along the regular object information feature like free space detection, which later can be consumed in example by automated parking assistance or lane change merge aid functions. typically the range of those devices is up to 80m and operate in same frequencies as allocated for automotive radars.

2.2 Vision Based

Next very heavily used family of sensors that find application in all solutions ranging from driver support systems to automated driving solutions, are based on image processing. Two main types can be distinguished from main principle of operation point of view, systems that base on visible light spectrum come mainly in two forms - single vision cameras and stereo vision. Those types of systems gained much popularity due to easy implementation of image processing methods (mainly software) not much sophisticated hardware is required like in case of radars.

Mono Vision The most common approach in application of the vision sensing systems come in a form of single lens and single image matrix solution allowing capturing of image data in resolutions spreading from VGA up to UHD resolutions, providing effective field of view of 70deg. In early stages of commercial implementations the systems were used mainly for road edge and lane detection, in most recent systems a widespread set of objects can be safely identified. The main advantage of camera systems is ability to distinguish various types of objects, on the other hand the ranging and velocity measurements are not the strongest attitude. As cameras are ambient sensors they have to be able to cope with many different factors influencing their ability to "see", such as weather conditions, occlusion or darkness.

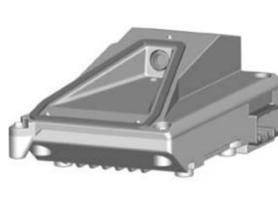


Fig. 2. Example of Mono Vision Camera - Delphi IFV200.

Stereo Vision Systems equipped with 2 cameras are much more capable in acquiring geometrical parameters from the environment, such as distance and height of the object. Ability to extract distance is based on capability to determine the perspective. Current advanced versions of these sensors in commercial segments find main application in front sensing. [UF1] Similarly to mono vision systems this solution is prone to problems with ambient conditions yet due to redundancy provides more reliable information than optical systems with one lens. Various different implementations exist with dual lens solutions, in which there are two lenses with different focal lengths feeding alternately image to one sensor.

NIR Due to the high dependence on ambient conditions of regular monochrome cameras attempts have been taken in integration of near infrared spectrum cameras for automotive purposes, along with integration of such systems for image processing (1 in each 4 pixels in sensors used to provide intensity in NIR spectrum) auxiliary systems were provided for the driver in form of head up displays with live streaming of superimposed images providing wider view in bad weather conditions and in darkness. Integration of NIR sensing into mono vision sensors is a common practice in modern ADAS sensing systems.

FIR As an extension of solutions provided by NIR sensors similar approach has been applied to sensing that is focused in spectrum not including visible light in any way. This approach enables providing higher contrast on objects with higher temperature, similarly as in previously discussed sensor stand alone operation without monochrome in visible spectrum is not most efficient solution.

Surrounding Cameras Image based sensors find application in sensing not only the area in front of the car but as a source of information on area behind and on the sides of the car. In commercial application the main features that

consume data from this sensors can be split in to two modes of operations. In low speeds the system composed of 4 cameras building a 360° field of view sometimes called bird eye's view or top view. Usually the data from this kind of sensor is merged in external processing unit and then processed by sensing algorithm that determine obstacles position mainly for automated parking. Second implementation is based on 2 cameras mounted under side mirrors providing data on lane markers and surroundings along with object data for features warning the driver on presence of objects in so called blind spot.

2.3 Laser

Ever since the invention of laser, the system has found application in ranging applications, the described systems base on the similar principle yet imply two different approaches for beam forming and apply to different philosophies of sensor application.

Laser Rangefinders Simple range finders can be found in many commercially available systems as a low cost ADAS sensor for autonomous emergency braking, This sensors usually do not provide a capability to distinguish between object types in front of the vehicle yet provide information on distance to receding and stationary object in narrow field of view. Due to low cost this systems have been broadly implemented in various segments of passenger cars providing a reasonable impact on number of accidents.

LIDAR The most know sensor applied currently in research and development of autonomous driving system.[UTK1]In Laser Imaging, Ranging and Detection systems most common approach in generation of 3D meshes is to measure range to objects using laser beam deflected by a moving mirror, during a rotation and tilting of the mirror several number of measurements is carried out to estimate distance and in some applications velocities of points reached by laser beam. Commercial application of currently existing solution is limited due to still high unit cost of production of those devices [EA1]. Systems on the other hand provide very detailed representation of the environment what imposes also high demands in computational power for processing the data obtained from such sensor. Processing is required for extraction of abstract definition of objects for further processing in control algorithms for automated driving purposes. Currently an intensified research and development efforts are invested in introduction of solid

stare LIDARs, which would be much more compact and lower cost than their industrial derivatives.



Fig. 3. Examples of Automotive LIDARs - Velodyne LIDAR Family (<http://velodynelidar.com/>)

2.4 Other

Apart from the presented previously system, there are sensor that don't fall exactly into the defines sub types.

PDCs Parking Distance Control sensors have found a successful application in supporting automated parking applications as well as adaptive cruise control and blind spot monitoring. [FMC1] Those system usually rely on ultrasonic sensors which emit acoustic pulses in frequencies above 20kHz. Angular discrimination is usually obtained by placing independent sensors along the bumper and measurement the round trip time of the pulse. Those systems are pretty primitive in comparison to previously discussed systems but are good enough for low speed operations even though they do not provide an abstract interpretation of the environment. Major advantage of those sensors is the low cost of production.

Fusion (Radar + Camera in one sensor) Devices based on merging two sensors gained popularity on the market as they provide a reliable source of information on moving and stationary objects in front of the vehicle, off course sensor fusion is a popular method of increasing the reliability of measurements and is commonly implemented with all mentioned above sensor, yet on the system level such fusion is carried out by an external device (like centralized safety unit). Yet there exist commercially successful implementations in which part of the sensor fusion is carried out in the device directly [LB1]. In case of fusion of radar system (medium and long range) with mono vision camera, objects detected and identified by camera are used to refine object information from radar

systems. Due to redundant implementation of sensors it is possible to implement a functional safety mechanisms that allow to identify faults for achieving high goals in functional safety requirements, this includes VRU detection for functions imposing direct activation of breaks.



Fig. 4. Example of Integrated Fusion System - Delphi RACam (Integrated Radar and Camera).

3 Synthetic Comparison

In this chapter a comparison of key components will be provided in tab1

	Radar			Camera				Laser			Other	
	Long Range	Medium Range	Short Range	Mono	Stereo	NIR	FIR	Surrounding	Rangefinders	LIDAR	PDCs	Fusion (Camera + Radar)
Performance	Range	++	+	+	+	+	+	-	+	+	-	++
	Long-Performance	++	++	-	+	-	-	-	+	+	-	++
	Lat-Performance	-	-	-	++	++	0	+	0	+	-	++
	Environmental Conditions	+	+	+	-	-	-	-	-	-	0	+
	Low Ambient Lighting	++	++	++	-	-	+	++	++	++	++	+
	Object Classification	0	0	-	++	++	++	+	-	+	-	++
	Stationary Objects Detection	-	-	-	++	++	++	++	+	+	++	++
Price	System	++	+	+	+	-	-	-	++	-	++	-
	Development	++	++	++	+	+	+	-	++	+	++	++
Pros	longitudinal performance -cost	-longitudinal performance -cost	-versatility -cost	-lateral performance -object classification	geometrical parameters estimation -object classification	-night vision	-warm object detection	-top view -close range sensing	-cost -low light performance	-precision -low light performance	-cost	-redundancy in geometrical parameters estimation
Cons	-lateral performance	-lateral performance -VRU	-VRU -lateral performance -short distance sensing	-longitudinal performance -weather influence	-weather influence	-weather influence -cost	-not available at day-time	-range -weather influence	-lack of object identification	-cost -still under development	-limited application	-size

4 Proposal of Sensor Optimization for Fusion Purposes

Depending on direct application - with regards on consumer of sensor data different levels of interaction between sensors are implemented in commercially existing solutions. As with every kind of implementation done for automotive industry - optimization of the cost vs. performance has to be accounted in the design process.

Various different methods for performing data fusion has been presented described and implemented through recent years. [HE1]From fully centralized methods in which raw data from sensors is fed to central processing unit for deriving object properties - Example A, to approaches in which already processed information in a form of object information is supplied to central unit for sensor fusion - Example B through hybrid approaches in which object data from one type of sensor are used to facilitate object detection in different type of sensor - Example C.

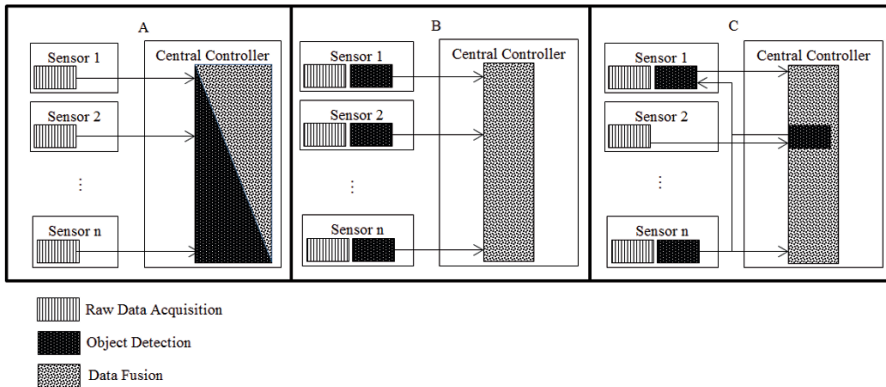


Fig. 5. Examples of Fusion Solutions

As most of the sensors operate on quite complicated raw data which usually would require large amounts of scarce memory to store (such as raw video data or raw radar reflection data), an approach is proposed in which chunks of raw data are being populated to centralized unit. This could be easily confused with original method presented in Example C and B - as objects generated by sensors when the detection part is performed inside the sensor - are in fact a derivative of

raw data. But in this case the idea behind this approach is to populate windowed out areas of low level data.

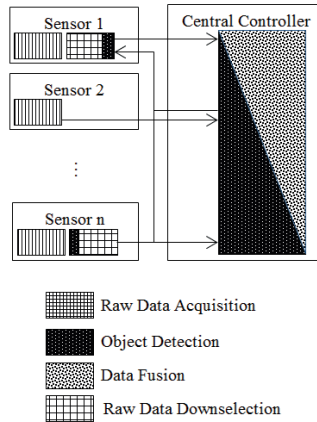


Fig. 6. Proposed Downselection Method

Populated chunks of low level data then would be processed for object detection which could include time domain tracking of low level information - as the central unit would allow for bigger computational resources and complex fusion algorithms.

As we are focusing on sensor design in this subject - this would allow reduction of sensor computational needs - to the point in which sensor is capable of determine regions of interest (without tracking) - from which low level data would be populated to central computational unit at a cost of higher need for bus throughput. The bus load on the other hand would be lower than in Example A from Fig 5. On top of that to reduce the computational needs of sensor. Area of interest selection will be auxiliary supported by information from central unit on already identified objects similarly to Example C. Proposed approach would be beneficial mainly for Radar and LIDAR systems which in most cases are limited to object building on their own. Reduction of tasks performed by an externally mounted sensor would transfer cost distribution towards the Centralized Controller, which is beneficial due to the fact sensor mounted on boundaries of the vehicle are prone to damage.

5 Conclusions and further steps

As a result of review of all currently applied methods of environment and obstacle sensing for Advanced Driver Assistance Systems and Autonomous Driving, a comprehensive comparison is provided to visualize advantages and disadvantages of certain solutions. The comparison had in mind not only blind performance comparison but as well taking into account influence of cost effectiveness and achievable safety goals. This comparison has as a target description and diversification of current state of the art solutions for better understanding on their application. This approach enables to reach a common ground between performance needs for performing required system tasks and cost impact delivered by a chosen solution. The review outlines possible paths of further development for some groups of systems for future increase of their competitiveness with optimization of sensors to cooperate with centralized fusion as a synergistic system.

References

- [N1] Eruo NCAP - 2020 ROADMAP - EUROPEAN NEW CAR ASSESSMENT PROGRAMME (2015)
- [NT1] Overview of NHTSA Priority Plan for Vehicle Safety and Fuel Economy, 2015 to 2017 (2015)
- [N2] Euro NCAPs Road Map 2020 The next steps for Vulnerable Road User AEB assessment
- [I1] International Standardization Organization ISO 26262-1:2011(en) Road vehicles Functional safety Part 1: Vocabulary
- [UF1] Uwe Franke, David Pfeiffer, Clemens Rabe, Carsten Knoeppel, Markus Enzweiler, Fridtjof Stein, Ralf G. Herrtwich - Making Bertha See (2013) IEEE International Conference on Computer Vision Workshops (ICCVW)
- [UTK1] Umit Ozguner, Tankut Acarman, Keith Redmill - Autonomous Ground Vehicles (2011)
- [FMC1] Ford Motor Company - See It, Hear It, Feel It: Ford Seeks Most Effective Driver Warnings for Active Safety Technology. Increased warnings indicate potentially hazardous lane changes. (2008)
- [EA1] Evan Ackerman - Cheap Lidar: The Key to Making Self-Driving Cars Affordable (2016) <http://spectrum.ieee.org/transportation/advanced-cars/cheap-lidar-the-key-to-making-selfdriving-cars-affordable>
- [LB1] Lindsay Brooke - Delphi's RACam integrated radar-vision system enables active-safety suites. (2015) <http://articles.sae.org/13953/>

[HE1] Hannes Estl - Sensor fusion: A critical step on the road to autonomous vehicles (2016) EE Times Europe. <http://www.electronics-eetimes.com/news/sensor-fusion-critical-step-road-autonomous-vehicles/page/0/4>

Part IV
Applications

The QDMC Model Predictive Controller for the Nuclear Power Plant Steam Turbine Control

Paweł Sokółski, Tomasz A. Rutkowski, and Kazimierz Duzinkiewicz

Gdańsk University of Technology
Faculty of Electrical and Control Engineering
psokolski@eia.pg.gda.pl, tomasz.adam.rutkowski@pg.gda.pl,
kazimierz.duzinkiewicz@pg.gda.pl

Abstract. There are typically two main control loops with PI controllers operating at each turbo-generator set. In this paper a model predictive controller QDMC for the steam turbine is proposed - instead of a typical PI controller. The QDMC controller utilize a step-response model for the controlled system. This model parameters are determined, based on the simplified and linear model of turbo-generator set, which parameters are identified on-line with RLS algorithm. It has been found that the proposed QDMC controller realize the reference trajectories of the effective power and the angular velocity, and damp the electromechanical oscillations with satisfactory quality in comparison to the typical PI and DMC controllers.

1 Introduction

The electrical energy plays unique and significant role in development of the modern society. With fast economic development, grows demand for the electric power. Accordingly, there is a growing need to increase the power plants efficiency and improve the electrical energy quality. The conventional power plants and nuclear ones, utilize turbine-generator sets, the steam turbine cooperating with the synchronous generator, to produce electrical energy. The aim of the work described in the paper is to improve the quality of the steam turbine control, operation of the whole turbo-generator set and in the consequence the electrical energy quality delivered to the power system network.

The steam turbine and the synchronous generator are the complex objects with non-linear character. Currently used methods for the steam turbine control are typically based on the Proportional-Integral (PI) controllers. With the current state of control theory and access to the modern computing units, with high computing power, it become possible to use the more complex and sophisticated control algorithms for control purposes. This paper deals with model predictive control (MPC) methodology for the stem turbine control purposes. With the MPC technology one can design a truly multi-variable optimizing control system that can handle the process constraints and accommodate the model-based knowledge combined with the hard measurements ([1], [10], [11], [15], [16]). Proposed in the paper controllers are designed and implemented in the form of Dynamic Matrix Control (DMC) and Quadratic Dynamic Matrix Control (QDMC)

algorithms [10]. In the first case, the DMC controller calculates moves on manipulated variables which minimize future predictions of controlled variable errors and constraint violations in the least-squares sense. And in the second case, the QDMC consists of the on-line solution of a quadratic program (QP) which minimizes the sum of squared deviations of controlled variable predictions from their set-points to maintaining predictions of constrained variables within bounds. In contrast with DMC controller, where constraints are enforced via least squares method, the use of a QP provides rigorous handling of constraint violations by formulating them as linear inequalities, and allowing tighter constraint control.

The mathematical models of the turbo-generator set elements can be divided into two main groups, the first of which is related to complex and accurate models, while the second represents less accurate, reduced and simple models. These two groups have different applications and purposes. The first group models may be used in design, or detailed analyses of phenomena and nature of processes occurring in the turbo-generator set elements ([4], [6], [8]). On the other hand, the less accurate models composing the second group can be used in control systems synthesis, and for education or training purposes ([9], [12], [13]). The second group models should also comply with several aspects which the first group is unable to fulfil, for instance easy implementation, convenient calculation time, or simple description. The DMC and QDMC controllers use a system unit step-response model. Taking into account the system wide range of operating point changes, the step-response model parameters are calculated, based on the simplified linear model of the turbo-generator set, which parameters are identified on-line with the recursive least squares algorithm (RLS). The models from the second group are used in the paper as a reference models for proposed controllers evaluation during simulation tests.

The paper is organized as follows. In Section 2 the typical control structure of the turbo-generator set is briefly described. In Section 3 the complex and simplified linear models of the steam turbine and the synchronous generator are shortly presented. In Section 4 the steam turbine control structure with DMC/QDMC controller is described. The results of simulation tests are presented in section 5. Finally, a brief summary of the obtained results is given in Section 6.

2 The turbo-generator set - typical control structure

The turbo-generator set consist of steam turbine and asynchronous generator. The work produced by steam in the steam turbine sections has a form of the rotary energy of the steam turbine shaft. The steam turbine is installed on the one shaft with the synchronous generator. Hence, the rotary energy is further converted into electrical energy by the synchronous generator.

The Fig. 1 present the simple diagram of the steam turbine cooperating with the synchronous generator, which are connected to the power system network. There are also shown their typical control system structures, with two main control loops operating at each turbo-generator set. The PI controller in the first control loop, regulates generator's active power by manipulating the steam mass

flow rate to the turbine, and in consequence impacting on the mechanical torque. The generators active power is almost equivalent to its electrical torque, hence mechanical and electrical torques must be equal to keep system in its steady condition. While, the PI controller in second control loop, regulates voltage on the generator’s terminal by manipulating the exciter voltage in the synchronous generator excitation system. This second PI controller, typically cooperate with the power system stabilizer ([6]). The power system stabilizer generate additional control signal for the PI controller in order to damp the potential electromechanical oscillations and to improve the dynamic stability of the turbo-generator set connected to the power system network. In the paper is assumed that the second control loop operate with the typical PI controller and power system stabilizer.

In order to improve the performance of the main control loops (Fig. 1), a numerous techniques have been proposed by various authors over the years [13]. In the paper, the data of the steam turbine 4CK-465 [8] and the synchronous generator GTHW-600 [4] are used in all simulation tests. That turbo-generator set was planned to be used in a first polish nuclear power plant (NPP) in Żarnowiec in 1989.

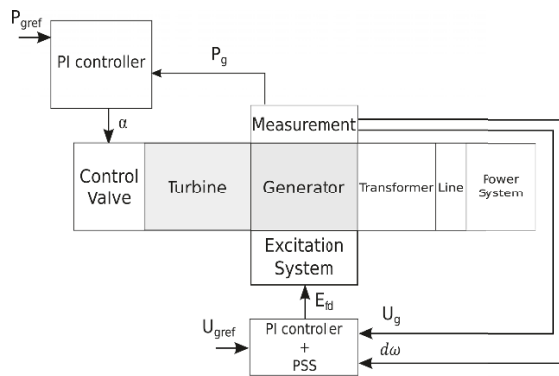


Fig. 1. Typical control system structure of the turbo-generator set.

3 The models of the turbo-generator set elements

3.1 The complex models

Steam turbine model The complex model of the steam turbine, used in the paper as a reference model, is described in terms of the mass and energy conservation equations, semi empirical relations and thermodynamic state conservation ([3], [5], [8], [9]). The stem turbine mass conservation equation, angular speed of turbine shaft and total mechanical power generated by the steam turbine may be presented shortly by set of the ordinary differential and algebraic equations

equations [8]

$$\frac{dp_n}{dt} = \frac{1}{\tau_n} [\dot{m}_{n,in} - \dot{m}_{n,out}], \quad (1)$$

$$P_{mech} = \sum_{i=1}^n P_i = \sum_{i=1}^n \Delta h_i \dot{m}_{i,out}, \quad (2)$$

$$\frac{d\omega_r}{dt} = \frac{1}{\Theta} \left(\frac{P_{mech}}{2\pi f} - 2\pi b f - M_g \right), \quad (3)$$

where: p_n denote steam pressure in the n -th steam turbine interstage space, τ_n is a time constant, $\dot{m}_{n,in}$ and $\dot{m}_{n,out}$ are steam mass flow through the n -th turbine stage, P_{mech} is overall steam turbine mechanical power, P_n is mechanical power of the n -th steam turbine stage, Δh_n is change of enthalpy at n -th steam turbine stage, Θ is moment of inertia, ω_r is angular speed of the steam turbine shaft, b is damping coefficient, M_g is synchronous generator torque, and f is shaft rotation frequency. All equation of the steam turbine complex model are presented in [12].

Synchronous generator model The complex model of the synchronous generator, used in the paper as a reference model, may be presented in the form of basic non-linear differential and algebraic equations ([4], [6])

$$[U_d, U_q, E_{fd}, 0, 0]^T = [Z] [I_d, I_q, I_{fd}, I_{kd}, I_{kq}]^T + \frac{1}{2\pi f} [\dot{\Psi}_d, \dot{\Psi}_q, \dot{\Psi}_{fd}, \dot{\Psi}_{kd}, \dot{\Psi}_{kq}]^T, \quad (4)$$

$$[\Psi_d, \Psi_q, \Psi_{fd}, \Psi_{kd}, \Psi_{kq}]^T = [X] [I_d, I_q, I_{fd}, I_{kd}, I_{kq}]^T, \quad (5)$$

$$P_g = U_d I_d + U_q I_q \quad Q_g = U_d I_d - U_q I_q, \quad (6)$$

$$M_g = \Psi_d I_q + \Psi_q I_d, \quad (7)$$

where: U_d, U_q are generator's voltages in q and d axis; E_{fd} is excitation voltage; $I_d, I_q, I_{kd}, I_{kq}, I_{fd}$ are stator, damper winding and field winding currents; $\Psi_d, \Psi_q, \Psi_{kd}, \Psi_{kq}, \Psi_{fd}$ are magnetic fluxes; f is frequency; P_g, Q_g are generator's active and reactive power; M_g is electrical torque; X, Z are reactance and impedance matrices (angular velocity and magnetic saturation depended). All equations of the synchronous generator complex model are presented in ([4], [6]), while its parameters may be found in [13].

3.2 The simplified linear models

Steam turbine model The complex model of the steam turbine has been reduced to the simplified linear model with two first order inertias [12], where each of them represent high and low pressure turbine corps, respectively. That model can be described in Z-domain as follows

$$P_{mech}(z) = g_t(z)\alpha(z) = K \left(\frac{K_1}{\frac{2T_1(z-1)}{T(z+1)} + 1} + \frac{K_2}{\frac{2T_2(z-1)}{T(z+1)} + 1} \right) \alpha(z), \quad (8)$$

where: P_{mech} is stem turbine mechanical power, α - degree of the control valve opening, and K, K_1, K_2, T_1, T_2 are model parameters. Notice, that the numerical values of the model parameters identified for various operating points of the steam turbine 4CK-465 may be found in [12].

Synchronous generator model The complex model of the synchronous generator has been reduced to the simplified 5-th order linear model, which may be presented in Z-domain in the following matrix form [7]

$$\begin{bmatrix} \omega(z) \\ P_g(z) \end{bmatrix} = \begin{bmatrix} g_{11}(z) & g_{12}(z) \\ g_{21}(z) & g_{22}(z) \end{bmatrix} \begin{bmatrix} P_{mech}(z) \\ E_{fd}(z) \end{bmatrix}. \quad (9)$$

Notice that the detailed definition of g_{11}, g_{12}, g_{21} and g_{22} parameters may be found in [7].

Turbo-generator set model Taking into account the stem turbine model (Eq. 8) and the synchronous generator model (Eq. 9) the turbo-generator set model may be presented also in Z-domain as follows

$$\begin{bmatrix} \omega(z) \\ P_g(z) \end{bmatrix} = \begin{bmatrix} g_{11}(z)g_t(z) & g_{12}(z) \\ g_{21}(z)g_t(z) & g_{22}(z) \end{bmatrix} \begin{bmatrix} \alpha(z) \\ E_{fd}(z) \end{bmatrix}, \quad (10)$$

and finally, model may be presented as discrete model in time domain in its general form as follows

$$\omega(k) = \sum_{j=1}^7 a(j)\omega(k-1-j) + \sum_{j=1}^7 b(j)\alpha(k-j) + \sum_{j=1}^7 c(j)E_{fd}(k-j), \quad (11)$$

$$P_g(k) = \sum_{j=1}^7 d(j)P_g(k-1-j) + \sum_{j=1}^7 e(j)\alpha(k-j) + \sum_{j=1}^7 f(j)E_{fd}(k-j), \quad (12)$$

where a, b, c, d, e and f are model parameters vectors. Those parameters should be identified on-line according to the wide range of turbo-generator set operating point changes. In the paper the recursive least-squares (RLS) identification algorithms for that purposes is proposed. With the RLS algorithm the unknown model parameters (Eqs. 11-12) are estimated on-line based on the set of the input and output measurements data [2]. Next, based on the identified simplified model (Eqs. 11-12) a unit step-response model for the QDMC and DMC controllers is determined.

4 The turbo-generator set - proposed control structure

4.1 The QDMC and DMC algorithms

The multi-variable QDMC quadratic optimization problem for a system with S controller outputs (manipulated variable) and R measured variables can be

presented in the following form ([10])

$$\min_{\Delta \mathbf{u}} J = [\mathbf{e} - \mathbf{A}\Delta \mathbf{u}]^T \Gamma^T \Gamma [\mathbf{e} - \mathbf{A}\Delta \mathbf{u}] + [\Delta \mathbf{u}]^T \Lambda^T \Lambda [\Delta \mathbf{u}] \quad (13)$$

subject to the constrains

$$\Delta \mathbf{u}_{s,min} \leq \Delta \mathbf{u}_s \leq \Delta \mathbf{u}_{s,max}, \quad (14)$$

$$\mathbf{u}_{s,min} \leq \mathbf{u}_s \leq \mathbf{u}_{s,max}, \quad (15)$$

$$\mathbf{y}_{r,min} \leq \mathbf{y}_r \leq \mathbf{y}_{r,max}, \quad (16)$$

where r denote the r -th process measured variable ($r = 1, \dots, R$), s denote the s -th process manipulated variable ($s = 1, \dots, S$), \mathbf{e} is the vector of predicted errors for the R measured process variables over the next P sampling instants (prediction horizon), \mathbf{u} is the vector of manipulated variables changes for the S controller output variables computed for the next M sampling instants (control horizon), \mathbf{y}_r is the predicted process variable profile for the r -th measured process variable over the next P sampling instances, \mathbf{A} is the multi-variable dynamic matrix formed from the unit step response coefficients of each controller output to measured process variable pair, $\Gamma^T \Gamma$ is the matrix of controlled variable weights, and $\Lambda^T \Lambda$ is the matrix of move suppression coefficients. The matrix $\Lambda^T \Lambda$ is a square-diagonal matrix of dimensions $M \cdot S \times M \cdot S$. The leading diagonal elements of the i -th $M \times M$ matrix block along the diagonal of $\Lambda^T \Lambda$ are λ_i^2 ($i = 1, \dots, S$) – all off-diagonal elements are zero. Similarly, the $P \cdot R \times P \cdot R$ matrix of controlled variable weights $\Gamma^T \Gamma$, has the leading diagonal elements as γ_i^2 ($i = 1, \dots, S$) – all off-diagonal elements are zero.

Finally, optimal vector of changes in manipulated variables is obtained based on the solution of quadratic optimization problem (Eqs. 13-16). Only the first elements from that resulting vector are applied as the control signal to the plant. In the next time instant the optimization task is solved again.

The optimization problem (Eqs. 13-16) without inequality constraints (Eqs. 14-16) has a unique solution, which can be expressed as the DMC control law

$$\Delta \mathbf{u} = (\mathbf{A}^T \Gamma^T \Gamma \mathbf{A} + \Lambda^T \Lambda)^{-1} \mathbf{A}^T \Gamma^T \Gamma \mathbf{e}. \quad (17)$$

4.2 The control structure with the QDMC controller for the steam turbine

Typical turbo-generator set control system consists of the two control loops with the PI controllers (Fig. 1). In this paper instead of the typical PI controller a model predictive one QDMC for steam turbine is proposed (Fig. 2). It is done to improve the quality of the steam turbine control, the operation of the turbo-generator set and in consequence the electrical energy quality delivered to the power system network.

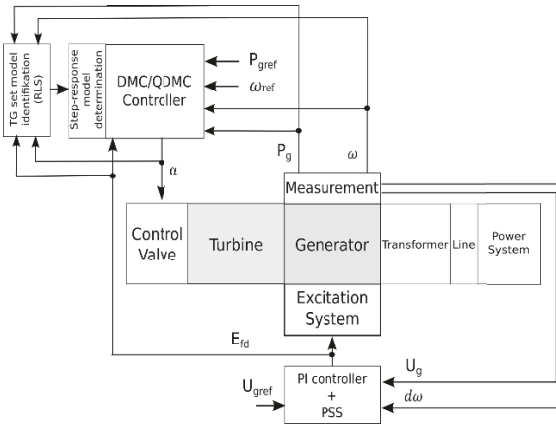


Fig. 2. Proposed control structure with DMC/QDMC steam turbine controller.

5 Simulation test results

The proposed control system and the reference process model were simulated with Matlab/Simulink environment. The results obtained with three different controllers for steam turbine were compared: typical PI controller with a standard power system stabilizer ([13], [14]), the DMC controller, and QDMC controller. During simulation studies, it was assumed that the synchronous generator was controlled by the typical PI controller cooperating with simple power system stabilizer. There was also assumed that the turbo-generator set was operated with the power system via the simple infinite-bus ([13], [14]).

The synchronous generator PI controller and power system stabilizer were characterized by following set of the parameters [14]: $K_P = 12.82$, $K_I = 29.03$, $T_1 = 0.65$, $T_2 = 1.74$. While, the parameters for DMC and QDMC controllers were assumed as follows: the control step $T = 0.01s$, the prediction horizon $P = 43$, the control horizon $M = 1$, and weights λ_i and γ_i equal 1. The constraints considered in the simulation test with the QDMC controller include the constraints on the control valve opening degree (manipulated variable) α .

The selected simulation results are based on the changes of the synchronous generator's active power reference trajectory. Initially synchronous generator worked with nominal active power equal $1p.u.$ ($470MW$), voltage on the generator's terminal equal $1p.u.$ ($21kV$) and angular velocity $1p.u.$ ($314rad/s$). Then active power set-point was changed $\pm 10\%$ at every 20 second of simulation. The results are shown in Figs. 3-6. To evaluate the performance of tested controllers, the performance index in the form of ISE criterion (integral of square error) was introduced and calculated (Tab. 1).

The use of DMC and QDMC predictive controllers increased the accuracy of the active power reference trajectory realization, but it causes deterioration of the generator voltage and angular velocity stabilization quality (Tab. 1). In that case the QDMC controller has almost four times smaller performance index than

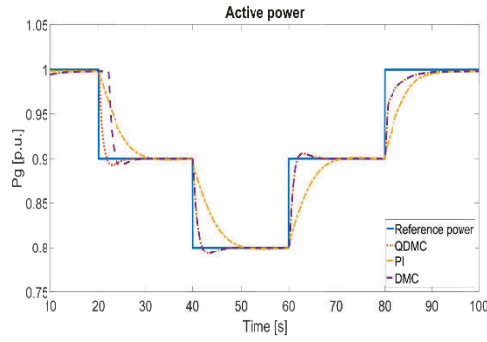


Fig. 3. The active power reference trajectory and its realization with PI, DMC and QDMC controllers.

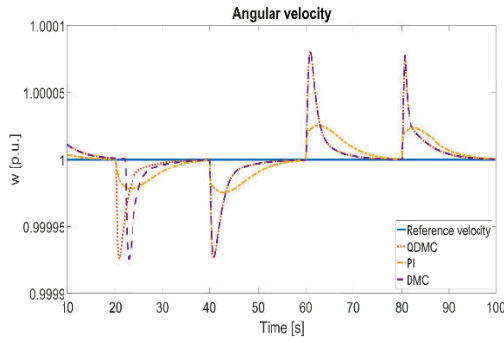


Fig. 4. The angular velocity reference level its realization with PI, DMC and QDMC controllers.

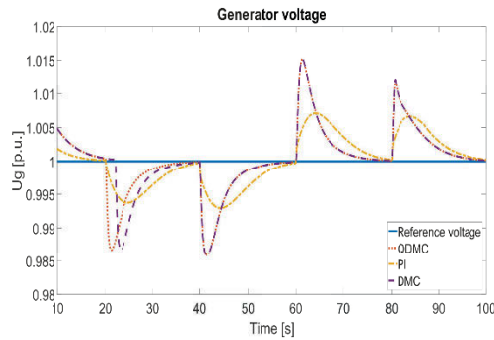


Fig. 5. The generator's terminal voltage reference level its realization with PI, DMC and QDMC controllers.

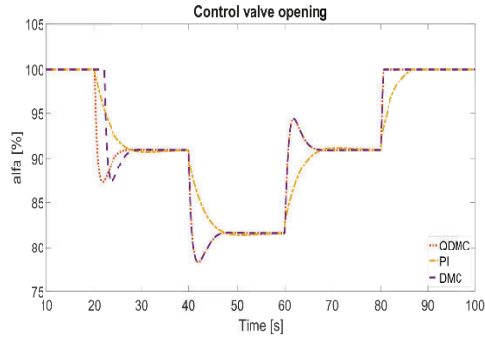


Fig. 6. The control valve opening degree (manipulated variable) generated with PI, DMC and QDMC controllers.

Table 1. The controllers performance index

	ISE P_g	ISE U_g	ISE ω
PI	0.2659	0.0045	$5.2e^{-8}$
DMC	0.1247	0.0077	$1.2e^{-7}$
QDMC	0.0650	0.0077	$1.2e^{-7}$

the PI controller. However, the PI controller has twice time smaller performance index in the case of angular velocity and generator's voltage reference trajectories realizations. In Figs. 3-5 can be seen that the settling time for DMC and QDMC predictive controllers is shorter then for the PI controller, but the bigger overshoots are observed.

6 Summary

Paper proposed an approach to design a model predictive controller of a nuclear power plant steam turbine. Three different control algorithms were compared: the typical PI controller, the DMC controller based on the algebraic control law without considering the constraints of the process, and the QDMC controller which at the each time sampling instant solving on-line an optimization problem with constrains for manipulated and controlled variables, expressed as linear inequalities. In both cases the recursive least squares (RLS) algorithm was used to obtain accurate simplified turbo-generator set model parameters, in order to determine appropriate unit step-response model for DMC and QDMC controllers. Simulation test shows that the QDMC controller realize the reference trajectories of the effective power and the angular velocity, and damp the electromechanical oscillations with satisfactory quality in comparison to the typical PI and DMC controllers. The authors ongoing work focuses on the improving control quality of the turbo-generator set, for example by application the model

predictive controller for the synchronous generator, and implementation a functional mechanism of cooperation between these two controllers.

References

1. Camacho E.F., Alba C. B.: Model Predictive Control. Springer Science & Business Media (2013)
2. Hakvoort, R. G.: System identification for robust process control : nominal models and error bounds, Technische Univ. Delft, Delft (1994)
3. IEEE Report: Dynamic Models for Steam and Hydro Turbines in Power System Studies. IEEE Transactions on Power Apparatus and Systems, Vol. PAS-92, Issue 6, pp. 1904-1915 (1973)
4. Imieliński A.: Mathematical model of synchronous generator for full-scope simulator. Gdańsk University of Technology, Faculty of Electrical and Control Engineering, Gdańsk, Poland (1987) - in polish, unpublished
5. Kulkowski K., Kobylarz A., Grochowski M., Duzinkiewicz K.: Dynamic model of nuclear power plant steam turbine. Arch. Control Sci., Vol. 25, No. 1, pp. 65-86 (2015)
6. Lipo T.A.: Analysis of Synchronous Machines. CRC Press (2012)
7. Loo C., Vanfretti L., Liceaga-Castro E., Enrique Acha E.: Synchronous Generators Modeling and Control Using the Framework of Individual Channel Analysis and Design: Part 1. International Journal of Emerging Electric Power Systems, Vol. 8, Issue 5, (2007)
8. Perycz S., Próchnicki W.: The mathematical model of a nuclear power plant VVER block steam turbine allowing to study transient processes with $w = var$. Gdańsk University of Technology, Faculty of Electrical and Control Engineering, Gdańsk, Poland (1989) - in polish, unpublished
9. Power System Dynamic Performance Committee, Power System Stability Subcommittee, Task Force on Turbine-Governor Modeling: Dynamic Models for Turbine-Governors in Power System Studies. IEEE Power & Energy Society (2013)
10. Rossiter J.A.: Model-Based Predictive Control: A Practical Approach. CRC Press, Boca Raton, FL (2013)
11. Seybold L., Witczak M., Majdzik P., Stetter R.: Towards robust predictive fault-tolerant control for a battery assembly system. Int. J. Appl. Math. Comput. Sci., vol. 25, no. 4, pp. 849862 (2015)
12. Sokolski P., Rutkowski T.A., Duzinkiewicz K.: Simplified, multiregional fuzzy model of a nuclear power plant steam turbine. IEEE, 2016 21ST International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 379-384 (2016)
13. Sokolski P., Rutkowski T.A., Duzinkiewicz K.: The excitation controller with gain scheduling mechanism for synchronous generator control. IEEE, 2015 20TH International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 23-28 (2015)
14. Sokółski P., Kobylarz A., Kulkowski K., Duzinkiewicz K., Rutkowski T.A., Grochowski M.: Advanced control structures of turbo generator system of nuclear power plant. Acta Energetica, Vol. 3, pp. 83-96 (2015)
15. Tatjewski P.: Advanced Control of Industrial Processes. Springer London (2007)
16. Lawryńczuk M.: Nonlinear state-space predictive control with on-line linearisation and state estimation. Int. J. Appl. Math. Comput. Sci., vol. 25, no. 4, pp. 833847 (2015)

Comparison of the feedforward control design methods for nonminimum-phase LTI SISO systems with application to the double-drum coiling machine

Krzysztof Łakomy, Maciej Marcin Michałek

Institute of Automation and Robotics (IAR), Poznań University of Technology (PUT),
Piotrowo 3A, 60-965 Poznań, Poland,

krzysztof.pi.lakomy@doctorate.put.poznan.pl, maciej.michalek@put.poznan.pl

Abstract. Due to the presence of right-half plane zeroes in the transfer function of a nonminimum-phase plant, conventionally designed feedforward controller becomes unstable. Main focus of this article is put on a comparison of the *Extended Bandwidth Zero Phase Error* and recently proposed *fixed-structure approximate inverse* feedforward control design methods applicable to the nonminimum-phase systems. Comparison of the techniques presented in the paper is based on the frequency analysis of a closed-loop error transfer function, followed by simulation examples and experimental validation on a laboratory setup with a double-drum coiling machine. Both simulation and experimental results showed the superiority of trajectory tracking obtained by the fixed-structure approximate inverse method under the assumed conditions.

1 Introduction

In case of minimum-phase systems, model-inverse based *two degrees of freedom* (2DOF) control structure, including feedback loop and *feedforward* (FF) controller, is a basic commonly used method of improving the quality of trajectory tracking. For *nonminimum-phase* (NMP) system (which characteristic feature is the presence of at least one zero in the *right-half plane* (RHP) of its *transfer function* (TF), see [5]), the model-inverse feedforward controller becomes unstable. Many techniques capable of providing a stable approximation of the inverse TF were presented in the literature. In this article, approximation methods are divided into two groups. First group contains classical methods, that is, the *nonminimum-phase zero ignore* (NZI) tracking control, *zero magnitude error* (ZME) tracking control, and *zero phase error* (ZPE) tracking control described in [1], [2], [8], and [7]. Approximation of the inverse plant dynamics computed with these techniques depends only on a form of the plant TF parameters. Feedforward controllers obtained with the methods belonging to the second group, i.e., *Extended Bandwidth Zero Phase Error* (EBZPE) tracking control [9], [3],

* This work was supported by the statutory grant No. 09/93/DSPB/0611.

and recently proposed *fixed-structure approximate inverse* (XAI) method [6], depend also on the values of specific design parameters.

In the article, all of the aforementioned methods are compared using formal analysis, simulation examples and experimental validation made on the model of a double-drum coiling machine. The comparison between XAI and classical methods have been presented before in [6], while the main contribution of this paper is the additional comparison of EBZPE and XAI methods.

2 Preliminary information

2.1 System description

The general *Linear Time-Invariant, Single-Input, Single Output* (LTI SISO) plant with input $u(t)$ and output $y(t)$ can be described by the transfer function

$$G(s) \triangleq G^*(s)e^{-sT_0} = \frac{d_{m_d}s^{m_d} + d_{m_d-1}s^{m_d-1} + \dots + d_1s + d_0}{c_{n_d}s^{n_d} + c_{n_d-1}s^{n_d-1} + \dots + c_1s + c_0}e^{-sT_0}, \quad (1)$$

where $d_i \in \mathbb{R}, i \in \{1, \dots, m_d\}$, $c_j \in \mathbb{R}, j \in \{1, \dots, n_d\}$ are the parameters of TF, while $T_0 \geq 0 \in \mathbb{R}$ is a time delay.

Assumption 1 *Transfer function $G^*(s)$ is proper or strictly proper ($m_d \leq n_d$), values of coefficients c_i, d_i and T_0 are perfectly known, while polynomials in the numerator and the denominator does not have any common factors.*

All of the selected FF control design methods require system representation in the form of a rational TF, so the time-delay term has to be approximated, e.g. by using Taylor series approximation

$$e^{-sT_0} \approx \frac{1}{1 + z_1s + z_2s^2 + \dots + z_\nu s^\nu}, \quad z_i = \frac{T_0^i}{i!}, \nu \in \mathbb{N}, \quad (2)$$

where ν determines an approximation degree. Using (2), an approximation of the general form of LTI SISO system (1) can be represented by

$$G(s) \approx \frac{b_ms^m + b_{m-1}s^{m-1} + \dots + b_1s + 1}{a_ns^n + a_{n-1}s^{n-1} + \dots + a_1s + a_0} = \frac{B^p(s)B^n(s)}{A(s)}, \quad (3)$$

where $B^p(s) : B^p(s_p) = 0 \Rightarrow \Re(s_p) > 0$ is a part of the numerator containing zeroes from RHP, while $B^n(s) : B^n(s_n) = 0 \Rightarrow \Re(s_n) \leq 0$ is a part containing zeroes from the open *left-half plane* (LHP).

Upon (3), it can be easily shown that the inverse plant dynamics $G^{-1}(s) = \frac{A(s)}{B^p(s)B^n(s)}$ is unstable for the NMP system - polynomial $B^p(s)$ in the denominator introduces RHP poles into $G^{-1}(s)$ transfer function.

Block diagram of the considered 2DOF control structure used in further analysis is presented in Fig. 1, where y_d denotes a reference trajectory, $G_{FF}(s)$ is a feedforward controller, $G_R(s)$ is a stabilizing controller, y is an output of the system, while e stands for a tracking error defined as

$$e(t) \triangleq y_d(t) - y(t). \quad (4)$$

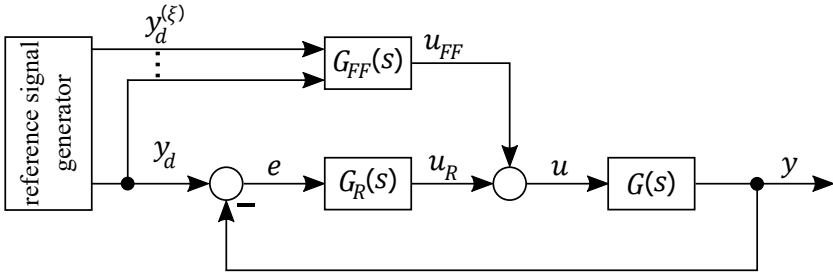


Fig. 1: Block diagram of the 2DOF control structure

Assumption 2 *The reference trajectory $y_d(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$ is a bounded and sufficiently smooth function of class C^ξ with sufficiently large $\xi > 1$, and signals $y_d(t), y_d^{(1)}(t), \dots, y_d^{(\xi)}(t)$ are exactly known for any time instant $t \geq 0$.*

Based on the tracking error definition (4), let us define the error transfer function

$$G_E(s) \triangleq \frac{E(s)}{Y_d(s)} = \frac{1 - G(s)G_{FF}(s)}{1 + G(s)G_R(s)} = S(s)[1 - G(s)G_{FF}(s)] = S(s)\Gamma(s) \quad (5)$$

where $S(s)$ is called a sensitivity transfer function, while $\Gamma(s)$ is the so-called feedforward mismatch transfer function introduced in [6]. The tracking control performance can be improved by introducing to the system a FF controller with the best possible stable approximation to the inverse plant dynamics $G^{-1}(s)$. When $G_{FF}(s) \triangleq G^{-1}(s)$, the mismatch function $\Gamma(s) \equiv 0$. If $G^{-1}(s)$ is stable, than one can obtain a perfect tracking of the reference trajectory $y_d(t)$ satisfying Assumption 2.

2.2 Feedforward control design methods

Inverse plant TF approximations obtained with the classical techniques (NZI, ZME, ZPE) depend only on the parameters of the approximated TF (3). FF controller obtained with NZI method is defined by $G_{FF}^{NZI}(s) \triangleq \frac{A(s)}{B^n(s)B^p(0)}$ and simply ignores the influence of RHP zeros in the plant dynamics. Application of ZME method results in the FF controller in a form $G_{FF}^{ZME}(s) \triangleq \frac{A(s)}{B^n(s)B^p(-s)}$, which is designed in such a way, that the magnitude of an inverse plant model and its approximation are equivalent in the frequency domain. Feedforward controller $G_{FF}^{ZPE}(s) = \frac{A(s)B^p(-s)}{B^n(s)[B^p(0)]^2}$ obtained with the ZPE method provides an approximation with phase changes equivalent in the frequency domain to the inverse plant TF.

Second group of the FF control design methods is dependent not only on the form of a plant TF, but also on the specific design parameters. FF controller obtained with EBZPE method, see [3], has the form

$$G_{FF}^{EBZPE}(s) = G_{FF}^{ZPE}(s) \sum_{k=0}^q [1 - G(s)G_{FF}^{ZPE}(s)]^k, \quad q \in \mathbb{N}, \quad (6)$$

where q is a prescribed design parameter, that affects the accuracy of an inverse TF approximation. Considering the limit case $q \rightarrow \infty$, one can write that

$$\begin{aligned} \lim_{q \rightarrow \infty} G_{FF}^{EBZPE}(s) &= G_{FF}^{ZPE}(s) \lim_{q \rightarrow \infty} \sum_{k=0}^q [1 - G_{FF}^{ZPE}(s)G(s)]^k \\ &= G_{FF}^{ZPE}(s) \frac{1}{1 - (1 - G_{FF}^{ZPE}(s)G(s))} = G^{-1}(s). \end{aligned} \quad (7)$$

Geometric series included under the limit operation in (7) converge to a finite value, when the magnitude of its common ratio satisfy $|1 - G_{FF}^{ZPE}(s)G(s)| < 1$.

The fixed-structure feedforward control law (XAI)

$$G_{FF}^{XAI}(s) = P(s) = \sum_{k=0}^{\mu} p_k s^k, \quad \mu \in \mathbb{N}, p_k \in \mathbb{R}, \quad (8)$$

depends on the design parameter μ . The fixed-structure FF control law is called *corrected-approximate-inverse* (CAI) controller, when $\mu = n$ (where n is a degree of denominator of (3)), and the *extended-approximate-inverse* (EAI) controller, when $\mu > n$. Design parameters p_k have the values determined as follows [6]:

$$\begin{cases} p_0 \triangleq a_0, \\ p_k \triangleq a_k - \sum_{i=1}^k b_i p_{k-i}, \text{ for } k \in \{1, \dots, n\}, \\ p_k \triangleq -\sum_{i=1}^k b_i p_{k-i}, \text{ for } k \in \{n+1, \dots, \mu\}. \end{cases} \quad (9)$$

Unlike the other feedforward controllers presented before, $G_{FF}^{XAI}(s)$ does not have a complex polynomial in the denominator (fixed-structure control law), leading to simplicity of its implementation, especially on digital controllers running physical control systems.

Based on the form of the feedforward mismatch function $\Gamma(s)$ introduced in (5), one is going now to compare achievable tracking control accuracy in the 2DOF control system from Fig. 1 for the selected feedforward control design methods.

3 Comparison of presented methods

Following [6], feedforward mismatch functions (5) derived for NZI, ZME, and ZPE methods get the forms

$$\Gamma^{NZI}(s) = 1 - \frac{B^p(s)}{|B^p(0)|} = -\beta_\gamma s^\gamma - \dots - \beta_1 s, \quad (10)$$

$$\Gamma^{ZME}(s) = 1 - \frac{B^p(s)}{B^p(-s)} = -\frac{[(-1)^\gamma - 1]\beta_\gamma s^\gamma - \dots - 2\beta_1 s}{(-1)^\gamma \beta_\gamma s^\gamma + \dots - \beta_1 s + 1}, \quad (11)$$

$$\Gamma^{ZPE}(s) = 1 - \frac{B^p(s)B^p(-s)}{|B^p(0)|^2} = -\bar{\beta}_{2\gamma} s^{2\gamma} - \dots - \bar{\beta}_2 s^2, \quad (12)$$

where $\beta_i, \bar{\beta}_i$ are some scalar coefficients, while γ stands for a number of positive zeros in the plant TF. Mismatch functions $\Gamma^{NZI}(s)$ and $\Gamma^{ZPE}(s)$ have an easy to analyze polynomial form, additionally $\Gamma^{ZPE}(s)$ includes only even powers of operator s . $\Gamma^{ZME}(s)$ has a complex polynomial in the denominator, and its numerator includes only odd powers of operator s .

Derivation of the feedforward mismatch function for the EBZPE method can be explained as follows:

$$\begin{aligned} \Gamma^{EBZPE}(s) &= 1 - G(s)G_{FF}^{ZPE}(s) \sum_{k=0}^q [\Gamma^{ZPE}(s)]^k \\ &= 1 - (1 - \Gamma^{ZPE}(s)) \sum_{k=0}^q [\Gamma^{ZPE}(s)]^k \\ &= 1 - [\bar{\beta}_{2\gamma} s^{2\gamma} + \dots + \bar{\beta}_2 s^2 + 1] \cdot [\hat{\beta}_{2q\gamma} s^{2q\gamma} + \dots + 1] \\ &= -\tilde{\beta}_{2(q+1)\gamma} s^{2(q+1)\gamma} - \tilde{\beta}_{2(q+1)\gamma-2} s^{2(q+1)\gamma-2} - \dots - \tilde{\beta}_2 s^2, \quad (13) \end{aligned}$$

Mismatch function (13) is a complex polynomial of the operator s with a degree $2(q+1)\gamma$ dependent on the design parameter q , and similarly to the ZPE method includes only even powers of operator s .

Recalling the results presented in [6], a mismatch function for the XAI method with p_k coefficients calculated according to (9) has the form

$$\Gamma^{XAI}(s) = 1 - G(s)G_{FF}^{XAI}(s) = \frac{A(s) - B(s)P(s)}{A(s)} = \frac{W(s)s^{\mu+1}}{A(s)}, \quad (14)$$

where $W(s) \triangleq w_{m-1}s^{m-1} + \dots + w_0$ is a resultant complex polynomial. Function $\Gamma^{XAI}(s)$ has the minimal degree of s operator equal $\mu+1$, what allows its description in a logarithmic scale as $\text{Lm}^{XAI}(\omega) \triangleq 20\log|\Gamma^{XAI}(j\omega)| = 20(\mu+1)\log(\omega) + 20\log|W(j\omega)| - 20\log|A(j\omega)|$. One can approximate a slope of $\text{Lm}^{XAI}(\omega)$ in a low frequency range as

$$N^{XAI}(\omega) \triangleq \frac{d \text{Lm}^{XAI}(\omega)}{d \log \omega} \approx 20(\mu - i + 1) \text{ dB/dec}, \quad (15)$$

where i is a number of poles equal to zero in the plant TF. Slopes for the other methods can be conservatively evaluated by using (10)-(13), that is,

$$N^{NZI}(\omega) \lesssim 20\gamma \text{ dB/dec}, \quad (16)$$

$$N^{ZME}(\omega) \lesssim 20\gamma \text{ dB/dec}, \quad (17)$$

$$N^{ZPE}(\omega) \lesssim 40\gamma \text{ dB/dec}, \quad (18)$$

$$N^{EBZPE}(\omega) \lesssim 40(q+1)\gamma \text{ dB/dec}. \quad (19)$$

By comparing (15) with (16) - (19), one may conclude for low frequency range

$$N^{XAI} > N^{NZI}, \quad \text{if } \mu > \gamma + i - 1, \quad (20)$$

$$N^{XAI} > N^{ZME}, \quad \text{if } \mu > \gamma + i - 1, \quad (21)$$

$$N^{XAI} > N^{ZPE}, \quad \text{if } \mu > 2\gamma + i - 1, \quad (22)$$

$$N^{XAI} > N^{EBZPE}, \quad \text{if } \mu > 2(q+1)\gamma + i - 1. \quad (23)$$

Based on the expressions (20) - (22), one can conclude that there exists sufficiently large μ that guarantees N^{XAI} to be larger than the slope of any other $\Gamma(s)$ obtained with the considered methods (in a low frequency range). The comparison between XAI and other selected methods is possible due to the determination of a slope of $\Gamma^{XAI}(s)$ in (15) for the specific values of μ . Although slopes of $\Gamma(s)$ obtained for the EBZPE and classical methods, determined by (16) - (19), are described with the inequalities - on the basis of (7) and satisfying $|1 - G(s)G_{FF}^{ZPE}(s)| < 1$ we can conclude that $N^{EBZPE} \geq N^{ZPE}$ for any q value.

4 Numerical examples

Numerical comparison of the discussed feedforward controllers has been made using two exemplary plant transfer functions, denoted as $G_a(s) = \frac{-0.1s+1}{s^2+s}$ and $G_b(s) = \frac{-0.4s+1}{0.3s^2+0.8s+1.5}e^{-0.5s}$. Transfer function $G_a(s)$ concerns a strictly proper nonminimum-phase second order integrating plant with one positive zero $z_1 = 10$ and two poles $s_1 = -1$ and $s_2 = 0$. $G_b(s)$ is a nonminimum-phase second order TF with positive zero $z_1 = 2.5$, two complex poles $s_1 = -1.333 + 1.7951j$, $s_2 = -1.333 - 1.7951j$, and time delay $T_0 = 0.5$ s. Taylor series approximation ($\nu = 2$) of the time delay part of $G_b(s)$ results in $\hat{G}_b(s) = \frac{-0.4s+1}{0.0375s^4+0.25s^3+0.8875s^2+1.55s+1.5}$.

Feedforward control law for the EBZPE method was derived in both cases for the design parameter $q = 1$. In case of the XAI methods, the maximum value of μ parameter was selected so that the degree of a complex polynomial $G_{FF}^{XAI}(s)$ is equal to the degree of a polynomial in the numerator of $G_{FF}^{EBZPE}(s)$ ($\mu = 5$ for $G_a(s)$, $\mu = 7$ for $\hat{G}_b(s)$). Equal degrees of the aforementioned polynomials guarantee the use of a reference trajectory and its derivatives up to the same degree by the controllers obtained with both methods.

Feedback control is realized by the stabilizing P-type controller $G_R(s) = k_p$ with the gain $k_p = 0.25$ for $G_a(s)$ and $k_p = 1.0$ for $G_b(s)$. For the simulation purposes, a reference trajectory was set to be a sinusoidal function $y_d(t) = A_d \sin(\omega_d t)$ with amplitude $A_d = 1.0$ and frequency $\omega_d = 0.31622$ rad/s.

The Bode magnitude plot presented in Fig. 2 concerns the $G_a(s)$ case. By increasing the value of μ parameter for the FF controller obtained with XAI method, the error TF magnitude is gradually reduced. Referring to (20) - (23), XAI controller for $\mu = 5$ ensures lower steady state tracking errors compared to the ones obtained with other selected FF design methods. Exemplary time-plot of the control error $|e|$ is presented in the logarithmic scale in Fig. 3. The reference frequency ω_d is marked as a dashed vertical line in Fig. 2 and Fig. 4.

Bode magnitude plot of the error transfer function for $G_b(s)$ is presented in Fig. 4. Characteristics obtained for EBZPE and XAI methods (for all μ) converge to a common asymptote in the range of low frequencies. The reason of such convergence is the model uncertainty caused by a Taylor approximation of the time delay part. Due to these uncertainties, further extensions of μ and q parameters are not providing any improvement to the accuracy of control system (Fig. 4). It is however possible to improve tracking quality by increasing the Taylor approximation degree ν .

5 Experimental validation

In order to verify the trajectory tracking performance in the presence of measurement noise and parametric uncertainty of a plant model, described feedforward controllers have been implemented in a fast-prototyping system with a double-drum coiling machine. Exchange of signals between the machine digital controller and the PC equipped with VisSim+RealTimePRO environment was

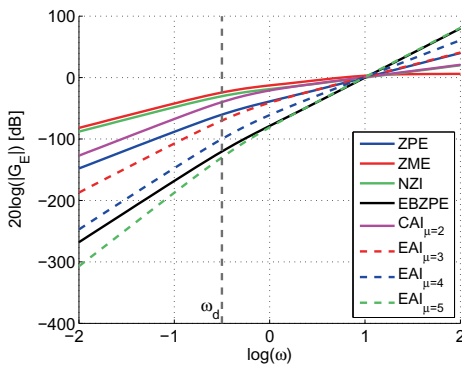


Fig. 2: $G_a(s)$: Bode magnitude plot of the error transfer function.

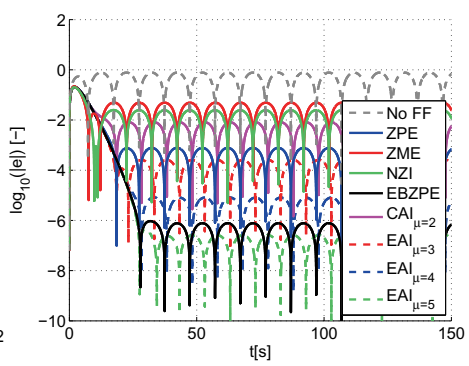


Fig. 3: $G_a(s)$: Absolute tracking error in a logarithmic scale.

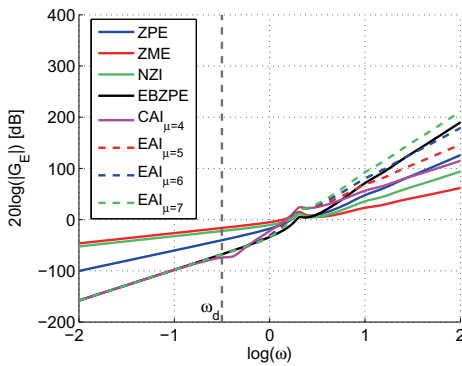


Fig. 4: $G_b(s)$: Bode magnitude plot of the error transfer function.

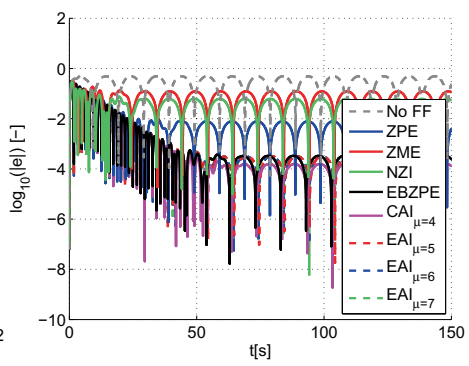


Fig. 5: $G_b(s)$: Absolute tracking error in a logarithmic scale.

realized through the I/O card PCI-DAS 1602/12 in a way presented in Fig. 6. Input signal u and output y are voltage analogue signals that can be interpreted as a scaled reference velocities for the drums and actual position of the weight attached to the coiled material, respectively. Sampling time interval was set to $T_s = 0.01$ s.

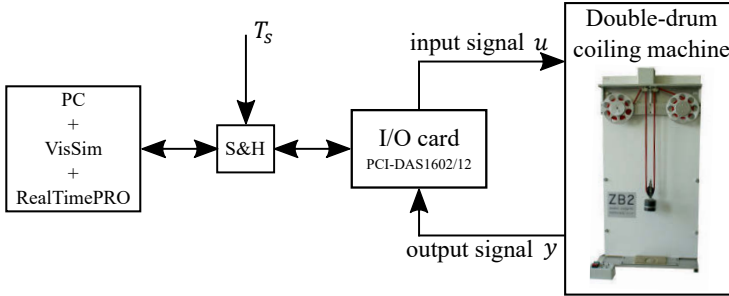


Fig. 6: Functional schema of the double-drum coiling machine control system.

The TFs obtained for particular drums, describing the relation of weight position to the drum rotational speed have the form $G_1(s) = \frac{\gamma_1 r_1 \eta_1 G_{F1}(s)}{2\gamma_2 s}$, and $G_2(s) = \frac{\gamma_1 r_2 \eta_2 G_{F2}(s)}{2\gamma_2 s}$, where γ_1, γ_2 are positive constants connected with the input/output signal scaling, $\eta_1 = 1 : 21$, $\eta_2 = 1 : 51$ are the gear ratio values of the drums actuators, while r_1 and r_2 correspond to the radii of coiled material on each drum. $G_{F1}(s) = \frac{k_1}{sT_1 + 1}$ and $G_{F2}(s) = \frac{k_2}{sT_2 + 1}$ are the low-pass filters with gains k_1, k_2 and time constants T_1, T_2 . Resultant TF of the double-drum coiling machine takes into account the movement of both drums and has the form

$$G(s) = \frac{Y(s)}{U(s)} \triangleq G_2(s) - G_1(s) = \frac{\gamma_1}{2\gamma_2 s} [r_2 \eta_2 G_{F2}(s) - r_1 \eta_1 G_{F1}(s)]. \quad (24)$$

Rewriting (24) in a more detailed manner, the plant TF takes the form

$$\begin{aligned} G(s) &= \frac{\gamma_1 k_2 \eta_2 r_2 - \gamma_1 k_1 \eta_1 r_1}{2\gamma_2} \cdot \frac{\frac{k_2 \eta_2 r_2 T_1 - k_1 \eta_1 r_1 T_2}{k_2 \eta_2 r_2 - k_1 \eta_1 r_1} s + 1}{s(sT_1 + 1)(sT_2 + 1)} = \\ &= k \cdot \frac{sT + 1}{s(sT_1 + 1)(sT_2 + 1)}, \end{aligned} \quad (25)$$

where k is a resultant gain, while $s = -1/T$ is a value of the plant zero. Setting the the gain values to be $k_1 = k_2 = 1$, and assuming $r_1 \approx r_2$ - inequality $T_1 > \frac{51}{21} T_2$ has to be satisfied to assure the presence of RHP zero in the coiling machine TF. It can be obtained for the exemplary pair of time constants $T_1 = 0.2$ s and $T_2 = 0.01$ s which allow one to clearly illustrate the tracking quality differences for selected FF methods. Parameters of a plant model have been identified using

SVF-RLS $_{\lambda}$ method, see [4], leading to the model

$$\hat{G}(s) = 0.35058 \cdot \frac{-0.09490s + 1}{s(0.2s + 1)(0.01s + 1)}. \quad (26)$$

Coiling machine TF has one positive zero $z_1 = 10.537$ and three poles $s_1 = -100$, $s_2 = -5$, and $s_3 = 0$. During the experiment, control system was excited by the reference trajectory $y_d(t) = 4 + 0.25 \sin(7.07t)$, while the trajectory tracking performance was specified with *Integral of Absolute value of Error* (IAE) criterion $J_e \triangleq \int_{t_1}^{t_2} |e(t)| dt$.

Fig. 7 presents Bode magnitude plot of the error transfer function obtained for the model (26). Vertical dashed line marks the reference trajectory frequency used in the experiment. The choice of a relatively high frequency is caused by the presence of a measurement noise. Comparison of the tracking performance in a low frequency range was impossible, because the values of the tracking error were much lower than the measurement noise level, what made the values of IAE criterion approximately equal for all of the considered FF design methods.

Table 1 contains the values of IAE criterion calculated in the steady conditions, between $t_1 = 5$ s and $t_2 = 16$ s. According to (20)-(23), the FF controller obtained with XAI method for $\mu > 4$ guarantees better control accuracy than the controllers based on the other methods. Relative tracking quality between considered FF control design methods is slightly different than the one presented in Fig. 7. Comparing tracking quality based on the values of IAE criterion, and the ones presented on the Bode magnitude diagrams, one can see that the tracking quality obtained with NZI method was better than CAI $_{\mu=3}$, and ZPE overcame EAI $_{\mu=4}$. It is caused by the presence of a measurement noise, and close location of NZI/CAI $_{\mu=3}$ and ZPE/EAI $_{\mu=4}$ characteristics on the Bode plot.

6 Conclusions

The results described in this paper complements the considerations presented in [6]. The main contribution of this article is the comparison of EBZPE with other selected FF design methods, including its experimental verification on the specific physical system - double-drum coiling machine. It has been shown that for the same reference trajectory, while degrees of the numerators of $G_{FF}^{XAI}(s)$ and $G_{FF}^{EBZPE}(s)$ are equal, the fixed-structure approximate inverse method ensures more accurate tracking relative to Extended Bandwidth Zero Phase Error tracking control method. For a perfectly known transfer functions without a time delay term, increasing values of the design parameters μ and q gradually increase tracking accuracy of, respectively, XAI and EBZPE method. In the presence of a measurement noise and/or parametric uncertainty of the plant transfer function, the use of EBZPE and XAI feedforward controllers with large values of q and μ seem unjustifiable, due to practical limitations of improving a tracking quality.

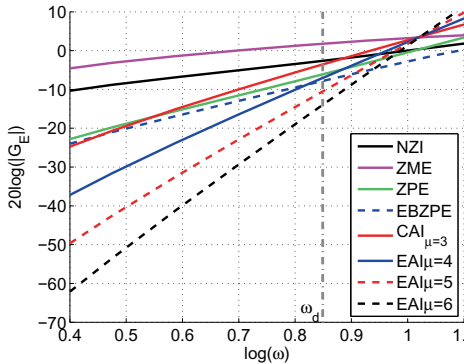


Fig. 7: $\hat{G}(s)$: Bode magnitude plot of the error transfer function

Table 1: Values of IAE criterion J_e for selected FF design methods

Method	J_e
<i>NFF</i>	1.8805
<i>NZI</i>	1.2394
<i>ZME</i>	2.3562
<i>ZPE</i>	0.9694
<i>EBZPE</i>	0.8679
<i>CAI</i> _{$\mu=3$}	1.3321
<i>EAI</i> _{$\mu=4$}	1.0386
<i>EAI</i> _{$\mu=5$}	0.7978
<i>EAI</i> _{$\mu=6$}	0.6759

References

1. J.A. Butterworth and D.Y. Abramovitch. Analysis and comparison of three discrete-time feedforward model-inverse control techniques for nonminimum-phase systems. *Mechatronics*, 22:577–587, 2012.
2. J.A. Butterworth, L.Y. Pao, and D.Y. Abramovitch. The effect of nonminimum-phase zero locations on the performance of feedforward model-inverse control techniques in discrete-time systems. *2008 American Control Conference*, pages 2696–2702, Seattle, Washington, USA, June 11-13 2008.
3. P. H. Chang and G. R. Cho. Enhanced feedforward control of non-minimum phase systems for tracking predefined trajectory. *Int J Control*, 83(12):2440–2452, 2010.
4. Editors: H. Garnier and L. Wang. *Identification of Continuous-time Models from Sampled Data, Chapter 1*. Springer, 2008.
5. J. B. Hoagg and D.S. Bernstein. Nonminimum-phase zeros, much to do about nothing. *IEEE Control Systems Magazine*, pages 45–57, June 2007.
6. M. M. Michałek. Fixed-structure feedforward control law for minimum- and nonminimum-phase LTI SISO systems. *IEEE Trans. Control Syst. Technol.*, 24(4):1382–1393, 2016.
7. H. Park, P.H. Chang, and D.Y. Lee. Trajectory planning for the tracking control of systems with unstable zeros. *Mechatronics*, 13:127–139, 2003.
8. B.P. Rigney, L.Y. Pao, and D.A. Lawrence. Nonminimum phase dynamic inversion for settle time applications. *IEEE Trans. Control Syst. Technol.*, 17(5):989–1005, 2009.
9. D. E. Torfs, R. Vuerinckx, J. Swevers, and J. Schoukens. Comparison of two feedforward design methods aiming at accurate trajectory tracking of the end point of a flexible robot arm. *IEEE Trans. Control Syst. Technol.*, 6(1):2–14, January 1998.

Camera in the control loop – methods and selected industrial applications

Ewaryst Rafajłowicz and Wojciech Rafajłowicz

Ewaryst Rafajłowicz and Wojciech Rafajłowicz are with the Faculty of Electronics,
Wrocław University of Technology, Wrocław , Poland.
ewaryst.rafajlowicz@pwr.wroc.pl

Abstract. Our aim is to discuss briefly methods of using cameras in control systems. Then, we concentrate on a new approach to iterative learning control (ILC) for nonlinear repetitive production processes. Finally, we propose the methodology of applying a camera for tuning ILC and illustrate it by the example of a multilayer system for laser power control in selective laser melting (SLM).

Keywords: Camera in the loop, iterative learning control, selective laser melting

1 Introduction

The idea of applying a camera in a control loop has been present in the literature for many years. In most of its applications, especially at the initial phase of development, a camera played the role of a measurement device or a sensor. More advanced applications of cameras in the loop for regulating industrial processes are described rather rarely (see [2], [15], [6], [6], [9] for several examples). Much more has been done in the area of computer vision quality monitoring (see [3] and extensive bibliography cited therein).

One can roughly classify the methods of using cameras for control of industrial processes into three groups, namely:

- G1)** image-driven control systems, which are completely or partly model free,
- G2)** model-based control systems that contain (a) camera(s) as an important part, i.e., an inference from a sequence of images is used in the control system,
- G3)** immediate usage of current images after their simple processing.

In this paper we briefly discuss G1) group of methods (Section 2). Our main focus is on G2) group with the particular emphasis on the repetitive processes that are dominate in modern day production processes. Within this group we propose an iterative learning of optimal control (ILOC) approach for nonlinear systems (Section 3) that is applicable to selective laser melting (SLM) processes (Section 4), which are expected to be a technology of the future. We skip discussion of the G3) group of methods, since they are simpler and well described in the literature, especially on robotics.

2 Image-driven control

A direct, model free, image driven control (IDC) seems to be a new idea. One can consider at least the following two ways of its implementation:

IDC1) features of a process to be controlled are extracted from images, then a decision unit is the subject of learning and finally, the decision unit with a feature extractor is fed by new images and control actions are provided as its outputs.

IDC2) as above, but the feature extraction is not made, instead the decision unit is learnt directly from whole images and this unit is also fed by whole images that serve as a base for decision making.

IDC1) approach was presented in [12]. One of the possible ways of implementing IDC2) is proposed in [13]. The images of flames of an industrial gas burner are clustered into groups using a (dis-)similarity measure between images. Then, decisions on increasing or decreasing the air supply rate to the burner are attached to each cluster. After this semi-supervised learning phase a decision unit is ready for decision making. When a new image is acquired, it is classified to one of the clusters and the corresponding decision on the air supply rate is provided as the output.

3 ILC algorithm for nonlinear repetitive processes

The theory and practice of ILC for repetitive processes has a relatively long history (see [16] and [10] for monographs of the field and the bibliography cited therein). The closest approaches to the one presented here can be found in [11], [4]. There are however also important differences, the main being in the control quality criterion. The approach considered in this paper extends the one proposed in [14] to nonlinear repetitive systems.

3.1 Description of a repetitive process

Consider the following repetitive process

$$\dot{x}_n(t) = f(x_n(t), u_n(t), a), \quad t \in (0, T], \quad x_n(0) = x_0, \quad n = 1, 2, \dots \quad (1)$$

where $x_n(t) = [x_n^{(1)}(t), x_n^{(2)}(t), \dots, x_n^{(d)}(t)]^{tr}$ is $d \times 1$ the state vector at time t and pass n . For simplicity of the exposition, we shall assume that $u_n(t)$ is a scalar input signal at time t and n -th pass. By $a \in R^m$ we denote the vector of parameters. We know their nominal values a^0 , but we would like to design a control system that is – to some extent – robust to their changes. We also assume that the initial condition $x_n(0) = x_0$ is the same for each pass. We assume that function $f : R^d \times R \times R^m \rightarrow R^d$ is continuously differentiable w.r.t. all of its arguments.

T denotes the length of the pass. We shall assume that T is constant and known. It is convenient to define the optimality criterion as a function $Q : R^d \rightarrow$

R that operates on the final state only, i.e., $Q(x_n(T))$, where Q is nonnegative and continuously differentiable. This is the so called Mayer functional (see, e.g., [1] page 10).

Clearly, one can easily incorporate the integral quality criterion (the Bolza functional) $\int_0^T q(x_n(t), u_n(t)) dt$ into this framework by defining an additional state variable.

An ILOC procedure that is proposed in this paper attempts to find a minimizer of $Q(\cdot)$ by proper selection of an operator Ψ which provides a new control signal along the pass $u_n(\cdot) = \Psi(u_{n-1}(\cdot), x_{n-1}(\cdot))$, based on the past pass of the control signal and state.

3.2 Optimal control problem for one pass and the Frechet derivative

Let us consider one, typical pass of the process (1), assuming that the parameters a have their nominal values $a = a^0$. Define $J(u(\cdot))$ as the quality criterion of a given input signal $u(\cdot)$, which is equal to as $Q(x(T))$, in which the dependence of $x(T)$ on $u(\cdot)$ is implicitly defined through the solution of the state equations:

$$\dot{x}(t) = f(x(t), u(t), a^0), \quad x(0) = x_0 \tag{2}$$

for a given x_0 .

For one pass of the process we formulate the following problem: find a piecewise continuous function $u^*(t)$ and the corresponding trajectory $x^*(t)$, $t \in (0, T]$ for which

$$\min_u J(u(\cdot)), \quad \text{s.t.} \quad \dot{x}(t) = f(x(t), u(t), a^0), \quad x(0) = x_0. \tag{3}$$

is attained. For simplicity of the exposition, we do not formulate explicitly constraints neither on control signals nor on state variables, assuming that they are included (e.g., as penalties) into Q .

Assumption 1 *We assume that there exists a certain (possibly very large) closed and bounded domain $\mathcal{D} \subset R^d$ such that $x(t) \in \mathcal{D}$ for all $t \in [0, T]$. Similarly, values $u(t)$ is selected from a certain (possibly large) interval $[-\hat{U}, \hat{U}]$, which is not considered as a constraint, but rather as a numerical safety bound for our searching procedures.*

We assume that a solution of problem (3) exists and the pair $(u^*(\cdot), x^*(\cdot))$ as well as $J(u^*(\cdot)) = Q(x^*(T))$ are our reference points that should be found by an iterative learning of optimal control procedure.

Whenever it is clear that we consider one pass of our process in the nominal cases, i.e., $a = a^0$, then we shall write $\dot{x} = f(x, u)$, instead of (3). We shall denote by $f_u(x, u)$ the $d \times 1$ vector of derivatives with respect to u . Similarly, $d \times d$ matrix $f_x(x, u)$ denotes derivatives w.r.t. elements of x . In the same vein, $f_{uu}(x, u)$ is the vector of second derivatives, while $f_{ux}(x, u)$ is the matrix of derivatives of f_u w.r.t. x , while Q_x is the gradient of Q .

Assumption 2 (Differentiability) *We assume that all the above defined derivatives of f exists and they are continuous in $\mathcal{D} \times [-\hat{U}, \hat{U}]$ (hence, also bounded there).*

It is useful to express the Frechet derivative of J at $u(\cdot)$ in terms of adjoint variables $\psi(t) \in R^d$ and the Hamiltonian H , which is defined as follows

$$H(u, x(t), \psi(t)) = \psi^{tr}(t) f(x(t), u), \quad (4)$$

where ψ is a solution of the following adjoint equations:

$$\dot{\psi}(t) = -f_x(x(t), u(t)) \psi(t), \quad \psi(T) = Q_x(x(T)). \quad (5)$$

Under our differentiability assumptions, there exists the Frechet derivative F of J at $u(\cdot)$, which – for $t \in [0, T]$ – is given by

$$F(u(t), x(t), \psi(t)) = H_u(u(t), x(t), \psi(t)) = \psi^{tr}(t) f_u(x(t), u(t)), \quad (6)$$

(see, e.g., [8] for the definition of the Frechet derivative), which is an abstract version of a gradient.

3.3 Iterative learning algorithm

In Algorithm 1 that is presented below one can easily recognize that it is a functional analog of the gradient search algorithm. Assuming that the model contains nominal (exactly known) parameters a^0 , it can be run off-line.

Algorithm 1 (Iterative learning of optimal control)

Step 0 Select step length $\gamma > 0$ and an initial guess for control signal along the pass $u_0(t)$, $t \in [0, T]$. Set the pass counter $n = 0$. Select $\epsilon > 0$ as a parameter for stopping the procedure and $0 < \chi < 1$ as a factor for the step length reduction.

Step 1 Solve the state equations $\dot{x}_n = f(x_n, u_n, a^0)$, $x_n(0) = x_0$ to get x_n and calculate the quality criterion $J(u_n)$.

Step 2 Solve the adjoint equations $\dot{\psi}_n = -f_x(x_n, u_n, a^0) \psi_n$, $\psi_n(T) = Q_x(x_n(T))$.

Calculate the Frechet derivative $F(x_n(t), u_n(t), \psi_n(t), a^0) \stackrel{def}{=} \psi_n^{tr}(t) f(x_n, u_n, a^0)$ for all $t \in [0, T]$.

Step 3 If $\max_{t \in [0, T]} |F(x_n(t), u_n(t), \psi_n(t))| < \epsilon$, then STOP (provide $u_n(\cdot)$ as the result). Otherwise, improve u_n as follows:

$$u_{n+1}(t) = u_n(t) - \gamma F(x_n(t), u_n(t), \psi_n(t)), \quad t \in [0, T]. \quad (7)$$

Step 4 If $J(u_{n+1}) < J(u_n)$, then set $n := n + 1$ and go to Step 1. Otherwise, reduce *gamma* as follows $\gamma := \chi \gamma$ and go to Step 3.

Algorithm 1 runs in the continuous time between passes. It is possible to implement it in this way using such systems as Mathematica. In more traditional implementation one can either calculate and store u_n on a sufficiently fine grid or to use a more coarse grid and an interpolating algorithm.

3.4 Convergence of iterative learning

We need one more assumption on behavior of the Hamiltonian in the vicinity of optimum.

Assumption 3 *There exist constants $\bar{\rho} > 0$ and $\underline{\rho} > 0$ such that*

$$\bar{\rho} \geq H_{uu}(u, x^*(t), \psi^*(t)) \Big|_{u=u^*(t)} \geq \underline{\rho} > 0, \quad t \in [0, T], \quad (8)$$

where H_{uu} denotes the second derivative of H w.r.t. u .

We shall state the following result without proof,

Theorem 1 *Let Assumptions 1, 2 and 3 hold. Then, assuming that the stopping condition in Step 3 is switched off*

A) the sequence $u_n(t)$, $n = 0, 1, \dots$ generated by Algorithm 1 is convergent to a control signal $u^(t)$ such that the Frechet derivative of $J(u)$ vanishes at each point t along the pass $t \in [0, T]$ at which $u^*(t)$ is continuous,*

B) $\lim_{n \rightarrow \infty} J(u_n) = J(u^)$,*

C) $\lim_{n \rightarrow \infty} \|x_n(t) - x^(t)\|_d = 0$ at each point t along the pass $t \in [0, T]$ at which $u^*(t)$ is continuous.*

The proof is based on the fact the Frechet derivative for the optimal input signal is zero using also a functional analog of Taylor's expansion.

4 Multilayer system for laser power control

The results presented in the previous section form a base for proposing here a multilayer control system for a selective laser melting (SLM) process (see Fig. 1 left panel) and [7], [5] and the bibliography cited therein. It is natural to consider this process as a repetitive system, since the laser head moves back and forth when constructing a 3D body, e.g., a wall.

4.1 SLM process and its model

For a multi-pass SLM process we adopt a model developed in [17]. This model describes the a dependence between the temperature of the melted lake at k -th pass at time t , denoted as $y_k(t)$ and the laser power $W_k(t)$. For our purposes it has to be extended by including the heat exchange between passes. In order to incorporate the influence of the temperature at k -th pass $y_k(t)$ on the temperature at $(k + 1)$ pass we assume the model of the following form:

$$\tau y'_{k+1}(t) + y_{k+1}(t) = K (W_{k+1}(t))^\beta + \xi y_k(t), \quad y_{k+1}(0) = Y_k(0), \quad (9)$$

for passes $k = 0, 1 \dots$ and $t \in (0, T)$, where $T > 0$ is the pass length (the time that the laser head needs to travel along a 3D object under construction), while ξ is the coefficient that governs the influence of the temperature at k -th pass,

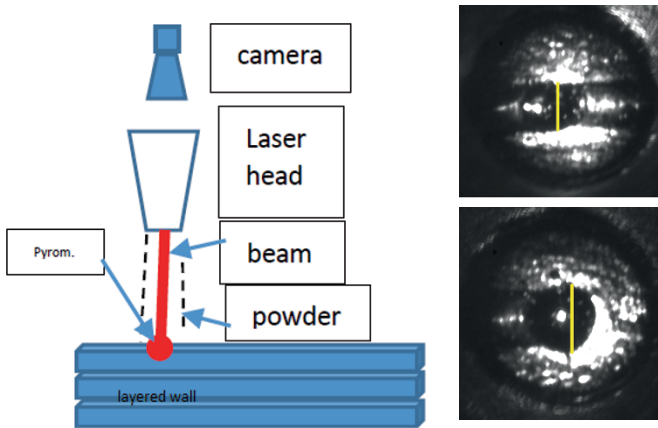


Fig. 1. Left panel – schematic view of the selective laser melting process (new powder is deposited and leveled). Right panel – images of deposits: upper image – proper width, lower image – a too fat deposit at the end of the wall

denoted as $Y_k(t)$, on the lake temperature at the next pass. Due to the back and forth movements of the laser head, for $t \in [0, T]$ $Y_k(t)$ is defined as follows:

$$Y_k(t) = \begin{cases} y_k(t) & \text{if } k \text{ odd,} \\ y_k(T-t) & \text{if } k \text{ even.} \end{cases} \quad (10)$$

The initial condition along the pass is assumed to be $Y_0(t) = Y_0 = \text{const}$, $t \in [0, T]$, where Y_0 is the temperature of the base. Notice that model (9) is nonlinear due to the presence of the term: $(W_{k+1}(t))^\beta$. The constants in (9) were identified experimentally in [17]):

- $\beta = 6.25 \cdot 10^{-2}$ is an experimentally selected constant,
- $\tau = 2.96 \cdot 10^{-2}$ – in sec. is the system time constant
- $K = 1413.58$ – overall system gain (depends on a metallic powder supply rate and the laser head speed).

4.2 Scheme of a multilayer control system with camera

The laser has its own local PI controller. However, this controller is not able to work properly at the ends of a formed 3D body (wall). This is visible in Fig. 1 (right panel). The upper image shows a proper width of the wall (bright yellow line) in its middle, while at the end point the wall is too thick. The reason is in that the laser head turns back at this point and stays longer in this area. For this reason the multilayer system (see Fig. 2) is proposed.

The role played by its blocks is the following:

Controller: The PI controller is applied. Its reference signal is calculated in the ILC block. In practice, the output signal, i.e., the temperature of the melted

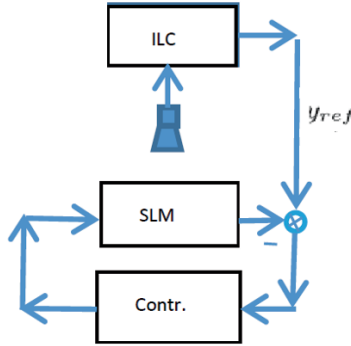


Fig. 2. Block diagram of the proposed multilayer control system of the laser power(see description in the text).

lake is provided by a pyrometer (in the simulations presented below, we use the solution of (9) as the output signal).

Camera: The camera provides images of the cladding process. Its main axis is parallel to the laser beam. The width of the cladding wall is estimated from images and the sequence of these widths is stored (at least for images taken along the present pass).

ILC: The main role of the ILC block is to provide a reference signal for several passes, This is done in two phases.

Phase 1. After an analysis of the stored sequence of widths a decision is made as to what should be a desirable reference signal, which is further denoted as $y_{des}(t)$. In the simulations reported below, we take $y_{des}(t)$ of the trapezoidal form (see Fig. 4 – dashed line). This form is aimed at reducing the negative turning point effect, while the analysis of the sequence of the wall widths allows to establish its parameters (the length of the flat area and the angles of its sides).

Phase 2. As J criterion we take the following one

$$\int_0^T [(y_{des}(t) - y_{k+1}(t))^2 + \delta W_{k+1}^2(t)] dt, \quad (11)$$

where $y_{k+1}(\cdot)$ and $W_{k+1}(\cdot)$ are constrained by (9), while $\delta > 0$ is a penalty factor for the excessive use of the laser squared power. Criterion (11) is to be minimized using the algorithm presented in the previous section (one can make one or more of its iterations). The resulting $y_k(\cdot)$ is passed to the lower layer as the reference signal for PI controller $y_{ref}(\cdot)$ at the next laser head pass.

Two questions may arise concerning the above multilayer control scheme. The first one is: why do we not use directly the signal $y_{des}(\cdot)$ as the reference signal for the PI control loop ? The reason is in that we can not be sure whether $y_{des}(\cdot)$

would be sufficiently safe for the laser system. By selecting sufficiently large δ we can assure that $y_{ref}(\cdot)$ obtained as above will be safe. The second question is: why do we not use the input signal, also calculated in the ILC block, that corresponds to $y_{ref}(\cdot)$. In fact, one can use this signal as a leading input signal and to use the PI controller to introduce necessary corrections. We have selected the way of not using this signal directly by technical reasons only – in order to avoid complicating further the control scheme. Notice that the camera and ILC block interfere with the built-in laser power PI control system only by $y_{ref}(\cdot)$.

Remark 1. One can add one more layer to the scheme in Fig. 2, namely a direct use of estimated wall widths for correcting the input signal of the SLM block. A discussion of this extension is outside the scope of this paper.

4.3 The results of simulations

The multilayer control system for (9) was simulated with δ set to 1. in (11). The results are shown in Fig. 3 and Fig. 4. The former one indicates that the rate of convergence of the ILOC algorithm is sufficiently fast, both for the optimality criterion (left panel) as well as for the mean squared tracking error (right panel). The l.h.s panel of the latter compares $y_{des}(\cdot)$ (dashed line) with the system output after 15 iterations (solid line). The difference between them indicates that $y_{15}(\cdot)$ is a good candidate for $y_{ref}(\cdot)$ signal, since the approximately optimal input signal (right panel) stays within safe bounds. One can further reduce the tracking error by reducing δ .

Alternatively, one can try to apply $y_k(\cdot)$ also for $k < 15$ from pass to pass, but this discussion is outside the scope of this paper. Two remarks are in order

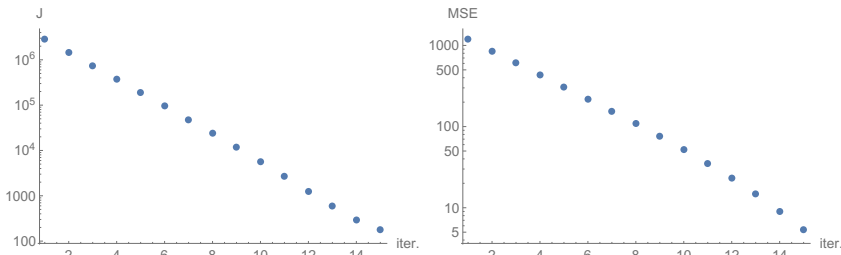


Fig. 3. Convergence rate of decay of (11) criterion (left panel) and the mean squared tracking error (right panel). Both plots are in log-scale.

concerning Fig. 4.

1) As it was explained in Section 3, signal u_{opt} in Fig. 4 (right panel) is not passed to the laser power system. Hence, it is not visible in Fig. 2. Instead, the calculated reference signal (solid line in Fig. 4 (left panel) is passed to the PI control system.

2) Initial oscialtions visible in Fig. 4 resulted from stopping the simulations after

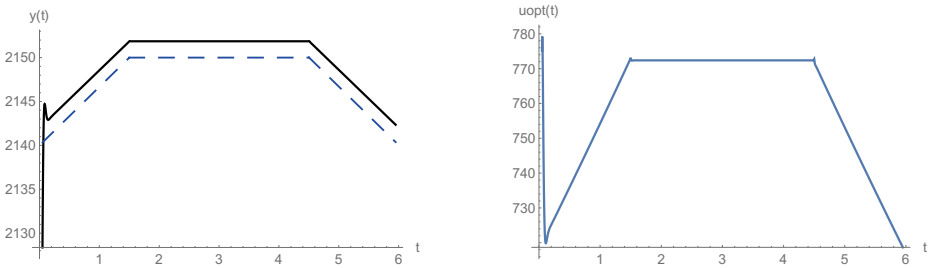


Fig. 4. Left panel – approximately optimal output signal of the ILC block (solid line) and the desired signal (dashed line). Right panel – approximately optimal input signal after 15 iterations.

15 iterations only. The starting point was the input constant signal with the level 780 W.

5 Conclusions

In the main part of the paper the algorithm of iterative learning control for repetitive, nonlinear processes was proposed. The results concerning its convergence are stated without proofs, by the lack of space. They will be published elsewhere. This algorithm formed a base for the multilayer control scheme with a camera that was proposed in Section 4 together with the results of simulations. The simulations indicate that the convergence rate of this approach is sufficiently fast for the SLM process, since its one pass takes 6 seconds.

In Section 2 preliminary ideas of image driven, model free control systems are also presented. It seems that this approach is promising, but requires further research.

References

1. Boltyanski V. and Poznyak A. *The Robust Maximum Principle: Theory and Applications*. Springer Science & Business Media, 2011.
2. Chapman K., Johnson W., and McLean T. A high speed statistical process control application of machine vision to electronics manufacturing. *Computers & Industrial Engineering*, 19(1):234–238, 1990.
3. Davies E. R. *Machine vision: theory, algorithms, practicalities*. Elsevier, 2004.
4. Hladowski L., Galkowski K., Cai Z., Rogers E., Freeman Ch., and Lewin P. Experimentally supported 2d systems based iterative learning control law design for error convergence and performance. *Control Engineering Practice*, 18(4):339–348, 2010.
5. Jurewicz P., Rafajłowicz W., Reiner J., and Rafajłowicz E. Simulations for tuning a laser power control system of the cladding process. In *IFIP International Conference on Computer Information Systems and Industrial Management*, pages 218–229. Springer, 2016.

6. King T. Vision-in-the-loop for control in manufacturing. *Mechatronics*, 13(10):1123–1147, 2003.
7. Kurzynowski T., Chlebus E., Kuźnicka B., and Reiner J. Parameters in selective laser melting for processing metallic powders. In *SPIE LASE*, pages 823914–823914. International Society for Optics and Photonics, 2012.
8. Luenberger D. *Optimization by vector space methods*. John Wiley & Sons, 1997.
9. OLeary P. Machine vision for feedback control in a steel rolling mill. *Computers in Industry*, 56(8):997–1004, 2005.
10. Owens D. *Iterative learning control. An optimization paradigm*. Springer, 2016.
11. Owens D. and Hätönen J. Iterative learning controlan optimization paradigm. *Annual reviews in control*, 29(1):57–70, 2005.
12. Rafajłowicz E., Pawlak-Kruczek H., and Rafajłowicz W. Statistical classifier with ordered decisions as an image based controller with application to gas burners. In *International Conference on Artificial Intelligence and Soft Computing*, pages 586–597. Springer, 2014.
13. Rafajłowicz E. and Rafajłowicz W. Image driven decision making with application to control gas burners. In *IFIP International Conference on Computer Information Systems and Industrial Management – submitted*.
14. Rafajłowicz E. and Rafajłowicz W. Iterative learning in repetitive optimal control of linear dynamic processes. In *International Conference on Artificial Intelligence and Soft Computing*, pages 705–717. Springer, 2016.
15. Rigelsford J. Industrial image processing: Visual quality control in manufacturing. *Sensor Review*, 21(2), 2001.
16. Rogers E., Galkowski K., and Owens D. *Control systems theory and applications for linear repetitive processes*, volume 349. Springer Science & Business Media, 2007.
17. Tang L. and Landers R. Melt pool temperature control for laser metal deposition processespart i: Online temperature control. *Journal of manufacturing science and engineering*, 132(1):011010, 2010.

Acknowledgements The research of the 1-st author has been supported by the National Science Center under grant: 2012/07/B/ST7/01216.

The authors express their thanks to Professor Jacek Reiner and MSc Piotr Jurewicz for providing images shown in Fig. 1 and for fruitful discussions.

Real-Time Control of Active Stereo Vision System

Przemysław Szewczyk

Institute of Automatic Control, Department of Robot Control,
ul. B. Stefanowskiego 18/22, PL-90-924 Lodz, Poland
przemyslaw.szewczyk@o2.pl
<http://automatyka.p.lodz.pl>

Abstract. This paper describes an approach for tracking objects in 3D scene using Stereo Vision System with the control of cameras gaze and vergence. The goal of camera positioning mechanism is to keep cameras fixed on a common visual target. Mechanical linkage between cameras ensures separate control of vergence and gaze angles and the same distance from the left and the right camera to the fixated world point. Keeping the tracking target fixated causes that object lies near *horopter* - a surface with zero disparity. It means that stereo images of this object have a narrow range of disparities and makes it possible to use stereo algorithm that accepts only a limited range of disparities. On the other hand, most of the stereo matching algorithms need fully rectified input images or at least accurate camera calibration parameters. Side effect of such active control of cameras positioning is a continuous degradation of calibration and needs re-calibration or estimation of stereo parameters for each new position.

Keywords: Stereo Vision, Vision-Based Control, Visual Servoing, Camera Positioning System

1 Introduction

The vergence angle of a stereo vision system is the angle between the optic axes of its cameras. This angle together with baseline length and gaze direction of a binocular system determine a particular fixation point, as shown in figure 1. Having the cameras verged toward each other causes that zero disparity occurs at a finite distance and it might be necessary to achieve greater depth resolution in the proximity of that particular distance determined by fixation point [1]. Keeping the fixation point near some visual target implies that object lies near zero disparity surface called *horopter*[7]. As a result stereo images of this object have a narrow range of disparities and it is possible to use stereo algorithm which operates only on limited range of disparities to reconstruct 3D space in neighbourhood of target object. Stereo correspondence algorithms, even in the basic correlation-based form, are computationally expensive[2], so using such approach can significantly speed up process of searching correspondences. Above

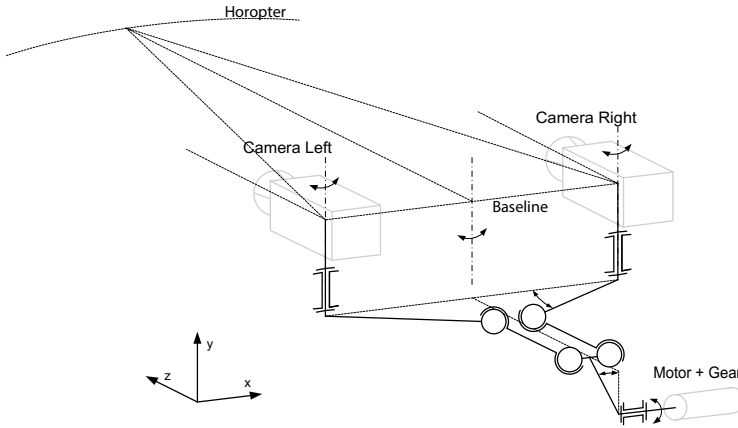


Fig. 1: Vergence control mechanism

features became a main motivation to build stereo vision system with active vergence and gaze control.

From the mechanical point of view, cameras vergence can be controlled by separate motors, each camera independently, or by a single motor that converges both cameras symmetrically via mechanical linkage. The second design ensures separate control of vergence and gaze angles and the same distance from the left and the right camera to the fixated world point. This approach was chosen to build this vision system (Fig.1). Detailed information about mechanical construction and parameters optimization of mechanical linkage was described in [6]. To control gaze and vergence angle DC motors with encoder feedback were used.

2 Visual servo system

Active Stereo Vision positioning control system, at the most abstract level, may be described by four major components typical for vision-based control [3]: *joint controller* that generates a response to the error between measured and required joint positions, *active stereo vision system* that executes the joint controller commands and produce stereo pair images on the output, *feature extractor* that converts image data to feature feedback, and *image-based visual servo controller*, that estimates required vergence and gaze angle based on feature feedback signal.

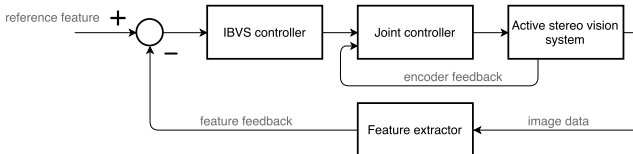


Fig. 2: Image-based visual servo system

These components can be mapped onto the traditional block diagram model of a feedback system, shown in figure 2.

Internal feedback loop from figure 2 can be considered as a standard closed-loop position control of DC motors and implementing *joint controller* as a pair of discrete PID regulators is quite natural. The desired behaviour of motor positioning system is to achieve required position as soon as possible without "ringing" effect, which has huge impact on quality of captured images (especially for CMOS rolling-shutter cameras). Good tuned PID controllers are critical for the whole system.

Feature extractor block, which is working directly on image data, for each stereo frame returns pixel positions of visual target relative to principal point of each single camera, what can be written as pair $[d^l, d^r]$. Because gaze control mechanism has only one degree of freedom and tilt angle cannot be adjusted, only horizontal x-component is taken for calculation. The task of whole system is to keeping the fixation point near some visual target and for this case measured pixel positions $[d^l, d^r]$ are close zero, so reference feature signal from figure 2 can be written as a zero vector.

Estimation of required vergence and gaze angles is done by *IBVS controller* in each iteration in following way:

$$\begin{aligned}\alpha_k &= \alpha_{k-1} + K_G(\hat{d}_k^l + \hat{d}_k^r) \\ \beta_k &= \beta_{k-1} + K_V(\hat{d}_k^l - \hat{d}_k^r)\end{aligned}\tag{1}$$

Where: α_k - estimated gaze angle, α_{k-1} - gaze angle from previous iteration, K_G - empirically determined gain for gaze component, β_k - estimated vergence angle, β_{k-1} - gaze vergence from previous iteration, K_V - empirically determined gain for vergence component, \hat{d}_k^l, \hat{d}_k^r - x-component of visual target relative position respectively for left and right camera, values predicted using Kalman filter.

3 Motor-based stereo camera parameters estimation

When two cameras view a 3D scene from two distinct points, there are a geometric relations between the 3D points and their projections onto the 2D images that lead to constraints between the image points. These relations can be described by epipolar geometry and precise determination of these relations is necessary for further stereo processes, such as image rectification, calculating image disparity or 3D reconstruction. Epipolar geometry is based on the assumption that the cameras can be approximated by the pinhole camera model. This assumption can be easily fulfilled complementing the pinhole camera model by Brown-Conrady distortion model, which ensures good results for standard-focal length camera lenses. All parameters describing above pinhole camera and distortion models can be closed respectively in intrinsic camera matrix M_i and distortion coefficient vector D . Assuming that these parameters are constant for ones calibrated cameras, they can be used as an input for stereo calibration process, which simplifies to computing geometrical relationship between two cameras in space. This

relation can be expressed by a pair of rotation matrix R and translation vector T or in more compact form by essential matrix E .

For mono-camera calibration Zhang's method was used. This method uses regular planar pattern - chessboard and calculates planar homography, which is defined as projective mapping from one plane to another (in this case, from the object plane to the image plane) and can be described by homography matrix H . Doing the decomposition of calculated H matrix not only intrinsic and distortion parameters can be determined but also relation between camera coordinate system and object coordinate system - external camera parameters (rotation matrix R_c and translation vector T_c). Using this method for both synchronised cameras, any given 3D point P in object coordinates can be put in the left and right camera coordinates:

$$\begin{aligned} P_l &= R_{cl}P + T_{cl} \\ P_r &= R_{cr}P + T_{cr} \end{aligned} \quad (2)$$

Where: $P_l, (P_r)$ - locations of the point P from the left and right camera coordinate system respectively, $R_{cl}, T_{cl}, (R_{cr}, T_{cr})$ - rotation matrix and translation vector from the left and right camera coordinate system respectively to the point P . Knowing that two views of point P are related by $P_l = R^T(P_r - T)$, where R and T are the rotation matrix and translation vector that bring the right camera coordinate system into the left, above relations can be transformed to:

$$\begin{aligned} R &= R_{cr}R_{cl}^T \\ T &= T_{cr} - RT_{cl} \end{aligned} \quad (3)$$

Using above method, rotation and translation between cameras are slightly different for each chessboard pair due to image noise and rounding errors so algorithm calculate median values of the R and T and use them as input for a robust Levenberg-Marquardt iterative algorithm to find the minimum of the reprojection error of the chessboard corners for both camera views, and the solution for R and T is returned.

The problem occurs when the vergence angle changes. This calibration method is computationally expensive and needs planar pattern visible in both cameras, so cannot be used for recalibration of system in real time. Also direct calculation of rotation matrix R and translation vector T is not possible due to some construction obstacles such as: it is not possible to mount cameras so that their axes of rotation pass through the nodal points - in this case each vergence movement includes translation components, the mechanical linkage in vergence control mechanism causes that cannot be measured rotation of each camera separately, mechanical linkage itself can be not ideally symmetrical.

Knowing that the relation between motor angle and vergence angle is not linear and can be expressed in general form by:

$$\beta = f(\theta, \gamma) \quad (4)$$

where: β - vergence angle, θ - motor angle, γ - parameter vector, the whole motor angle range can be divided into few uniform subranges and the

stereo calibration process can be run for N motor angles. Next, doing the conversion of each rotation matrix R to vector form $u = [u_x u_y u_z]^T$ using Rodriguez formula:

$$\sin(\beta) \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ u_y & u_x & 0 \end{bmatrix} = \frac{R - R^T}{2} \quad (5)$$

the vergence angle β and associate unit vector s representing the rotation axis can be calculated from formulas:

$$\begin{aligned} \beta &= \|u\| \\ s &= \frac{u}{\|u\|} \end{aligned} \quad (6)$$

From collected during calibration measurements in form of N pair (β_i, θ_i) it is possible to determine relation (4) and next based on this relation for any motor angle θ a new vergence angle β can be calculated. Any vergence angle complemented with s is transformed back to R matrix from equation:

$$R = \cos(\beta) \cdot I + (1 - \cos(\beta)) \cdot uu^T + \sin(\beta) \cdot \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ u_y & u_x & 0 \end{bmatrix} \quad (7)$$

4 Experiments and Results

4.1 Hardware and Software configuration

Figure 3 shows a system component diagram with selected data flow. There are three main blocks:

- *Active Stereo Vision*, where are located two PointGrey Dragonfly2 cameras with 1/3-inch global-shutter sensors. Cameras heads with fixed 4mm focal length lenses are mounted on rotating platforms, which can be precisely adjustable to minimise displacement between cameras nodal points and their axes of rotation (Fig.11). The vergence angle is changed by DC gearhead motor with ratio 172:1 and the angle position is returned to *Joint controller* from encoder attached by motor side. To change gaze angle of vision system, the whole platform with both cameras are rotated around the axis of rotation located in the centre of baseline. For gaze adjustment also DC gearhead motor with encoder is used.
- *Embedded Control System* with the FreeRTOS system running on STM32F4 ARM processor with DSP and FPU cores. The block is responsible for joints control and cameras synchronisation. Joint PID controller task works with 5ms period. For each camera capture requests positions of vergence and gaze angles are send to the Host PC. The System also controls DC motors currents to protect motors against damage.

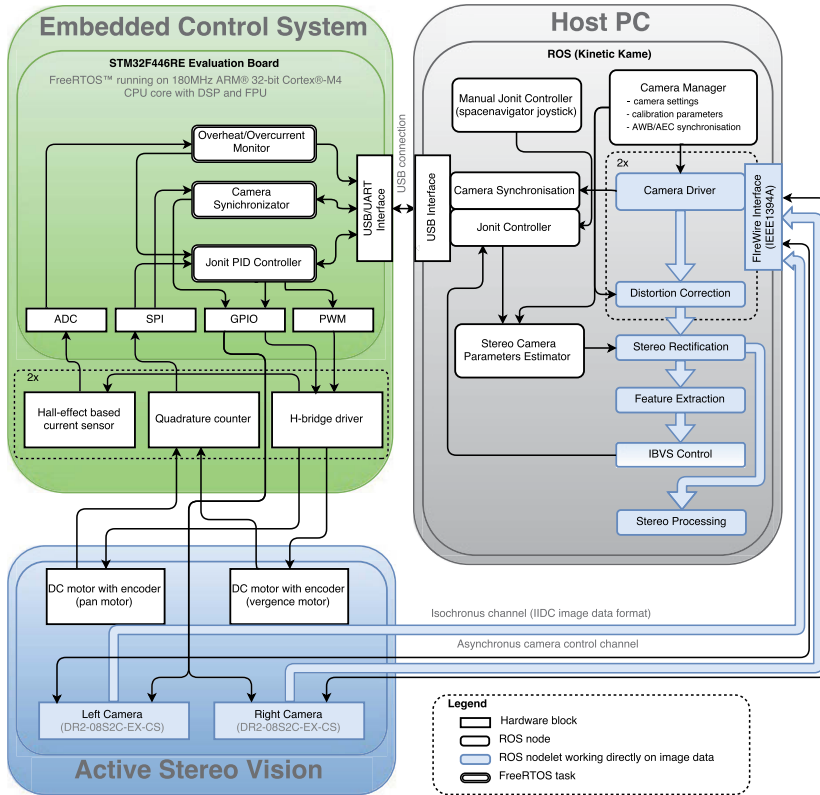


Fig. 3: System component diagram

- *Host computer* with the ROS system on board. The whole image processing part is done by this block. Figure 3 shows main system which use ROS inter-process communication. The image processing part has cascaded pipeline architecture and operate on nodelets blocks, which provide a way to run multiple algorithms in the same process with zero copy transport between algorithms [5].

4.2 Vergence and gazo control system tuning

The quality of the motor system is critical for the work of whole system. To achieve fast response to setpoint signal and avoid any oscillation a good model of motor system or plant is needed. DC motors traditionally are modelled as second order linear system, which ignores the dead nonlinear zone of the motor. Unfortunately, the dead zone caused by the nonlinear friction has huge impact to servo systems [4]. For plant identification and get higher-fidelity model of the DC motor System Identification Toolbox from MathWorks was used. Nonlinear ARX model, which has various adjustable components, such as model orders, delays, type of nonlinear function, and the number of units in the nonlinear

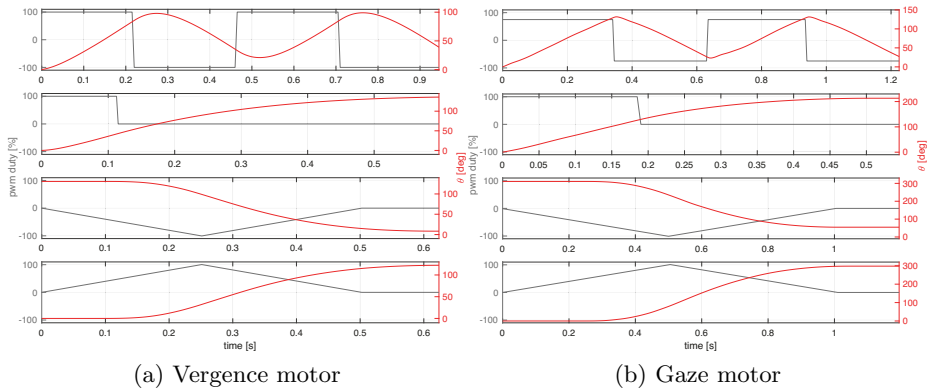


Fig. 4: Motors' responses to the square wave, step and triangle signals (Sampling time $T_s = 5\text{ms}$)

function, was chosen. After added regressors that represent saturation and dead-zone behavior and several iterations excellent fit greater than 85% was achieved. Motors' responses to the square wave, step and triangle signals were used as estimation and validation data (Fig.4)

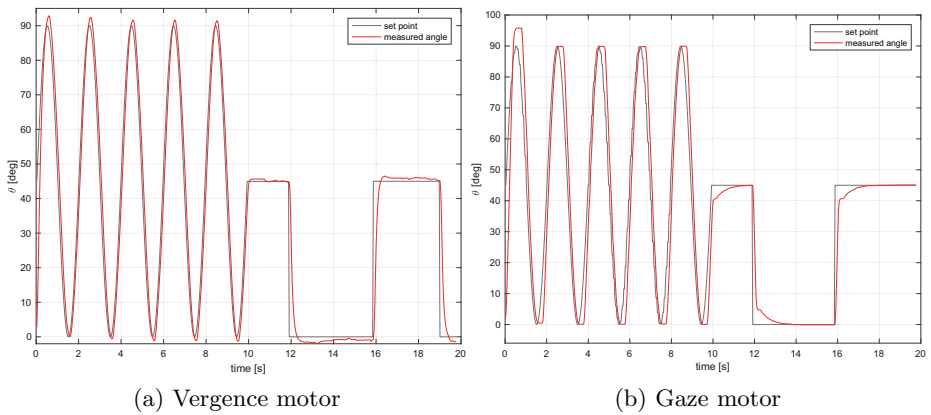


Fig. 5: Closed-loop system responses ($T_s = 33\text{ms}$)

The discrete PID controller gains were chosen using trial-and-error approach to achieve slightly underdamped response. Closed-loop system responses to mixed sinusoidal and step signal was shown on figure 5.

4.3 Stereo camera parameters estimation results

To verify quality of estimation stereo camera parameters the whole range of vergence motor angle θ was divide onto $N_k = 80$ uniform subranges. For each θ_i , where $i = 1..N_k$, process of stereo calibration was repeated and in the result set of (R_i, T_i) was obtained. Figures 6,7,8 shows respectively translation vector T_i

and pair - unit vector s_i and vergence angle β_i obtained form decomposition of rotation matrix R_i , for each motor angle θ_i .

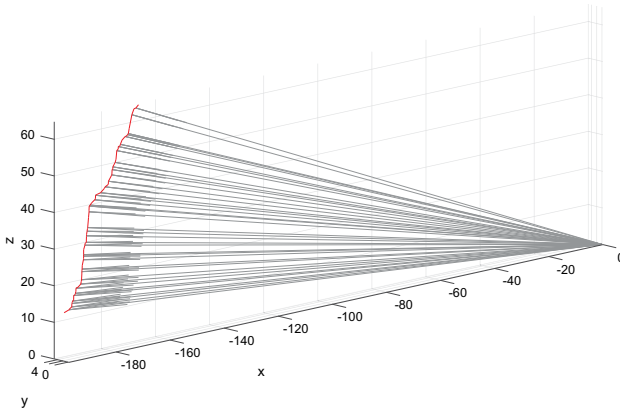


Fig. 6: Translation vector T_i calculated in calibration process for each θ_i

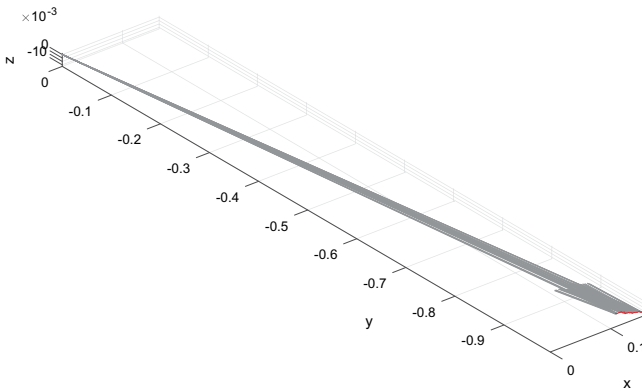


Fig. 7: Unit vector s_i representing the rotation axis for each θ_i

Figure 7 shows that rotation axis does not coincide perfectly with y-axis and it tends to slight move with θ changes. On figure 8, next to relation between β_i and θ_i calculated in stereo calibration process, the theoretical relation obtained from kinematic model of mechanical linkage was added [6].

During the mono calibration process a quality of calculated parameters is verified by using the reprojection method [1]. It means that 3D points of the checkerboard are transformed using the computed intrinsic and extrinsic parameters, projected to the left and the right image and compared to the detected corners. The reprojection error can be defined in below form:

$$err_i = \frac{\sqrt{\left(\sum_{j=0}^{N_C N_I} \|\hat{p}_{ijL} - p_{ijL}\|\right)^2 + \left(\sum_{j=0}^{N_C N_I} \|\hat{p}_{ijR} - p_{ijR}\|\right)^2}}{2N_C N_I} \quad (8)$$

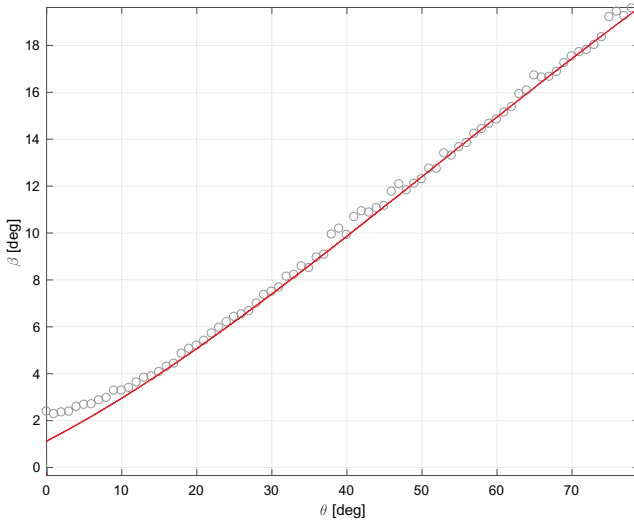


Fig. 8: Relation between β_i representing the rotation axis for each θ_i

where: $p = [x, y]$ - a image point of chessboard pattern, respectively for left and right image, \hat{p} - the same point obtained by reprojection, N_C - number of corners in chessboard pattern, N_I - number of chessboard images.

To verify correctness of rotation matrix R estimation, the similar to above approach was used. For some vergence motor angles θ_i (for which calibration process was done) the reprojection error was calculated, but instead of using extrinsic camera parameters from calibration, for one camera, extrinsic parameters were obtained from equation 3 using estimated rotation matrix R and converted from vector form by (7). The results are shown in figure 9.

Visual verification of estimated stereo parameters presents figure 10, where the estimated stereo camera parameters were used as an input for Bouguet's rectification algorithm.

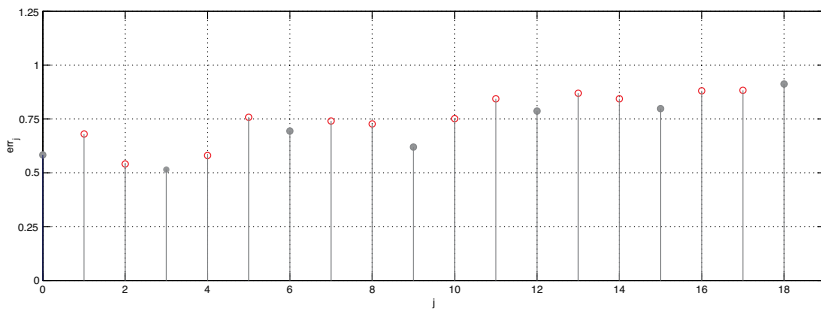


Fig. 9: Reprojection error for parameters obtained from calibration (filled circle) and for estimated parameters (empty circle)

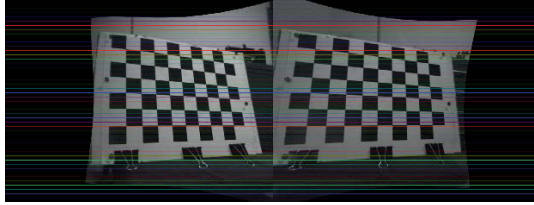


Fig. 10: Result of stereo rectification for vergence angle $\beta = 19deg$

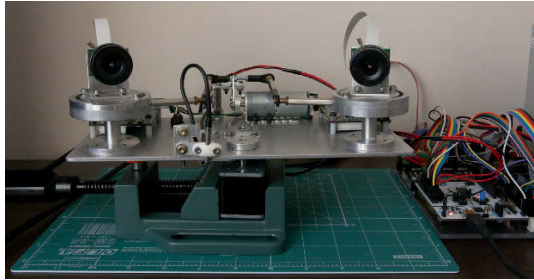


Fig. 11: Active stereo vision system

5 Conclusion

The results described in this paper show that estimation of stereo parameters based on motor angle is sufficient for practical use. Unfortunately, for cases where relation between motor angle and vergence angle is highly nonlinear it needs many singular calibration processes. Because algorithm operates only on stereo parameters from previous calibrations and updating only already obtained parameters, it can be use in real-time implementation.

References

1. Bradski, G., Kaehler, A.: Learning OpenCV 3 - Computer Vision in C++ with the OpenCV Library. O'Reilly (2017)
2. Brown, M.Z., Burschka, D., Hager, G.D.: Advances in computational stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(8), 993–1008 (Aug 2003)
3. Corke, P.: Robotics, vision and control : fundamental algorithms in MATLAB. Springer, Berlin (2013)
4. Nayana P. Mahajan, S.D.: Study of nonlinear behavior of dc motor using modeling and simulation. International Journal of Scientific and Research Publications 3 (2-13)
5. Quigley, M., Gerkey, B., Smart, W.D.: Programming Robots with ROS: A Practical Introduction to the Robot Operating System. O'Reilly Media (2015)
6. Szewczyk, P.: Stereovision system with active vergence and gaze control mechanism. PAK (11), 1344 – 1347 (Nov 2011)
7. Thomas J. Olson, D.J.C.: Real-time vergence control for binocular robots. International Journal of Computer Vision (7), 67 – 89 (1991)

A receding-horizon approach to state estimation of the battery assembly system

Pawel Majdzik, Ralf Stetter

Institute of Control and Computation Engineering,
University of Zielona Góra,
ul. Podgórna 50, Zielona Góra, Poland
Faculty of Mechanical Engineering, University of Applied Sciences
Ravensburg-Weingarten, Doggenriedstrae, Weingarten, Germany
p.majdzik@issi.uz.zgora.pl
stetter@hs-weingarten.de

Abstract. The paper addresses the issue of the state estimation problem of a class of discrete-event systems. The receding-horizon approach is employed to solve above problem. The system and its variables are described within the $(\max, +)$ algebra. Thus making possible to incorporate robustness within the overall framework. The paper also shows the transformation of the interval cost function into the scalar one, and hence, making the computational procedure trackable within the quadratic programming framework. Resulting in interval estimates of the system state, which can be used for both fault diagnosis and control purposes.

Keywords: $(\max, +)$ algebra, interval arithmetics, observers, predictive control

1 Introduction

Nowadays industrial systems increases in complexity, that a proper control and detailed insight into them become more and more difficult. The industrial plants have to work more efficiently, they should be cheap to built and their maintenance costs should be as low as possible. At the same time, plant safety as well as quality of products need to be at the highest possible level. Taking into account above properties of systems and the scope of control engineering, it is obvious fact that a single strategy is not able to guarantee all industrial demands. In the flexible production systems, the low level control (e.g., PLC or PAC) requires some upper level scheduling of individual production tasks, which usually depend on current production times. In this paper, a Discrete Event Systems (DES) are taken into consideration, while in real life a lot of plants are driven by events occurred during the operation time. In this class of systems, a main description formalism is a $(\max, +)$ algebra, which, due to a nature of DES, is a very natural and intuitive mathematical formalism [6, 10]. While the control, and especially, Fault-Tolerant Control (FTC) [11, 13, 14] and Fault Diagnosis (FD) [9] are mainly developed in conventional algebra and used to maintain a

sub-systems of industrial plants (e.g., manipulators), DES are still in the shadow of the main research areas of FTC [12, 8] and FD [5]. Due to the fact that there is a need for development of such tools for DES, $(\max,+)$ algebra is a natural candidate. The main contribution of this paper is to provide a tool that can be applied to state estimation of uncertain DES using $(\max,+)$ algebra. To the best of authors knowledge, there is no tool like that for uncertain DES. The only available state estimator is proposed in [5]. However, it can be used for deterministic systems only. Moreover, it is obvious that the direct measurement of all operation times is not possible in a highly sophisticated, especially in distributed industrial plant. This means that their (state variables) estimation can be of leading importance. In this paper, the idea of receding horizon observers [15, 4, 7] is extended for $(\max,+)$ algebra [5]. The paper is organized as follows. Section 2 describes essential definitions and concepts. The receding-horizon approach for state estimation is provided in details in Section 3. Then, section 4 contains a complete description of the multicopter assembly system. The performance of the proposed approach is illustrated as well. Eventually, the last section concludes the paper and indicates future research directions.

2 Preliminaries

The main objective of this section is to recall interval arithmetics and define mathematical concept $(\text{imax}, +)$ algebra in order to cope with uncertain systems.

2.1 $(\text{Max}, +)$ Algebra

The $(\max,+)$ algebraic structure $(\mathbb{R}_\varepsilon, \oplus, \otimes)$ is defined as follows:

$$\begin{aligned} \mathbb{R}_\varepsilon &\triangleq \mathbb{R} \cup \{-\infty\}, \\ \forall a, b \in \mathbb{R}_\varepsilon, a \oplus b &= \max(a, b), \\ \forall a, b \in \mathbb{R}_\varepsilon, a \otimes b &= a + b, \end{aligned} \tag{1}$$

where \mathbb{R} is the field of real numbers. The operators \oplus and \otimes denote the $(\max,+)$ algebraic addition and $(\max,+)$ algebraic multiplication, respectively. The main properties of $(\max,+)$ algebra operators are as follows:

$$\begin{aligned} \forall a \in \mathbb{R}_\varepsilon : a \oplus \varepsilon &= a \text{ and } a \otimes \varepsilon = \varepsilon, \\ \forall a \in \mathbb{R}_\varepsilon : a \otimes e &= a, \end{aligned} \tag{2}$$

where $\varepsilon = -\infty$ and $e = 0$ are the neutral elements for the $(\max,+)$ algebraic addition and $(\max,+)$ algebraic multiplication operators, respectively.

For matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}_\varepsilon^{m \times n}$ and $\mathbf{Z} \in \mathbb{R}_\varepsilon^{n \times p}$

$$(\mathbf{X} \oplus \mathbf{Y})_{ij} = x_{ij} \oplus y_{ij} = \max(x_{ij}, y_{ij}), \tag{3}$$

$$(\mathbf{X} \otimes \mathbf{Z})_{ij} = \bigoplus_{k=1}^n x_{ik} \otimes z_{kj} = \max_{k=1, \dots, n} (x_{ik} + z_{kj}) \tag{4}$$

Further definitions and details related to the $(\max,+)$ algebra formalism can be found in [1, 2]. In general, the description of a discrete event system is nonlinear in a conventional algebra. However, there exists a class of DES - called the $(\max,+)$ linear discrete event systems - that can be described by $(\max,+)$ linear model. In the linear discrete event system there is only synchronization of tasks but no concurrency. Typical examples of linear DES are transportation systems with tight connection constraints (railway network), logistic systems (conveyance and storage of goods) and flexible manufacture systems with fixed scheduling rules. An example of the last kind of DES constitutes the subject of this paper and is described in 4. Thus, using $(\max,+)$ algebra, DES can be described by:

$$\mathbf{x}_{k+1} = \mathbf{A} \otimes \mathbf{x}_k \oplus \mathbf{B} \otimes \mathbf{u}_k, \tag{5}$$

$$\mathbf{y}_k = \mathbf{C} \otimes \mathbf{x}_k, \tag{6}$$

where the index k is an event counter, while:

- $\mathbf{x}_k \in \mathbb{R}_\varepsilon^n$ represents the state typically containing the time instants at which the internal events occur for the k th time,
- $\mathbf{u}_k \in \mathbb{R}_\varepsilon^r$ is the input vector containing the time instants at which the input events occur for the k th time,
- $\mathbf{y}_k \in \mathbb{R}_\varepsilon^m$ states for the output vector containing the time instants at which the output events occur for the k th time,
- $\mathbf{A} \in \mathbb{R}_\varepsilon^{n \times n}$ is the state transition matrix, $\mathbf{B} \in \mathbb{R}_\varepsilon^{n \times r}$ is the control matrix, $\mathbf{C} \in \mathbb{R}_\varepsilon^{m \times n}$ is the output matrix.

Having a general system description, there exist the possibility of developing its uncertain version with interval-based representation.

2.2 Interval Arithmetics

While exact values of any physical variable are impossible to obtain or measure, interval arithmetics can be an appealing alternative to typical stochastic description of the uncertainty. In the stochastic case, the core task is to assign appropriate distribution to an uncertain variable. Therefore instead of deliberating on the nature, source and distribution of noise/disturbances it is more valuable (to the common sense) to obtain the maximum and minimum value of a given variable.

An interval is a closed connected set of reals:

$$a = [\underline{a}, \bar{a}] = \{x \in \mathbb{IR} \mid \underline{a} \leq x \leq \bar{a}\} \tag{7}$$

where \underline{a} and \bar{a} are upper and lower endpoints, respectively. For $a, b \in \mathbb{IR}$ arithmetic operations are defined by:

$$\exists a, b \in \mathbb{IR}, a \circ b := \{x \circ y \mid x \in a, y \in b\}, \forall \circ \in \{+, -, \cdot, /\} \tag{8}$$

where \mathbb{IR} is the closed set of real compact intervals with respect to these operation. Each operation $a \circ b$ can be represented by using only the bounds of a and b , and hence, the following hold:

$$\begin{aligned} a + b &= [\underline{a} + \underline{b}, \bar{a} + \bar{b}], a - b = [\underline{a} - \bar{b}, \bar{a} - \underline{b}] \\ a \cdot b &= [\min\{\underline{a}\underline{b}, \underline{a}\bar{b}, \bar{a}\underline{b}, \bar{a}\bar{b}\}, \max\{\underline{a}\underline{b}, \underline{a}\bar{b}, \bar{a}\underline{b}, \bar{a}\bar{b}\}] \\ a/b &= a \cdot (1/b), \quad \text{where } 1/b = \{1/x : x \in b\} = [1/\bar{b}, 1/\underline{b}] \quad \text{if } 0 \notin b \end{aligned} \tag{9}$$

2.3 Interval (Max,+) Algebra

The goal of this point is to provide a novel methodology that can be applied to express the robustness to parameter uncertainties of the matrices \mathbf{A} , \mathbf{B} and \mathbf{C} related to the system (5)–(6). This idea has never been used for state estimation purposes, but it was only applied to analyze uncertain systems [3].

The (imax,+) algebraic structure $(\mathbb{IR}_\varepsilon, \oplus, \otimes)$ is defined as follows:

$$\begin{aligned} \mathbb{IR}_\varepsilon &\text{ is a set of real compact intervals of the form} \\ a &= [\underline{a}, \bar{a}] = \{x \in \mathbb{IR}_\varepsilon \mid \underline{a} \leq x \leq \bar{a}\} \\ \forall a, b \in \mathbb{IR}_\varepsilon, a \oplus b &= [\max(\underline{a}, \underline{b}), \max(\bar{a}, \bar{b})] \\ \forall a, b \in \mathbb{IR}_\varepsilon, a \otimes b &= [\underline{x}_{ij} + \underline{y}_{ij}, \bar{x}_{ij} + \bar{y}_{ij}] \end{aligned} \tag{10}$$

Similarly as in the previous point, for matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{IR}_\varepsilon^{m \times n}$ and $\mathbf{Z} \in \mathbb{IR}_\varepsilon^{n \times p}$:

$$\begin{aligned} (\mathbf{X} \oplus \mathbf{Y})_{ij} &= x_{ij} \oplus y_{ij} = [\max(\underline{x}_{ij}, \underline{y}_{ij}), \max(\bar{x}_{ij}, \bar{y}_{ij})] \\ (\mathbf{X} \otimes \mathbf{Z})_{ij} &= \bigoplus_{k=1}^n x_{ik} \otimes z_{kj} = \bigoplus_{k=1}^n [\underline{x}_{ik} + \underline{z}_{kj}, \bar{x}_{ik} + \bar{z}_{kj}]. \end{aligned}$$

Having a general (imax,+) framework, it is possible to extend the usual (max,+) linear model state-space model to interval matrices. The equations (5)–(6) can be described in a robust form that takes into account parameter uncertainty:

$$\mathbf{x}_{k+1} = \mathbf{A} \otimes \mathbf{x}_k \oplus \mathbf{B} \otimes \mathbf{u}_k, \tag{11}$$

$$\mathbf{y}_k = \mathbf{C} \otimes \mathbf{x}_k, \tag{12}$$

- $\mathbf{x}_k \in \mathbb{IR}_\varepsilon^n$ is n -dimensional state vector;
- $\mathbf{u}_k \in \mathbb{IR}_\varepsilon^r$ is r -dimensional input vector ;
- $\mathbf{y}_k \in \mathbb{IR}_\varepsilon^m$ is m -dimensional output vector;
- $\mathbf{A} \in \mathbb{IR}_\varepsilon^{n \times n}$ is the state transition matrix, $\mathbf{B} \in \mathbb{IR}_\varepsilon^{n \times r}$ is the control matrix, $\mathbf{C} \in \mathbb{IR}_\varepsilon^{m \times n}$ is the output matrix;

3 Receding-horizon approach to state estimation

In this section, an algorithm is proposed, which allows solving a receding-horizon state estimation problem of (11)–(12) using available output measurements. In

the receding-horizon strategy, at any time $k = N, N + 1, \dots$, one has to determine estimates of \mathbf{x}_{k-N} using the output measurements \mathbf{y}_{k-N}^k collected over a predefined time window $[k - N, k]$. Let us define $\hat{\mathbf{x}}_{k-N,k}, \dots, \hat{\mathbf{x}}_{k,k}$ the estimates of $\mathbf{x}_{k-N}, \dots, \mathbf{x}_k$ made at the $k - th$ event counter. Moreover, apart from the state estimate, a prediction of the state is defined as well $\tilde{\mathbf{x}}_{k-N} = \mathbf{A} \otimes \hat{\mathbf{x}}_{k-N-1,k-1} \oplus \mathbf{B} \otimes \mathbf{u}_{k-N-1}$. Due to applying (imax,+), it is also convenient to define an interval matrix:

Definition 1. Let $\underline{\mathbf{A}}, \overline{\mathbf{A}} \in \mathbb{R}_\varepsilon^{m \times n}$. The interval matrix is composed as follows $\mathbf{A} = [\underline{\mathbf{A}}, \overline{\mathbf{A}}] = \{\mathbf{M} \in \mathbb{R}_\varepsilon^{m \times n} : \underline{\mathbf{A}} \leq \mathbf{M} \leq \overline{\mathbf{A}}\}$, where matrices $\underline{\mathbf{A}}$ and $\overline{\mathbf{A}}$ contain the lower and upper bound of $\mathbf{A} \in \mathbb{I}\mathbb{R}_\varepsilon^{m \times n}$, respectively.

The observation vector \mathbf{y}_{k-N}^k , over a given time interval $[k - N, k]$, can be written as follows:

$$\mathbf{y}_{k-N}^k = \mathbf{F} \otimes \mathbf{x}_{k-N,k} \oplus \mathbf{H} \otimes \mathbf{u}_{k-N}^{k-1} \tag{13}$$

where

$$\mathbf{u}_{k-N}^{k-1} = \begin{bmatrix} \mathbf{u}_{k-N} \\ \mathbf{u}_{k-N+1} \\ \vdots \\ \mathbf{u}_{k-1} \end{bmatrix}, \quad \tilde{\mathbf{y}}_{k-N}^k = \begin{bmatrix} \mathbf{y}_{k-N} \\ \mathbf{y}_{k-N+1} \\ \vdots \\ \mathbf{y}_k \end{bmatrix},$$

Using (11)–(12) it can be show that:

$$\mathbf{F} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C} \otimes \mathbf{A} \\ \mathbf{C} \otimes \mathbf{A}^{\otimes 2} \\ \vdots \\ \mathbf{C} \otimes \mathbf{A}^{\otimes N} \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \varepsilon & \varepsilon & \dots & \varepsilon \\ \mathbf{C} \otimes \mathbf{B} & \varepsilon & \dots & \varepsilon \\ \mathbf{C} \otimes \mathbf{A} \otimes \mathbf{B} & \dots & \varepsilon \dots & \varepsilon \\ \vdots & \ddots & \ddots & \varepsilon \\ \mathbf{C} \otimes \mathbf{A}^{\otimes N-1} \otimes \mathbf{B} \dots \mathbf{C} \otimes \mathbf{A} \otimes \mathbf{B} \mathbf{C} \otimes \mathbf{B} \end{bmatrix}, \tag{14}$$

where the \mathbf{F} and \mathbf{H} are in $\mathbb{I}\mathbb{R}_\varepsilon$.

Thus, the elements of $\hat{\mathbf{y}}_{k-N}^k$ are represented by intervals and can be written as follows:

$$\mathbf{y}_{k-N}^k = [\underline{\mathbf{y}}_{k-N}^k, \overline{\mathbf{y}}_{k-N}^k], \tag{15}$$

$$\hat{\mathbf{y}}_{k-N}^k = [\underline{\hat{\mathbf{y}}}_{k-N}^k, \hat{\overline{\mathbf{y}}}_{k-N}^k]. \tag{16}$$

Therefore (13) can be split into

$$\hat{\overline{\mathbf{y}}}_{k-N}^k = \overline{\mathbf{F}} \otimes \hat{\mathbf{x}}_{k-N,k} \oplus \overline{\mathbf{H}} \otimes \mathbf{u}_{k-N}^{k-1}, \tag{17}$$

$$\underline{\hat{\mathbf{y}}}_{k-N}^k = \underline{\mathbf{F}} \otimes \hat{\mathbf{x}}_{k-N,k} \oplus \underline{\mathbf{H}} \otimes \mathbf{u}_{k-N}^{k-1} \tag{18}$$

Similarly, the state prediction can be formulated in an interval fashion as

$$\tilde{\mathbf{x}}_{k-N} = [\underline{\tilde{\mathbf{x}}}_{k-N}, \tilde{\overline{\mathbf{x}}}_{k-N}], \tag{19}$$

with lower and upper bounds

$$\tilde{\mathbf{x}}_{k-N} = \overline{\mathbf{A}} \otimes \hat{\mathbf{x}}_{k-N-1,k-1} \oplus \overline{\mathbf{B}} \otimes \mathbf{u}_{k-N-1}, \tag{20}$$

$$\tilde{\mathbf{x}}_{k-N} = \underline{\mathbf{A}} \otimes \hat{\mathbf{x}}_{k-N-1,k-1} \oplus \underline{\mathbf{B}} \otimes \mathbf{u}_{k-N-1} \tag{21}$$

leading to the interval state estimate

$$\hat{\mathbf{x}}_{k-N} = [\hat{\underline{\mathbf{x}}}_{k-N}, \hat{\overline{\mathbf{x}}}_{k-N}], \tag{22}$$

which has to be determined. In order to obtain the above estimate, it is necessary to define the output error

$$\mathbf{e}_{k-N}^k = [\underline{\mathbf{e}}_{k-N}^k, \overline{\mathbf{e}}_{k-N}^k] = \mathbf{y}_{k-N}^k - \hat{\mathbf{y}}_{k-N}^k, \tag{23}$$

$$\mathbf{y}_{k-N}^k - \hat{\mathbf{y}}_{k-N}^k = [\mathbf{y}_{k-N}^k - \hat{\mathbf{y}}_{k-N}^k, \mathbf{y}_{k-N}^k - \underline{\hat{\mathbf{y}}}_{k-N}^k], \tag{24}$$

and the prediction error

$$\mathbf{e}_{k-N} = [\underline{\mathbf{e}}_{k-N}, \overline{\mathbf{e}}_{k-N}] = \hat{\mathbf{x}}_{k-N,k} - \tilde{\mathbf{x}}_{k-N}, \tag{25}$$

$$\hat{\mathbf{x}}_{k-N,k} - \tilde{\mathbf{x}}_{k-N} = [\hat{\underline{\mathbf{x}}}_{k-N,k} - \tilde{\mathbf{x}}_{k-N}, \hat{\overline{\mathbf{x}}}_{k-N,k} - \tilde{\mathbf{x}}_{k-N}]. \tag{26}$$

Having in mind the fact that the estimate should be a possibly thight interval, the cost function can be defined as follows:

$$J = \|\hat{\mathbf{x}}_{k-N,k} - \tilde{\mathbf{x}}_{k-N}\|_Q^2 + \|\mathbf{y}_{k-N}^k - \hat{\mathbf{y}}_{k-N}^k\|_V^2 \tag{27}$$

where the positive definite matrices Q and V can be used as design parameters. Assuming a diagonal form of weighting matrices, the cost function can be described using the following quadratic forms:

$$\mathbf{e}_{k-N}^{kT} Q \mathbf{e}_{k-N}^k = \left[\sum_{i=1}^{m \cdot N} q_i (\underline{\mathbf{e}}_{k-N,i}^k)^2, \sum_{i=1}^{m \cdot N} q_i (\overline{\mathbf{e}}_{k-N,i}^k)^2 \right] \tag{28}$$

where $Q = \text{diag}(q_1, q_2, \dots, q_{m \cdot N})$ and

$$\mathbf{e}_{k-N}^T V \mathbf{e}_{k-N} = \left[\sum_{i=1}^n v_i \underline{\mathbf{e}}_{k-N,i}^2, \sum_{i=1}^n v_i \overline{\mathbf{e}}_{k-N,i}^2 \right] \tag{29}$$

where $V = \text{diag}(v_1, v_2, \dots, v_n)$. Having in mind that the intervals of the output and prediction errors should be as thight as possible, using (28)–(29) the cost function (27) can be refoemulated as follows:

$$J = \sum_{i=1}^n v_i (\overline{\mathbf{e}}_{k-N,i}^2 + \underline{\mathbf{e}}_{k-N,i}^2) + \sum_{j=1}^{m \cdot N} q_j \left((\overline{\mathbf{e}}_{k-N,j}^k)^2 + (\underline{\mathbf{e}}_{k-N,j}^k)^2 \right) \tag{30}$$

Since the quadratic cost function (30) is defined, it is possible to introduce the constraints, which govern the system behavior. Indeed, since all states represent given times let the set P_r contain all pairs (j, r) , which determine the order of individual operations. This means that j th operation precedes r th one in the production route. Thus, the resulting constrains are:

- the precedence of operations, means that there are operations, which have to be finished before the subsequent operation begins

$$x_{j,k} \leq x_{r,k} \quad \forall (j, r) \in P_r \tag{31}$$

- the cycle duty, the operation on a given product needs to be finished, before the same operation on a new instance will begin

$$x_{j,k} \leq x_{j,k+1} \quad \forall j = 1, 2, \dots, n, \quad k = 1, 2, \dots \tag{32}$$

Finally, the estimation problem can be defined as follows: given the pair $(\tilde{x}_{k-N}, y_{k-N}^k)$, find the optimal estimate $\hat{x}_{k-N,k}^*$, which minimizes the cost (27) under constraints (31)–(32), where N is prediction horizon. It is also obvious the above optimisation problem can be solved with conventional efficient quadratic programming solvers.

To summarize, the state estimation algorithm has the following structure: Hav-

Algorithm 1: (max,+) receding horizon state estimation

Step 0: Select \hat{x}_0

Step 1: At any time $k = N + 1, N + 2, \dots$ obtain the prediction \tilde{x}_{k-N} by means of (11) and $\hat{x}_{k-N-1,k-1}^*$ i.e.

$$\begin{aligned} \tilde{\mathbf{x}}_{k-N} &= A \otimes \hat{\mathbf{x}}_{k-N-1,k-1} \oplus B \otimes \mathbf{u}_{k-N-1} \\ \tilde{\mathbf{x}}_{k-N} &= A \otimes \hat{\mathbf{x}}_{k-N-1,k-1} \oplus B \otimes \mathbf{u}_{k-N-1} \end{aligned}$$

Step 2: Solve the problem

$$\hat{\mathbf{x}}_{k-N,k}^* = \arg \min_{\hat{\mathbf{x}}_{k-N,k}} J$$

Step 3: Set $k = k + 1$ and go to *Step 1*.

ing the state estimation algorithm it is possible to examine its performance with a real-life example production system.

4 Example: Battery Assembly System

A flexible Battery Assembly System (BAS) will be introduced for high volume serial production system containing two assembly cycles. The sequence of the first production cycle starts with the robots: 1 and 2 in a starting setting. A robot 1 goes to the frame storage to pick up an empty battery module frame. Then, the battery module controller is mounted into this frame. At the same time, the robot of type 2 provides the appropriate number of basic cells. The type 1 robot transports the final assembly of the battery module to a meeting position with the type 3 robot. After transferring the module to the type 3 robot, the robot of

type 1 returns to its starting position. A robot of type 3 receives the assembled battery module from the robot of type 1. Then it transports this module to the final assembly system. The robot of type 4 picks up additional wiring. Then it moves to the storage of the rack housings to pick up the housings and bring them to the assembly system. Subsequently, the robot of type 3 returns to the rendez-vous position (with the robot of type 1) and the robot of type 4 brings the fully assembled rack to the final storage and returns to its initial position. Due to the size of system, only the first production cycle (Fig. 1) is considered. The processing times and transportation times from one to another station are clearly depicted in (Fig. 1) For the interval processing and transportation times

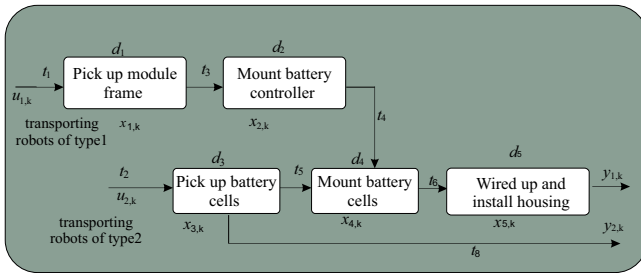


Fig. 1. Details of the assembly process cycle 1

described in the Table 1, the system matrices are given below. The initial state for the system is $\mathbf{x}_0 = [4, 12, 4, 19, 31]^T$ while the one for the observer is $\hat{\mathbf{x}}_0 = [3, 14, 2, 21, 34]^T$. All output measurements are portrayed in Fig. 4.

Table 1. Nominal and interval processing and transportation times for the battery assembly system

	Nominal time	Interval time [min]		Nominal time	Interval time [min]
d_1	6	[4,7]	d_2	3	[3,4]
d_3	5	[4,6]	d_4	8	[6,10]
d_5	3	[3,4]	t_1	4	[3,5]
t_2	4	[3,5]	t_3	2	[2,3]
t_4	4	[3,5]	t_5	4	[3,5]
t_6	4	[3,5]			

$$\mathbf{A} = \begin{bmatrix} [4, 7] & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ [10, 17] & [3, 4] & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & [4, 6] & \varepsilon & \varepsilon \\ [16, 26] & [9, 13] & [11, 17] & [6, 10] & \varepsilon \\ [25, 41] & [18, 28] & [20, 32] & [15, 25] & [3, 4] \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} [3, 5] & \varepsilon \\ [9, 15] & \varepsilon \\ \varepsilon & [3, 5] \\ [15, 24] & [10, 16] \\ [24, 39] & [19, 31] \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} \varepsilon & \varepsilon & [4, 6] & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & [3, 4] \end{bmatrix}$$

Having the initial estimates and the output measurements collected within a moving window, it is possible to apply *Algorithm 1*.

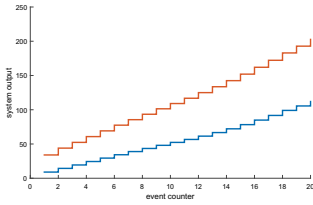


Fig. 2. Output measurements

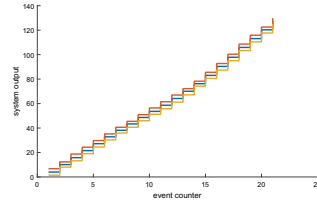


Fig. 3. Real state x_1 and its interval estimate

As a result the interval estimates are obtained (see Figs. 3– 5). For the purpose of experimental comparison the real states x_1 , x_2 and x_4 were measured as well.

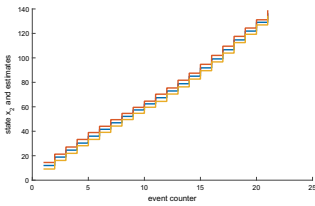


Fig. 4. Real state x_2 and its interval estimate

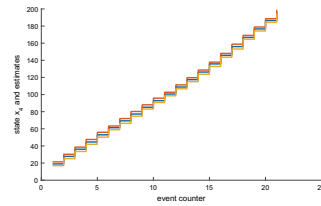


Fig. 5. Real state x_4 and its interval estimate

The obtained results clearly show that the interval estimates bounds the real states with a decent quality.

5 Conclusion

The main objective of this paper was to provide a novel robust state estimation algorithm for a class of DES. For that purpose the $(\max,+)$ algebra was employed and suitably extended with interval arithmetics. As a result, an interval DES description was proposed. Subsequently, the state estimation problem was defined as a receding-horizon one and a suitable cost function was developed and formed in a quadratic form. Moreover, the optimization problem was accompanied with production constraints. The proposed approach was applied to the real battery assembly system. The obtained results clearly show that the obtained estimates closely overbound the real states. This recommends its practical application. The objective of the subsequent research is to use the proposed approach

along with the existing control strategy. However, this requires the development of *separation principle* for DES described with $(imax,+)$.

References

1. Baccelli, F., Cohen, G., Olsder, G.J., Quadrat, J.P.: Synchronization and linearity: an algebra for discrete event systems. *JOURNAL-OPERATIONAL RESEARCH SOCIETY* 45, 118–118 (1994)
2. Butkovic, P.: *Max-linear systems: theory and algorithms*. Springer (2010)
3. Cechlářová, K.: Eigenvectors of interval matrices over max-plus algebra. *Discrete applied mathematics* 150(1), 2–15 (2005)
4. Elbanhawi, M., Simic, M., Jazar, R.: Receding horizon lateral vehicle control for pure pursuit path tracking. *Journal of Vibration and Control* p. 1077546316646906 (2016)
5. Hardouin, L., Maia, C., Cottenceau, B., Mendes, R.S.: Max-plus linear observer: Application to manufacturing systems. *IFAC Proceedings Volumes* 43(12), 161 – 166 (2010)
6. Heidergott, B., Olsder, G., van der Woude, J.: *Max Plus at Work: Modeling and Analysis of Synchronized Systems: A Course on Max-Plus Algebra and Its Applications*. Princeton Series in Applied Mathematics, Princeton University Press (2014), <https://books.google.pl/books?id=yPocBAAAQBAJ>
7. Ling, K.V., Lim, K.W.: Receding horizon recursive state estimation. *IEEE Transactions on Automatic Control* 44(9), 1750–1753 (1999)
8. Majdzik, P., Akielaszek-Witczak, A., Seybold, L., Stetter, R., Mrugalska, B.: A fault-tolerant approach to the control of a battery assembly system. *Control Engineering Practice* 55, 139–148 (2016)
9. Mrugalski, M.: *Advanced neural network-based computational schemes for robust fault diagnosis*, ISBN: 978-3-319-01546-0. Studies in Computational Intelligence, Vol. 510, Springer-Verlag, Berlin - Heidelberg (2014)
10. Polak, M., Majdzik, P., Banaszak, Z., Wójcik, R.: The performance evaluation tool for automated prototyping of concurrent cyclic processes. *Fundamenta Informaticae* 60(1-4), 269–289 (2004)
11. Puig, V.: Fault diagnosis and fault tolerant control using set-membership approaches: Application to real case studies. *International Journal of Applied Mathematics and Computer Science* 20(4), 619–635 (2010)
12. Seybold, L., Witczak, M., Majdzik, P., Stetter, R.: Towards robust predictive fault-tolerant control for a battery assembly system. *International Journal of Applied Mathematics and Computer Science* 25(4), 849–862 (2015)
13. Theilliol, D., Join, C., Zhang, Y.: Actuator fault tolerant control design based on a reconfigurable reference input. *International Journal of Applied Mathematics and Computer Science* 18(4), 553–560 (2008)
14. Witczak, P., Luzar, M., Witczak, M., Korbicz, J.: A robust fault-tolerant model predictive control for linear parameter-varying systems. In: *Proceedings of Methods and Models in Automation and Robotics - MMAR*. pp. 462–467 (2014)
15. Yoon, T.W., Clarke, D.W.: Observer design in receding-horizon predictive control. *International Journal of Control* 61(1), 171–191 (1995)

Defining the Optimal Number of Actuators for Active Device Noise Reduction Applications

Stanisław Wrona, Krzysztof Mazur and Marek Pawełczyk

Institute of Automatic Control, Silesian University of Technology
Akademicka 16, 44-100 Gliwice, Poland,
stanislaw.wrona@polsl.pl

Abstract. It is possible to improve sound insulation of devices and machinery from the environment by appropriately controlling vibrations of their casings. When implementing this approach, there is a need to select efficient locations of actuators on each of the casing walls. A common solution is to employ an optimization algorithm to find favourable arrangement according to a given objective. One of the essential parameters of this process is a number of actuators to be optimized, but most often it is assumed a priori, without any proper considerations.

The aim of this paper is to give an insight into the problem of defining the number of actuators for active noise reduction implementations. The optimal value depends on the particular application, its mechanical limitations, costs, considered frequency band to be controlled, etc. However, a general approach for analysis and decision making can be formulated. As an exemplary structure, a light-weight device casing is considered. The relationships between the number of actuators, the considered frequency band, and obtained values of the optimization index are given.

Keywords: Active casing, actuators placement, controllability Gramian, optimization, memetic algorithm.

1 Introduction

Some of the most common noise sources in the human environment are devices and machinery. Prolonged exposure to a high-level noise (e.g. in industrial halls, factories, etc.) may lead to hearing losses and health problems. On the other hand, noise generated by home appliances does not represent a health threat, but may successfully obstruct work or leisure. Passive sound-insulating materials are commonly applied to reduce the excessive device noise, however, they are ineffective for low frequencies and often are inapplicable due to increase of size and weight of the device and its potential overheating. When passive methods are exhausted, alternatively, active control methods can be applied. Sound radiation and transmission through individual elastic plates and other barriers have been investigated over the years [4, 21].

Sound insulation of devices and machinery from the environment can be improved with active methods by appropriately controlling vibrations of their

casings. Such approach is called the active casing method and its acoustic isolation efficiency has been confirmed by the authors for several laboratory casings in previous publications [8, 9, 12, 13]. Active methods efficiently complement the passive methods in their weak points—low-frequency noise and heat transfer problems. When appropriately implemented, it results in a global noise reduction instead of only local zones of quiet at distinguished areas [11]. It neither requires structural modifications of the device nor affects its regular operations, but it allows to enclose the source of noise inside the casing and isolates it acoustically from the environment. It has been also observed that for an effective active control it is crucial to select appropriate locations of actuators on each of the casing walls. The efficiency of selected arrangement determines the overall control performance of the system. A common solution is to employ an optimization algorithm to find favourable arrangement according to a given objective. The optimization can include a predefined control strategy and related control performance index [5, 7], or an open-loop system analysis can be utilized, making the obtained results independent on controller choice [1, 16, 17].

One of the essential parameters of such process is a number of actuators to be optimized. Gao et al. studied the influence of actuators number on a vibrating plate control system [2]. Yan and Yam applied the eigenvalue distribution of the energy correlative matrix of the control input force to determine the number of actuators required [20]. Xu et al. presented an integrated optimization of trusses structural topology with number and placement of piezoelectric actuators [18, 19]. Li and Huang employed a penalty function method to include actuators number optimization in an unconstrained minimization problem [6]. However, only a limited number of reports available in the literature discuss the number of actuators. There is no unified approach to determine the number of actuators and most often it is assumed a priori, without any proper considerations.

The aim of this paper is to give an insight into the problem of defining the number of actuators for active noise reduction implementations basing on the controllability-oriented criteria. The optimal value depends on the particular application, its mechanical limitations, costs, considered frequency band to be controlled, etc. However, a general approach for analysis and decision making can be formulated. The presented considerations are based on experimental vibration measurements of the structure, its mathematical modelling and extensive simulation studies.

In this paper, as an exemplary structure, a light-weight device casing is considered, which is described in Section 2. Then, the mathematical model of the structure is briefly introduced in Section 3. It is employed for numerical simulations and assessment of a particular inertial actuators arrangement. Subsequently, the optimization process and results are given in Section 4. A memetic algorithm is used for the purpose of optimization itself [10]. Controllability measure of the modelled control system is used as optimization index. The relationships between the number of actuators, the considered frequency band, and obtained values of the optimization index are given. Finally, advantages and limits of the proposed approach are pointed out and discussed.

2 Vibrating structure

The light-weight device casing used as an exemplary structure in this work is shown in Fig. 1. It is made of 1.5 mm thick steel plates bolted together, forming a closed cuboid of dimensions 500 mm \times 630 mm \times 800 mm. The casing is made without an explicit frame, hence it constitutes a self-supporting structure. It results in strong couplings between individual walls of both vibrational and acoustical nature. However, what has been previously validated in [15], observed natural frequencies and modeshapes of the whole structure are a consequence of superposition of resonances of each wall excited individually (but as a part of the whole structure). Therefore, it is justified to analyse each wall separately for the purpose of optimization of actuators locations, considering only eigenmodes due to the given wall (if the resonance is controlled with actuators at the wall where it originates, it will be reduced for the whole casing).

Although the following considerations will be shown on the example of this particular casing, the presented approach and drawn conclusions remain general and they can be applied for a number of different structures.



(a) A photograph from the inside.



(b) A photograph from the outside.

Fig. 1: Photographs of the light-weight casing with sensors and actuators.

3 Mathematical model

For the purpose of actuators arrangement optimization, a precise mathematical model of the plant is the crucial component. It enables numerical simulations and analysis to assess particular actuators locations (providing an objective numerical index describing the control efficiency of a given arrangement).

As mentioned above, each of the casing walls is considered separately, hence a mathematical model of an individual vibrating plate is employed. It is based on the Mindlin plate theory. The inertial actuators are considered in this research,

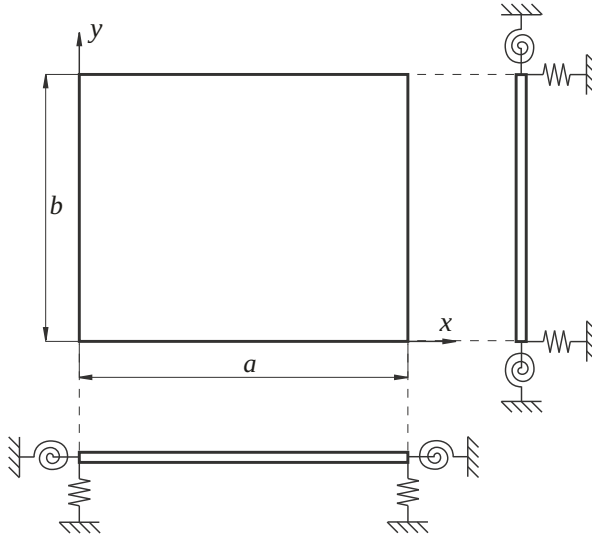


Fig. 2: A multiview orthographic projection of the plate with boundary conditions represented as rotational and translational springs.

which possess a considerable mass comparing to the mass of the casing walls (each actuator weights 0.115 kg). Hence, the model includes the impact of additional loading due to mounted actuators. It is noteworthy, that it constitutes the major computational difficulty, because it makes necessary to recalculate the model for each actuators arrangement and increases the computational effort of the whole process. The model solution is based on the Rayleigh-Ritz assumed mode-shape method. Characteristic orthogonal polynomials having the property of Timoshenko beam functions, which satisfy edge constraints, are used. Moreover, due to the absence of any stiffening frame, the casing walls are connected directly to each other, what results in boundary conditions that have to be modelled as elastically restrained against both rotation and translation. The model is presented schematically in Fig. 2.

Employing the Rayleigh-Ritz method, the model can be written in the regular state-space form:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad (1a)$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad (1b)$$

where \mathbf{x} and \mathbf{y} are the state vector and the output vector, respectively; and \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} are matrices describing the system. The state vector \mathbf{x} is formulated in such a way that its following elements correspond to respective eigenmodes of the casing wall. It is a direct consequence of the Rayleigh-Ritz method, where the infinite dimensional system of the vibrating plate is approximated with a finite dimensional system (taking into account only a limited number of initial

eigenmodes, making the model valid for a limited frequency range). However, taking advantage of the state space notation, classical methods can be used to describe the controllability of the system [3]. Namely, the controllability Gramian matrix is employed in this research. It is convenient to use due to its specific feature—if the i -th value at the diagonal of the Gramian matrix, λ_i , corresponding to the i -th eigenmode is small, the eigenmode is difficult to control (it can be regulated only if a large control energy is available). Such information is used in the following Section to define a criterion in the optimization process of actuators placement. Formally, controllability is a dichotomous property, but "controllable" does not say how high control effort is needed to reach the final state.

In this paper, the description of the model is very brief, as it focuses mainly on the model application for actuators arrangement optimization rather than the mathematical modelling itself. However, a detailed derivation of the model is given in [16].

4 Optimization process, results and discussion

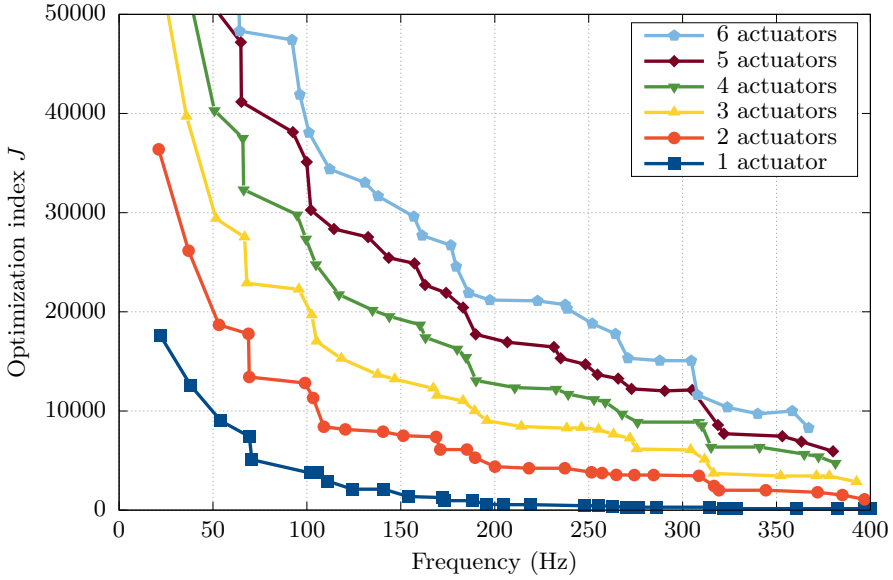
For the purpose of the process described in this Section it is assumed that the employed mathematical model has properly identified parameters and is successfully validated for the given structure (the process of identification and validation has been discussed in details in [14]). Then, such model can be utilized to optimize actuators arrangement in the view of a given objective function.

In this research, such process is performed for several predefined numbers of actuators, N_a , to provide an insight into the consequences of different values of N_a . A straightforward declaration of N_a as one of the optimization variables usually results in a solution with a maximum allowed number of actuators (a greater number of actuators generally facilitates the control task and increases the optimization index value). To mitigate this feature, a penalty function can be added to the index in order to limit the N_a , but it is not a trivial task to properly balance such multi-objective problem. The selection of weights in the optimization index determines the resulting number of actuators, what actually is equivalent to manual selection of its value. The general trends pointed out in this Section can provide a basis for both ways mentioned above: (i) manual selection of actuators number and (ii) the desired balance of different objectives in the optimization index.

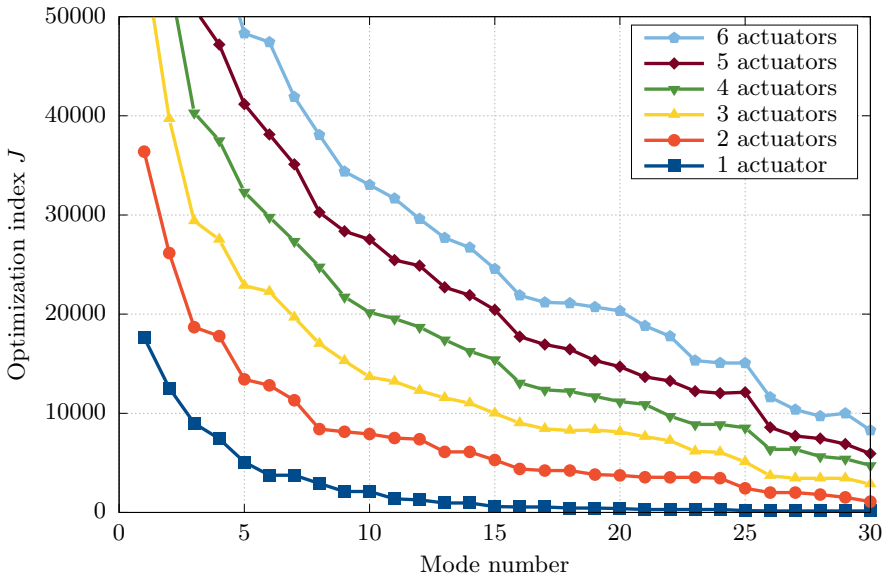
4.1 Optimization process of actuators arrangement

The optimization itself is performed with a memetic algorithm. The optimization variables are the coordinates of actuators on the plate surface (number of actuators is predefined). The optimization index J selected in this research is a measure of controllability of the least controllable mode:

$$J = \min_i \lambda_i , \quad (2)$$



(a) A plot with respect to eigenfrequencies.



(b) A plot with respect to mode numbers.

Fig. 3: Plots representing relationships between the number of actuators in the optimized arrangement, the considered frequency band (a number of eigenmodes), and obtained values of the optimization index.

where λ_i is the i -th diagonal element of the controllability Gramian matrix, as defined in the previous Section. Such optimization index represents an objective to guarantee controllability in the given frequency range (in other words, to avoid any uncontrollable eigenmodes in the frequency range of interest). More detailed description of the optimization process is given e.g. in [17].

Depending on the modal density of frequency responses of each casing wall, different number of eigenmodes should be taken into account in the optimization index in particular application, e.g. if the range up to 300 Hz is considered, then there should be taken into account 25 modes for top wall, 21 modes for front and back wall, 17 modes for left and right wall. For the sake of brevity, optimization results are shown and discussed only for the top wall. However, similar results were obtained for the remaining walls.

4.2 Optimization results and discussion

The relationships between the considered frequency band (or a number of eigenmodes taken into account) and obtained values of the optimization index are given in Fig. 3. Different controllability curves are plotted for several predefined numbers of actuators (up to six actuators). Up to 30 initial eigenmodes are considered, resulting in frequency range up to 400 Hz. The obtained optimization index values are suboptimal, as the memetic algorithm does not guarantee to find the global optimum. However, the presented values are a result of multiple runs of the algorithm with highly consistent results. Basing on this plots, several general conclusion can be drawn.

Although the optimization index values are dimensionless, a reference point can be established to interpret the obtained controllability measures. Such reference point can be defined as the result obtained for a single actuator and only one eigenmode considered (the fundamental frequency). For this scenario, the actuator is always placed at the centre of the casing wall (at its anti-nodal point for the first mode), and it can be assumed that such value represents a satisfactory level of controllability of a given mode. If more eigenmodes are taken into account, then the employed controllability measure drops rapidly. It means that if the number of considered eigenmodes exceeds the number of actuators, then the optimization algorithm is selecting trade-off locations to preserve controllability of all considered eigenmodes. However, it also means that at some point, actuators locations are far from any of the anti-nodal points, and none of the modes is controllable at a satisfactory level. Then, more actuators should be applied.

It follows from the analysis of Fig. 3 that if two actuators are selected to control the initial four eigenmodes, the controllability measure can be guaranteed at a level equal or greater as for the reference level of one actuator controlling only the first mode. Extending this approach, three actuators would be recommended to control up to 8 eigenmodes, four actuators for up to 13 eigenmodes, etc. In such a way, a reasonable controllability level would be preserved for a given frequency range of interest.

On the other hand, results obtained for a particular number of eigenmodes can also be analysed, e.g. results obtained for 10 eigenmodes. As it is shown in Fig. 3, an increase of the number of actuators from one to two, improves dramatically the controllability of the least controllable mode—approximately four times. Further increase to three actuators doubles the controllability measure, but the relative improvement is not as great as before. Difference between three and four actuators is even lower. It shows that the relative gain by adding another actuator is becoming lower with each following actuator. This can also constitute a point for decision making when selecting the number of actuators.

It is also noteworthy that employment of such approach and optimization index makes it easy to compare obtained controllability levels for each of the casing walls. Therefore, the actuators numbers for different walls can be selected in such a way that the achieved controllability levels are of similar order, resulting in a well-balanced control system.

5 Conclusion

The problem of defining the number of actuators for active noise reduction implementations has been considered in this paper. As the exemplary structure, a light-weight device casing has been utilised. The mathematical model of an individual casing wall has been used in the optimization process. The relationships between the number of actuators, the considered frequency band, and obtained values of the optimization index have been presented.

Although the optimization index values are dimensionless, a reference point has been proposed to interpret and compare the obtained controllability measures. General trends that have been pointed out can provide a basis for both the manual selection of actuators number and the definition of properly balanced multi-objective optimization index.

Acknowledgement

The research reported in this paper has been supported by the National Science Centre, Poland, decision no. DEC-2014/13/B/ST7/00755, and the Ministry for Higher Education and Science.

References

1. Bruant, I., Gallimard, L., Nikoukar, S.: Optimal piezoelectric actuator and sensor location for active vibration control, using genetic algorithm. *Journal of Sound and Vibration* 329(10), 1615–1635 (2010)
2. Gao, F., Shen, Y., Li, L.: The optimal design of piezoelectric actuators for plate vibroacoustic control using genetic algorithms with immune diversity. *Smart Materials and Structures* 9(4), 485 (2000)
3. Klamka, J.: Controllability of dynamical systems. a survey. *Bulletin of the Polish Academy of Sciences: Technical Sciences* 61(2), 335–342 (2013)

4. Klamka, J., Wyrwal, J., Zawiski, R.: Mathematical model of the state of acoustic field enclosed within a bounded domain. In: *Methods and Models in Automation and Robotics (MMAR)*, 2015 20th International Conference on. pp. 191–194. IEEE (2015)
5. Kumar, K.R., Narayanan, S.: The optimal location of piezoelectric actuators and sensors for vibration control of plates. *Smart Materials and Structures* 16(6), 2680 (2007)
6. Li, W., Huang, H.: Integrated optimization of actuator placement and vibration control for piezoelectric adaptive trusses. *Journal of Sound and Vibration* 332(1), 17–32 (2013)
7. Liu, W., Hou, Z., Demetriou, M.A.: A computational scheme for the optimal sensor/actuator placement of flexible structures using spatial h2 measures. *Mechanical systems and signal processing* 20(4), 881–895 (2006)
8. Mazur, K., Pawelczyk, M.: Internal model control for a light-weight active noise-reducing casing. *Archives of Acoustics* 41(2), 315–322 (2016)
9. Mazur, K., Pawelczyk, M.: Virtual microphone control for a light-weight active noise-reducing casing. In: *Proceedings of 23th International Congress on Sound and Vibration*. vol. 24, p. 411284 (2016)
10. Nalepa, J., Blocho, M.: Adaptive memetic algorithm for minimizing distance in the vehicle routing problem with time windows. *Soft Computing* 20(6), 2309–2327 (2016)
11. Wiora, J., Wrona, S., Pawelczyk, M.: Evaluation of measurement value and uncertainty of sound pressure level difference obtained by active device noise reduction. *Measurement* 96, 67–75 (2017)
12. Wrona, S., Pawelczyk, M.: Active reduction of device multi-tonal noise by controlling vibration of multiple walls of the device casing. In: *Proceedings of 19th International Conference On Methods and Models in Automation and Robotics (MMAR)*, IEEE. Miedzyzdroje, Poland, 2-5 September (2014)
13. Wrona, S., Pawelczyk, M.: Feedforward control of a light-weight device casing for active noise reduction. *Archives of Acoustics* 41(3), 499–505 (2016)
14. Wrona, S., Pawelczyk, M.: Identification of elastic boundary conditions of light-weight device casing walls using experimental data. In: *Methods and Models in Automation and Robotics (MMAR)*, 2016 21st International Conference on. pp. 212–217. IEEE (2016)
15. Wrona, S., Pawelczyk, M.: Optimal placement of actuators for active structural acoustic control of a light-weight device casing. In: *Proceedings of 23rd International Congress on Sound and Vibration*. Athens, Greece, 10-14 July (2016)
16. Wrona, S., Pawelczyk, M.: Shaping frequency response of a vibrating plate for passive and active control applications by simultaneous optimization of arrangement of additional masses and ribs. Part I: Modeling. *Mechanical Systems and Signal Processing* 70-71, 682–698 (2016)
17. Wrona, S., Pawelczyk, M.: Shaping frequency response of a vibrating plate for passive and active control applications by simultaneous optimization of arrangement of additional masses and ribs. Part II: Optimization. *Mechanical Systems and Signal Processing* 70-71, 699–713 (2016)
18. Xu, B., Jiang, J., Ou, J.: Integrated optimization of structural topology and control for piezoelectric smart trusses using genetic algorithm. *Journal of Sound and Vibration* 307(3), 393–427 (2007)
19. Xu, B., Ou, J., Jiang, J.: Integrated optimization of structural topology and control for piezoelectric smart plate based on genetic algorithm. *Finite Elements in Analysis and Design* 64, 1–12 (2013)

20. Yan, Y., Yam, L.: Optimal design of number and locations of actuators in active vibration control of a space truss. *Smart Materials and Structures* 11(4), 496 (2002)
21. Zawieska, W.M., Rdzanek, W.P., Rdzanek, W.J., Engel, Z.: Low frequency estimation for the sound radiation efficiency of some simply supported flat plates. *Acta acustica united with acustica* 93(3), 353–363 (2007)

Part V
Computer Methods in Control Engineering

CPDev engineering environment for control programming

Dariusz Rzońca, Jan Sadolewski, Andrzej Stec, Zbigniew Świder, Bartosz Trybus, and Leszek Trybus

Rzeszów University of Technology
Department of Computer and Control Engineering
al. Powstańców Warszawy 12
35-959 Rzeszów, Poland
{drzonca, js, astec, swiderzb, btrybus, ltrybus}@kia.prz.edu.pl
<http://kia.prz.edu.pl>

Abstract. Programming a control example by CPDev tool using LD, FBD, and SFC graphical languages of IEC 61131-3 standard is presented. Another example demonstrates the use of data structures and arrays of function blocks in ST textual language. The two parts indicate that CPDev provides now essential functionalities specified in the standard. Remarks how the tool relates to established engineering environments such as CoDeSys, STEP 7, Control Builder F, and ISaGRAF are also given.

Keywords: IEC 61131-3, programming languages, engineering tools

1 Introduction

First version of CPDev tool (*Control Program Developer*) for programming PLC/PAC controllers in ST and IL textual languages of IEC 61131-3 standard [1, 2] was developed 10 years ago with support from KBN/MNiSW, in partnership with LUMEL Zielona Góra [3]. By assumption, CPDev was target-independent, so now it can be implemented on AVR, ARM, x86, and FPGA platforms [4]. Over the years experimental editors of LD, FBD, and SFC graphical languages have been added, as well as visualization, SysML modeling, and automated testing [5]. Thus CPDev has grown into a kind of engineering environment, although with strong academic flavour.

During the last two years, motivated by industrial partners, upgraded versions of graphical languages have been developed. Also compiler of ST language has been substantially extended to support advanced programming. Hence the purpose of this paper is to overview current version of CPDev and indicate how it looks like in relation to known engineering environments, such as CoDeSys [6], STEP 7 [7, 8], Control Builder F [9], and ISaGRAF [10].

2 Restart protection example

Turning MOTOR on and off by means of START, STOP pushbuttons with protection against too frequent restarts is a simple example programmed here in LD, FBD, and SFC graphical languages. Restart protection means that after turning MOTOR off another turning on is possible only after some time T . Relevant time plots are shown in Fig. 1a.

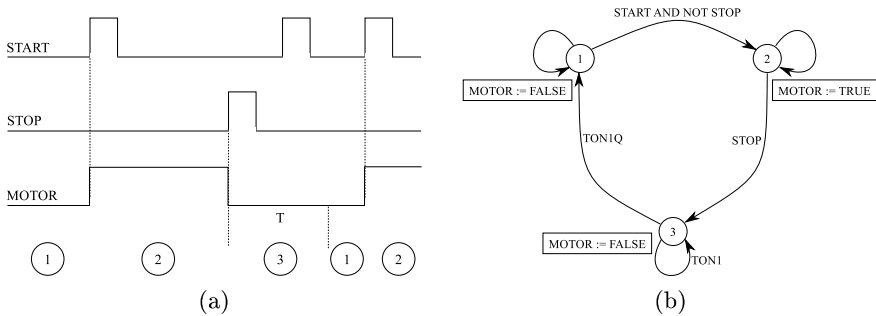


Fig. 1. Restart protection: a) time plots, b) state-machine diagram

Restart protection can be implemented in a few ways. Here state-machine diagram is chosen as most typical for sequential controls. The diagram corresponding to time plots is shown in Fig. 1b, where the states mean: 1 - *ready*, 2 - *running*, 3 - *protection*. ST language notation is used. TON1 denotes an instance of TON timer, with TON1Q output.

3 LD diagram

Main window of CPDev tool for *Restart_LD* project is shown in Fig. 2. Left part presents tree with structure of the project, involving POU's (*Program Organization Units*), global variables, tasks, and libraries. The tree is independent of programming language, which is selected after declaring new POU, i.e. program, function block or function.

The middle and right part of Fig. 2 are components of the inner LD editor window. The tree on the left (middle in Fig. 2) contains tree whose elements can be dragged on the drawing board on the right between vertical lines (power rails). The tree involves:

- basic elements, such as contact, coil, input/output variable, etc.
- a few frequently used blocks and functions
- blocks of *IEC_61131* library in standard form
- blocks of the same library but with additional ENable input, i.e. *with EN*
- *Functions*, *System blocks*, and blocks of *Basic_blocks* library, also *with EN*.

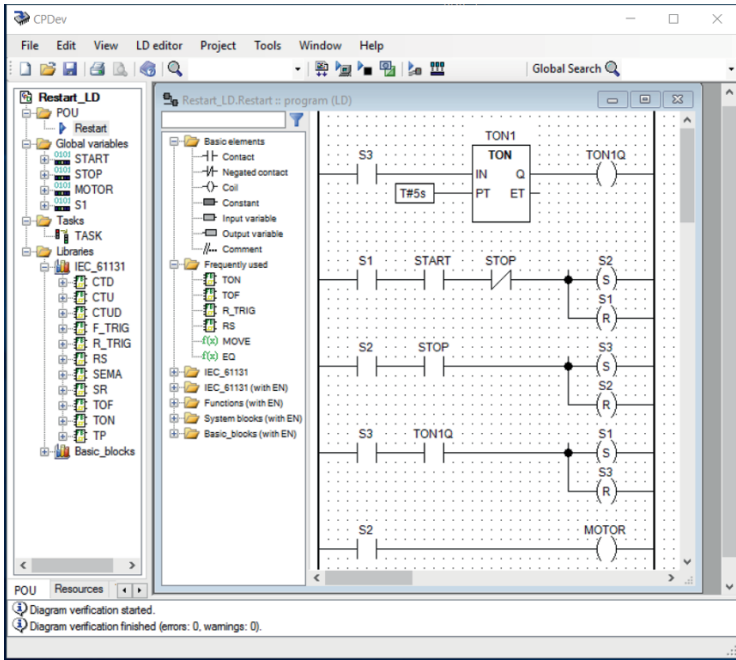


Fig. 2. CPDev window for *Restart_LD* project

Contrary to standard blocks of *IEC_61131* library, all functions and blocks *with EN* are executed only when TRUE appears at EN input (*IEC 61131-3* admits *ENO* output as well [1,2]). This allows to implement in LD language some simple diagrams typical for FBD. *Functions* branch consists of subbranches listed further. *Basic_blocks* library involves special arithmetic blocks, switches, memory blocks, comparator, flip-flop, pulsers, filters, and some nonlinearities. Hence other types than *BOOL* may be also processed. The blocks have been adapted from old software for multifunction controllers [11].

PID control is supported by *Complex_blocks* library (not shown in Fig. 2) which involves PID controller, set-point block, automatic tuning, and a few other blocks, also partly adapted from [11]. Auto-tuning applies relay oscillation method. If needed in particular project, *Complex_blocks* library is imported from external file.

LD diagram of the *Restart* program is on the right. Global variable *S1*, as well as locals *S2*, *S3*, indicate activity of the corresponding states. So the timer *TON1* in the first rung becomes active in the *protection* state *S3* (compare Fig. 1b). Next three rungs with *S* (set) and *R* (reset) coils implement branches of the state-machine with appropriate conditions. Last rung sets *MOTOR* output to TRUE in the state *S2* (*running*). To begin with *ready* state, the variable *S1* must be initialized to TRUE (local variables *S2*, *S3* cannot be initialized by

the editor). Details of creating LD diagrams in CPDev and examples involving blocks *with EN* can be found in [12, 13].

Empty straight line from which creating a rung begins in CoDeSys, STEP 7, Control Builder F, and ISaGRAF environments is the main difference from CPDev. Contacts, coils, and blocks are dropped on the line with connections updated automatically. Additional parallel connections are usually opened downwards (to simplify translation into target language). CoDeSys admits blocks with EN input only (added automatically). STEP 7 (LAD language), Control Builder F, and ISaGRAF do not impose such restrictions, provided that at least one boolean input is connected to a contact. Right power-rail is not shown on Control Builder F drawing board.

We remark that since Rockwell has acquired ISaGRAF some time ago, the website has not been updated. Similarly as CoDeSys, unlike STEP 7 and Control Builder F, ISaGRAF has been used, and still is, by a number of manufacturers (x86 platforms).

4 Compilation and simulation

During compilation the LD diagram is first translated into a program in ST language by introducing local variables, replacing serial and parallel contacts by AND, OR gates, etc. The ST code can be displayed by selecting an editor option. The code, shown in Fig. 3, is split into parts corresponding to declarations and rungs (translated separately).

```

001 PROGRAM Restart
002
003 VAR_EXTERNAL
004   S1: BOOL;
005   START: BOOL;
006   STOP: BOOL;
007   MOTOR: BOOL;
008 END_VAR
009
010 VAR
011   S3: BOOL;
012   S2: BOOL;
013   TONI_Q: BOOL;
014   out_contact_S3_80_70: BOOL;
015   out_contact_S1_80_180: BOOL;
016   out_contact_START_150_180: BOOL;
017   out_contact_STOP_220_180: BOOL;
018   out_contact_S2_80_270: BOOL;
019   out_contact_STOP_150_270: BOOL;
020   out_contact_S3_80_360: BOOL;
021   out_contact_TONI_Q_150_360: BOOL;
022   out_contact_S2_80_460: BOOL;
023   out_bp_260_170: BOOL;
024   out_bp_260_260: BOOL;
025   out_bp_260_350: BOOL;
026   TONI: TON;
027   out_TONI_Q: BOOL;
028 END_VAR
029
030 out_contact_S3_80_70 := S3;
031 TONI(IN:=out_contact_S3_80_70,PT:=T#5s,Q:=out_TONI_Q);
032 TONI_Q := out_TONI_Q;
033
034 out_contact_S1_80_180 := S1;
035 out_contact_START_150_180 := out_contact_S1_80_180 AND START;
036 out_contact_STOP_220_180 := out_contact_START_150_180 AND NOT STOP;
037 out_bp_260_170 := out_contact_STOP_220_180;
038 IF (out_bp_260_170 = TRUE) THEN S2 := TRUE; END_IF;
039 IF (out_bp_260_170 = TRUE) THEN S1 := FALSE; END_IF;
040
041 out_contact_S2_80_270 := S2;
042 out_contact_STOP_150_270 := out_contact_S2_80_270 AND STOP;
043 out_bp_260_260 := out_contact_STOP_150_270;
044 IF (out_bp_260_260 = TRUE) THEN S3 := TRUE; END_IF;
045 IF (out_bp_260_260 = TRUE) THEN S2 := FALSE; END_IF;
046
047 out_contact_S3_80_360 := S3;
048 out_contact_TONI_Q_150_360 := out_contact_S3_80_360 AND TONI_Q;
049 out_bp_260_350 := out_contact_TONI_Q_150_360;
050 IF (out_bp_260_350 = TRUE) THEN S1 := TRUE; END_IF;
051 IF (out_bp_260_350 = TRUE) THEN S3 := FALSE; END_IF;
052
053 out_contact_S2_80_460 := S2;
054 MOTOR := out_contact_S2_80_460;
055
056 END_PROGRAM

```

Fig. 3. ST code translated from LD diagram

In the next stage the ST program is compiled into universal VMASM code executed by runtime virtual machine in the controller [14]. VMASM (*Virtual Machine Assembler*) is an assembly-type language, not related to any particular processor, but oriented much towards IEC_61131-3 standard. The virtual machine is written in C, so it may run on different hardware platforms, from 8-bit microcontrollers up to 32/64-bit general purpose processors and FPGA [4]. With respect to virtual machine concept, CPDev is quite different than CoDeSys which generates native assembly code for several types of processors.

Tracing variable values during execution is provided by CPSim simulator. Variables can be displayed on lists, in control panels involving variables assigned to buttons, LED-type rectangles or numeric cells, and in small individual views. Simulation window for *Restart_LD* project with three types of components is shown in Fig. 4. Variables are placed in the window or on the panels by dragging them from simulation tree. The tree involves global and local variables, and inputs/outputs of function blocks. Although much simpler, CPSim may remind PLCSim tool of STEP 7. Unlike in PLCsim, layout of CPSim window is fairly flexible.

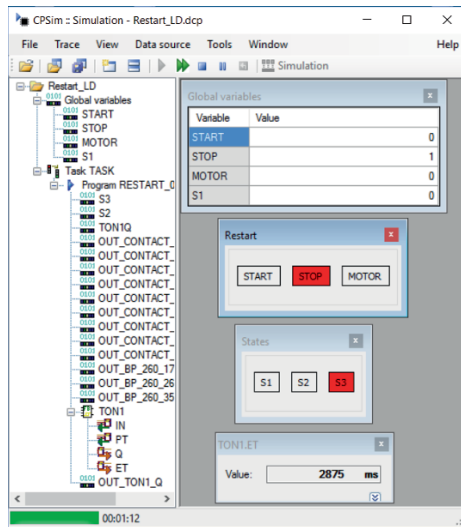


Fig. 4. Simulation window for *Restart_LD* project

5 FBD diagram

Implementation of state-machine in FBD language is not so straightforward as in LD, since counterparts of S, R coils are not available. Hence FBD diagrams

of state-machines are usually not presented in programming instructions and books on PLCs. Anyway, states with conditions for outgoing branches can be implemented as user-defined blocks shown below, and connected together as in the state-machine diagram.

Two such blocks, STATE_C and STATE_T, each implementing one outgoing transition are shown in Figs. 5a,b. Numbers 1, 2 at the bottom of components indicate execution order. The blocks exploit the idea that RS flip-flop with feedback from Q output to R input generates one-cycle pulse, provided that it has been set earlier by S input. In case of STATE_C the pulse appears at the output PL when both condition variables C1, C2 are TRUE, and in case of STATE_T when preset time PT expires. Plots in Fig. 5c demonstrate the operation. The output ST indicates activity of the block's state, i.e. when the flip-flop is set. Fig. 5d shows translation of one of the diagrams into ST language.

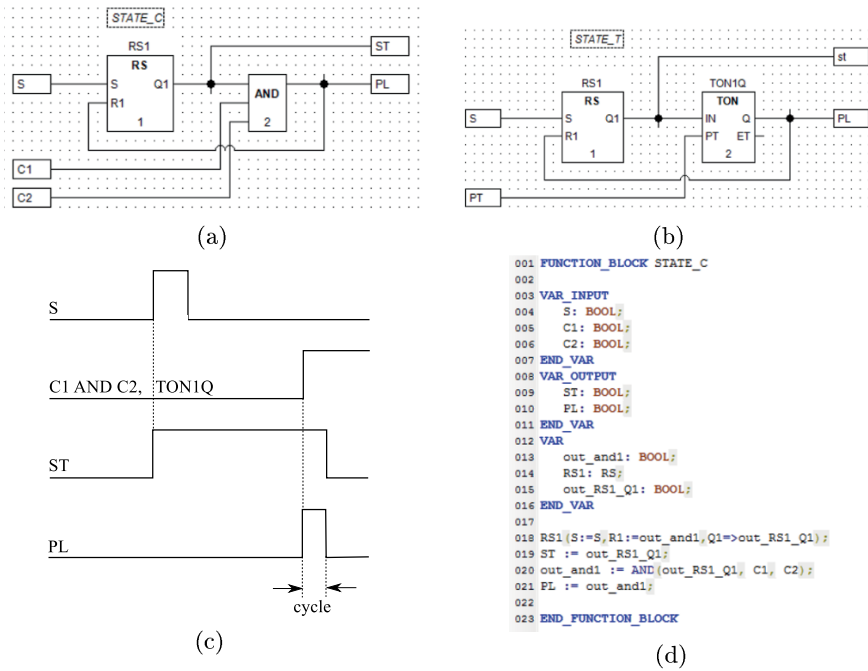


Fig. 5. State-machine states as FBD blocks with outgoing transitions dependent on: a) two variables, b) time; c) time plots, d) ST code translated from the first diagram

Main window for *Restart_FBD* project is shown in Fig. 6. Project tree (left part) involves now user-defined *Automat_blocks* library with STATE_C and STATE_T blocks. Tree of the FBD editor (middle) looks similarly as before (Fig. 2), although basic elements are now different and elements *with EN* are not needed. *Functions* branch is expanded here to show subbranches such as

arithmetic ADD, SUB, ..., logic AND, OR, ..., conversions INT_TO_REAL, ..., etc.

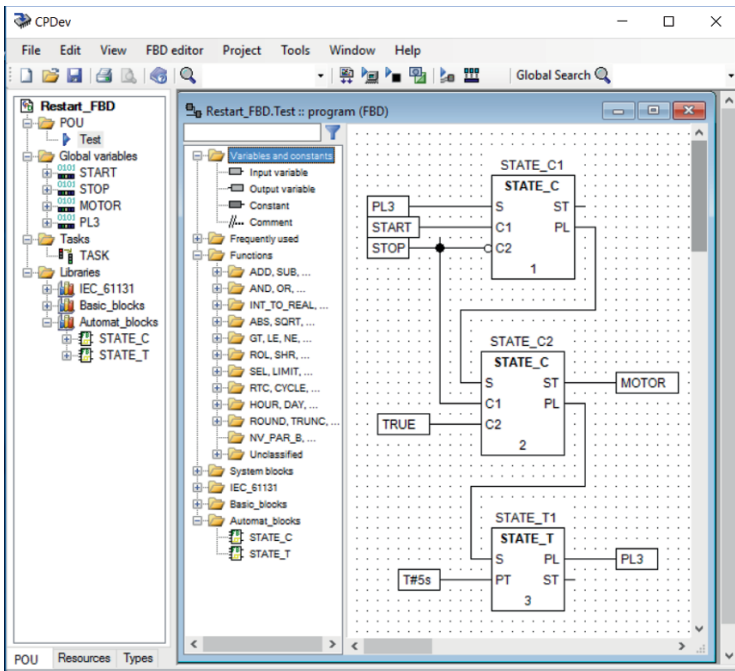


Fig. 6. CPDev window for *Restart_FBD* project

The FBD diagram on the right represents state-machine of Fig. 1b. START and (NOT) STOP inputs to the block STATE_C1 implement outgoing transition to STATE_C2 by means of PL output. Similarly for transitions from STATE_C2 and STATE_T1. The variable PL3 (global) connecting STATE_T1 with STATE_C1 must be initialized to TRUE. This makes the block STATE_C1 active (*ready*) at the beginning.

FBD diagrams are created similarly as before, i.e. by dragging elements from editor tree on the drawing board. Automatically assigned temporary names, as well as execution order, can be changed in properties windows of corresponding elements. Connections are created automatically by A* algorithm [15]. It suffices to point out one end, drag cursor to the other, and the path connecting both ends is calculated and drawn automatically. One has to admit however, that in case of tight diagrams or distant connections the path often does not correspond to the one intuitively expected. Note that the same mechanism is applied to make connections in the LD editor. In LD however the connections, as short, look as expected.

Simulation window, similar to the one before, can also be created. Basic panel for verification of execution is shown in Fig. 7. The ET element on the right denotes elapsed time of TON1 timer in STATE_T1 block. This indicates that nested local variables, not only global ones, can be displayed by CPSim.

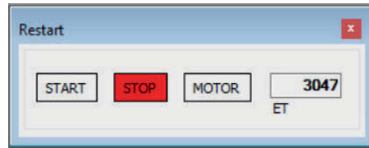


Fig. 7. Simulation of *Restart_FBD* project

FBD editors of CoDeSys (CNC), STEP 7, Control Builder F, and ISaGRAF are basically similar to that of CPDev. CoDeSys, STEP 7, and ISaGRAF provide automatic connections. Another CoDeSys tool, called FBD, strongly reminds LD, with successive sections instead of the rungs. So numbers indicating execution order are not needed. STEP 7 FBD diagram can be expanded only downwards. ISaGRAF allows to create logic portions of diagram using contacts. Except for automatic connections, CPDev seems fairly close to Control Builder F.

6 SFC diagram

As compared with Fig. 2 and Figs. 5,6, SFC diagram shown in Fig. 8 for the example is fairly compact. It should not be surprising though, since SFC language is particularly suitable to implement state-machines. The project tree on the left involves *SFC_blocks* library.

The steps Init, Step2 and Step3 on the diagram represent states 1, 2, 3 of the state-machine (Fig. 1b). By default, Init step is active at the beginning (*ready*). Expressions below Trans elements denote conditions to pass from one step to another. Condition Step3.T \geq T#5s replaces timer output TON1Q in Fig. 1b. Actions 1, 2, 3 bound to successive steps implement assignments written in rectangles on the state-machine diagram. They are defined using the list in the lower-right corner of the window. For instance, by clicking *edit* for Action2 a window with assignment MOTOR:=TRUE; appears.

Elements of SFC diagram are placed on the drawing board by choosing appropriate option of the editor or clicking an icon in the toolbar above the drawing board. First five icons of the toolbar insert:

- new step with following transition
- transition with following step
- sequence selection, i.e. new branch with a sequence of steps and transitions

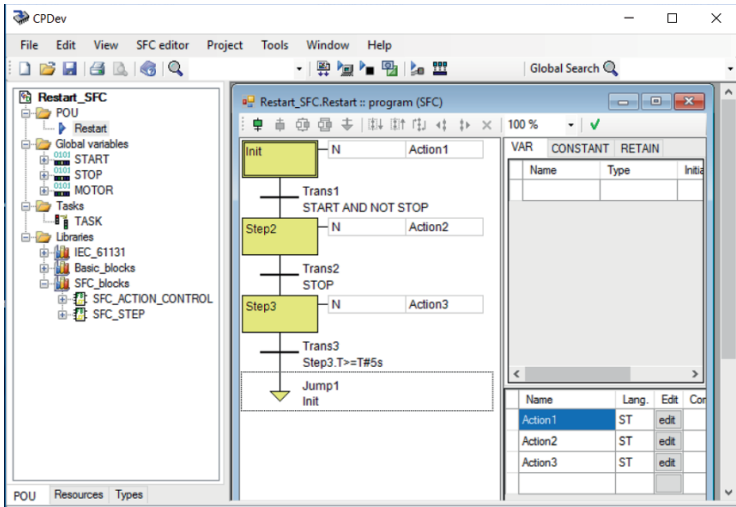


Fig. 8. CPDev window for *Restart_SFC* project

- simultaneous sequence, i.e. new branch with concurrent sequence of steps and transitions
- jump moving execution to another step.

Remaining icons in the toolbar provide editing functions, such as moving, exchanging, deleting, etc. More on CPDev SFC editor can be found in [16]. Some solutions have been adapted from CoDeSys [6].

To create the diagram of Fig. 8 the first icon suffices (*step+transition*). Step followed by sequence selection (not needed in the example) implements state with a few outgoing branches, whereas simultaneous sequence represents control of concurrent (parallel) processes with synchronization of final steps (states).

Main part of ST code (not shown) translated from SFC diagram is a set of IF Step instructions. Steps are implemented as instances of SFC_STEP function block from *SFC_blocks* library. Likewise, actions are handled by instances of STEP_ACTION_CONTROL block.

Tracing steps while testing an SFC diagram is much needed feature of engineering tool. CPSim provides this by creating panels with *Step.X* and *Step.T* system variables. Corresponding window for *Restart_SFC* project is shown in Fig. 9.

CoDeSys editor, seemingly most convenient, has been a pattern for CPDev. S7-GRAPH is a part of STEP 7 Professional package [8], whereas LAD and FBD belong to STEP 7 Basic [7]. Control Builder F creates SFC diagram by means of separate programs for steps, transitions, and actions. This may be appropriate for large projects, but is cumbersome for small ones. Similarly as CoDeSys (and CPDev), ISaGRAF applies an intelligent algorithm to indicate what elements can be added at current stage of diagram development.

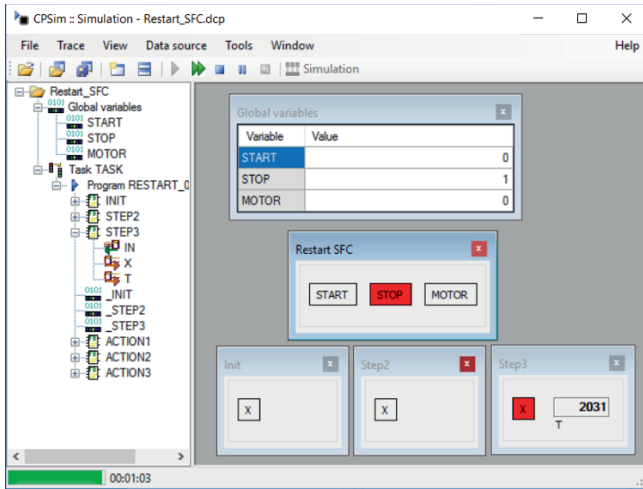


Fig. 9. Simulation of *Restart_SFC* project

7 Extensions of ST compiler

Usage of CPDev to program an industrial PC in DCS system have been planned since some time, with PC as a master over number of low-level PLCs. The PC would execute optimization, filtering of various signals, monitoring, diagnostics, etc. Runtime program will be implemented as Windows Service [17].

All this requires substantial extension of ST compiler which until recently accepted only single-dimensional arrays of simple variables. Current version of the extended compiler can handle:

- single-layer and hierarchical structures
- multidimensional arrays of variables, structures, and function blocks
- structures as inputs/outputs of function blocks.

Naturally, the set of VMASM instructions of virtual machine had to be extended accordingly, particularly by directives handling pointer type.

Figure 10 presents CPDev window with *StaSto_Tab_2L_Str project* in which some of new functionalities are employed. The project implements three Start-Stop RS flip-flop based circuits, each one turning immediately a MOTOR on and off, and a PUMP after some delays provided by TON, TOF timers.

Structures are declared in additional `_TYPES` program (middle), so at first single-layer `IN_signals`, `OUT_signals`, and then two-layer `IO_signals`. Array of structures is declared in global variables (left part) as `ARRAY[0..2] OF IO_signals`. Main program (right part) in `VAR` declarations involves arrays of function blocks `RS`, `TON`, `TOF` from *IEC_61131* library. `FOR` loop handles the three Start-Stop circuits. Elements of structures are inputs to function blocks.

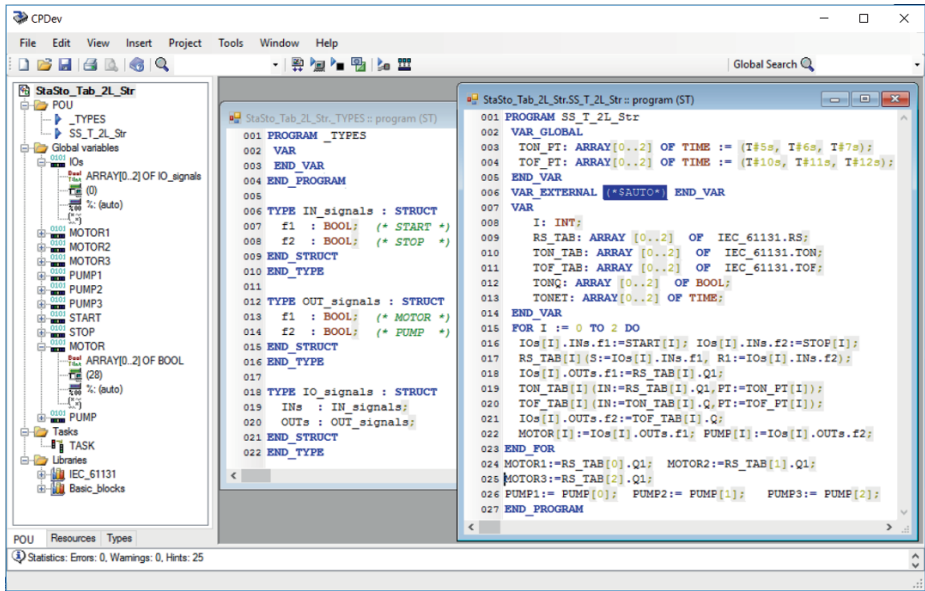


Fig. 10. Project with three Start-Stop circuits involving structures and arrays

All analyzed environments support advanced features of ST language (Siemens in S7-SCL from STEP 7 Professional) needed to implement large complicated programs, especially in DCS systems. It happens rarely, though, in case of manufacturers of small and medium size PLC/PAC controllers, where LD, FBD, and eventually SFC are sufficient (for now).

8 Conclusions

Overview of programming in IEC 61131-3 languages in CPDev environment has been presented. Remarks how CPDev looks like in relation to CoDeSys, STEP 7, Control Builder F, and ISaGRAF have also been given. Current version of CPDev provides all basic functionalities required by IEC 61131-3 standard. Modification of LD editor, so as to begin creating a rung from empty straight line, would remove the major difference from the editors. Since CPDev is not a commercial tool, it does not support some user-oriented features which speed up project development. So a few manufacturers who adopted CPDev use it to program themselves their controllers and systems. As being developed at a university, it is also a test platform for research in industrial informatics and real-time systems, not to mention teaching and student projects.

References

1. IEC 61131-3: Programmable controllers – part 3: Programming languages (2013)

2. Kasprzyk, J.: Programowanie sterowników przemysłowych. WNT, Warszawa (2006)
3. Rzońca, D., Sadolewski, J., Stec, A., Świder, Z., Trybus, B., Trybus, L.: Programming controllers in Structured Text language of IEC 61131-3 standard. *Journal of Applied Computer Science* **16**(1) (2008) 49–67
4. Hajduk, Z., Trybus, B., Sadolewski, J.: Architecture of FPGA Embedded Multi-processor Programmable Controller. *IEEE Transactions on Industrial Electronics* **62**(5) (2015) 2952–2961
5. Jamro, M., Rzońca, D., Sadolewski, J., Stec, A., Świder, Z., Trybus, B., Trybus, L.: CPDev Engineering Environment for Modeling, Implementation, Testing, and Visualization of Control Software. In Szewczyk, R., Zieliński, C., Kaliczyńska, M., eds.: *Recent Advances in Automation, Robotics and Measuring Techniques*. Springer International Publishing, Cham (2014) 81–90
6. CODESYS download area. <https://www.codesys.com/download.html>
7. SIMATIC STEP 7 Basic V14.0 System Manual. <https://support.industry.siemens.com/cs/document/109742266/step-7-basic-v14-0>
8. SIMATIC STEP 7 Professional V14 System Manual. <https://support.industry.siemens.com/cs/document/109742272/simatic-step-7-professional-v14-0>
9. Freelance Quickstart Tutorial. <http://new.abb.com/control-systems/essential-automation/freelance/additional-pages/freelance-quickstart-tutorial>
10. ISaGRAF Download Center. <http://www.isagraf.com/index.htm?> http://www.isagraf.com/pages/support/download_centre.htm
11. Trybus, L.: *Regulatory wielofunkcyjne*. WNT, Warszawa (1992)
12. Rzońca, D., Sadolewski, J., Stec, A., Świder, Z., Trybus, B., Trybus, L.: LD Graphic Editor Implemented in CPDev Engineering Environment. In Szewczyk, R., Zieliński, C., Kaliczyńska, M., eds.: *Automation 2017. ICA 2017. Advances in Intelligent Systems and Computing*, vol 550. Springer, Cham (2017) 178–85
13. CPDev. <http://www.cpdev.kia.prz.edu.pl>
14. Trybus, B.: Development and Implementation of IEC 61131-3 Virtual Machine. *Theoretical and Applied Informatics* **23**(1) (2011) 21–35
15. Jamro, M., Rzonca, D.: Automatic connections in IEC 61131-3 Function Block Diagrams. In: *2013 Federated Conference on Computer Science and Information Systems*. (Sept 2013) 463–469
16. Stec, A.: SFC Graphic Editor for CPDev Environment. In Szewczyk, R., Zieliński, C., Kaliczyńska, M., eds.: *Automation 2017. ICA 2017. Advances in Intelligent Systems and Computing*, vol 550. Springer, Cham (2017) 186–194
17. Introduction to Windows Service Applications. [https://msdn.microsoft.com/en-us/library/d56de412\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/d56de412(v=vs.110).aspx)

Automatic Code Generation of MIMO Model Predictive Control Algorithms using Transcompiler

Patryk Chaber, Maciej Ławryńczuk

Institute of Control and Computation Engineering, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland, tel. +48 22 234-71-24
pjchaber@gmail.com

Abstract. This paper describes a system for code auto-generation of Model Predictive Control algorithms for Multiple-Input, Multiple-Output processes. Transcompiler – the main part of the system – generates C code of the algorithm, basing on MATLAB code, which contains definition of both algorithms and its parameters. The resulting code is optimised for microcontrollers in terms of memory and computational power necessary for on-line calculation of the optimal values of manipulated variables. This approach may decrease the development time of prototype controllers and lower their cost. Tests are conducted using STM32 microcontroller for simulated processes and results are demonstrated and described.

Keywords: Model Predictive Control, microcontroller, transcompiler, code auto-generation.

1 Introduction

Advanced control algorithms, like Model-Predictive-Control (MPC), are commonly known to offer much better control quality, compared to the classical Proportional-Integral-Derivative (PID) algorithm [1]. Other advantages like ability to take into account dimensionality of a process (multiple-input, multiple-output), its nonlinearity and at last limitations of signals, allows MPC algorithms to unconditionally surpass the basic PID approach. A variety of MPC algorithms are currently in use in many fields, e.g. medicine [2], power inverting [3], driving systems [4] and robotics [5].

The technological advancement over last years caused a great reduction in prices and huge increase in terms of computational capabilities of microcontrollers and FPGA. This allows to implement complex control algorithms, in applications where the intervention time of a controller is of the order of milliseconds, maintaining its low production costs.

The simplicity of PID algorithm is one of the most important factor for its common appearance in the industry [6]. To ease the process of implementation of MPC algorithms the automatic generation tool is introduced. It is expected to

help replacing outdated and low quality controllers. This approach is already successfully applied outside simulated environment [7, 8]. What is more, automatically generated code can be optimised in such a way, only a minimum amount of resources and computational time are used [9, 10]. Even though this is often ignored, for a battery powered microcontroller it becomes a significant designing requirement.

2 Automatic Code Generation

There is no precise definition of automatic code generation. Generally speaking it is a process of creating a fully functional code, in some programming language, based on information provided by user. Sometimes this information is provided in form of program code in higher-level language, or using graphical user interface. Earlier approach is the most popular one. In the presented approach this is also the case – as an input language, the MATLAB and C mixture is chosen, and the output language is pure C.

Despite the fact that resulting code is focused on specific task and hardware platform, the code, based on which it has been generated, is easily reusable. Its modification is not as laborious compared to manually producing whole code, thus reducing manufacturing costs. Difference between automatic code generation and basic translation is, that additional optimising steps can be performed so that the generated code uses as little resources as possible. Some of those depend on lengthening the volume of the code in exchange for lower execution time (due to less jumps and branches). Other limits its usage to a single highly optimised algorithm, or optimises only a part of code which may be a bottleneck – mainly the optimisation procedure. Presented approach assumes that the automatic code generation should allow high flexibility of code writing, being able to generate task oriented code at the same time.

3 Model predictive Control

Model Predictive Control (MPC) algorithms calculates new control signal value based on an implemented model in such a way the cost function

$$J(k) = \sum_{p=1}^N \|\mathbf{y}^{\text{sp}}(k+p|k) - \hat{\mathbf{y}}(k+p|k)\|^2 + \lambda \sum_{p=0}^{N_u-1} \|\Delta \mathbf{u}(k+p|k)\|^2 \quad (1)$$

where

$$\hat{\mathbf{y}}(k+p|k) = \begin{bmatrix} \hat{y}_1(k+p|k) \\ \vdots \\ \hat{y}_{n_y}(k+p|k) \end{bmatrix}, \quad \mathbf{y}^{\text{sp}}(k+p|k) = \begin{bmatrix} y_1^{\text{sp}}(k+p|k) \\ \vdots \\ y_{n_y}^{\text{sp}}(k+p|k) \end{bmatrix} \quad (2)$$

$$\Delta \mathbf{u}(k+p|k) = \begin{bmatrix} \Delta u_1(k+p|k) \\ \vdots \\ \Delta u_{n_u}(k+p|k) \end{bmatrix} \quad (3)$$

is minimised. First part determines a squared error between the predicted trajectory of the output signal $\hat{\mathbf{y}}(k+p|k)$ and the setpoint trajectory $\mathbf{y}^{\text{sp}}(k+p|k)$ over the prediction horizon N , i.e. $p = 1, \dots, N$. Second component acts as a penalty for high increases in control signal $\Delta \mathbf{u}(k+p|k)$ over next N_u (control horizon) discrete time instants. For $p \geq N_u$, the control signal value is assumed to be constant $\Delta \mathbf{u}(k+p|k) = 0$. The parameter λ allows to change an influence of those components on the final value of cost. Notation $x(k+p|k)$ should be understood as "value of x denoted in a time instant k for a time instant $k+p$ ".

In this paper analytic Generalised Predictive Control (GPC) algorithm is used as an example of MPC algorithm. The model in form of differential equations is used

$$y_m(k) = - \sum_{i=1}^{n_A} a_i^m y_m(k-i) + \sum_{j=1}^{n_u} \sum_{i=0}^{n_B} b_i^{m,j} u_j(k-1-i), \quad m = 1, \dots, n_y \quad (4)$$

where $y_m(k)$ is m -th output signal. Values a_i^m and $b_i^{m,j}$, $m = 1, \dots, n_y$, $n = 1, \dots, n_u$, $i = 1, \dots, n_A$, $l = 1, \dots, n_B$, are parameters of the model used. Based on those equation elements of step response

$$s_k^{m,j} = - \sum_{i=1}^{\min\{k-1, n_A\}} a_i^m s_{k-i}^{m,j} + \sum_{i=0}^{\min\{k-1, n_B\}} b_i^{m,j} \quad (5)$$

matrix M can be evaluated as follows

$$M = \begin{bmatrix} S_1 & 0 & \dots & 0 \\ S_2 & S_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ S_{N-1} & S_{N-2} & \dots & S_{N-N_u} \\ S_N & S_{N-1} & \dots & S_{N-N_u+1} \end{bmatrix}, \quad S_k = \begin{bmatrix} s_k^{1,1} & s_k^{1,2} & \dots & s_k^{1,n_u} \\ s_k^{2,1} & s_k^{2,2} & \dots & s_k^{2,n_u} \\ \vdots & \vdots & \ddots & \vdots \\ s_k^{n_y,1} & s_k^{n_y,1} & \dots & s_k^{n_y,n_u} \end{bmatrix} \quad (6)$$

Defining vector $\mathbf{y}^0(k+p|k) = [y_1^0(k+p|k), \dots, y_{n_u}^0(k+p|k)]^T$ and

$$\hat{\mathbf{Y}}(k) = \begin{bmatrix} \hat{\mathbf{y}}(k+1|k) \\ \vdots \\ \hat{\mathbf{y}}(k+N|k) \end{bmatrix}, \mathbf{Y}^0(k) = \begin{bmatrix} \mathbf{y}^0(k+1|k) \\ \vdots \\ \mathbf{y}^0(k+N|k) \end{bmatrix}, \Delta \mathbf{U}(k) = \begin{bmatrix} \Delta \mathbf{u}(k+0|k) \\ \vdots \\ \Delta \mathbf{u}(k+N_u-1|k) \end{bmatrix} \quad (7)$$

Values $y_m^0(k+p|k)$, $p = 1, \dots, N$, are defined as follows

$$y_m^0(k+p|k) = d_m(k) - \sum_{i=1}^{n_1} a_i^m y_m^0(k+p-i|k) - \sum_{i=n_1+1}^{n_A} a_i^m y_m(k+p-i) + \sum_{j=1}^{n_u} \left[\sum_{i=0}^{n_2} b_i^{m,j} u_j(k-1) + \sum_{i=n_2+1}^{n_B} b_i^{m,j} u_j(k-1+p-i) \right] \quad (8)$$

where $n_1 = \min\{n_A, p - 1\}$, $n_2 = \min\{n_B, p\}$, and $d_m(k + 1|k) = \dots = d_m(k + N|k) = d_m(k)$ is a disturbance of "DMC" type calculated as

$$d_m(k) = y_m(k) + \left(- \sum_{i=1}^{n_A} a_i^m y_m(k - i) + \sum_{j=1}^{n_u} \sum_{i=0}^{n_B} b_i^{m,j} u_j(k - 1 - i) \right) \quad (9)$$

where $y_m(k)$ is measured output signal, and the part in parentheses is $y_m(k)$ calculated from model (4).

Finally prediction formula can be described as

$$\hat{Y}(k) = M\Delta U(k) + Y^0(k) \quad (10)$$

Without taking constraints into account, optimal values of control signals increments are calculated as

$$\Delta U(k) = (M^T M + \lambda \mathbf{I})^{-1} M^T \cdot (Y^{\text{sp}}(k) - Y^0(k)) \quad (11)$$

where $Y^{\text{sp}}(k) = [\mathbf{y}^{\text{sp}}(k)^T, \dots, \mathbf{y}^{\text{sp}}(k)^T]^T$ is a vector with length of N .

At the end of each iteration of GPC algorithm only the first element of vector $\Delta U(k)$, i.e. $\Delta u(k|k)$ is applied to the control process, other values are discarded. In the next iteration, value of time instant k is incremented and whole procedure is repeated.

4 Development Tools

A set of tool for automatic code generation was created. Their purpose is to generate efficient code, i.e. code which does not waste computational power nor memory space. Having in mind that the generation and compilation process are usually performed on powerful computers, it is worth to calculate as much as possible on those machines, leaving only the input-dependant values to be updated in the microcontroller program. It is worth noting, that these simplification of the generated code might exceed the capabilities of the compiler's optimisation procedure.

Data continuity is an another significant aspect that is assumed to have to be assured. Presented approach is designed to be able to change control algorithms seamlessly during the program execution. This will be useful for coping with failures of the controlled process, where having lost one of the actuators, the change in the algorithm used is required.

Proposed approach allows to generate code in C language (which is usually the best for microcontrollers) based on the mixture of C code (which defines target platform) and MATLAB script (which defines the algorithms and their parameters). The transition between the input mixture of languages into pure C code is performed using Transcompiler. The resulting code has a structure, designed to ease the process of controller implementation. Moreover this allows to divide this process to separate specialists in fields of microcontrollers, MATLAB programmer, MPC algorithms. Implementation of profiler and simulation

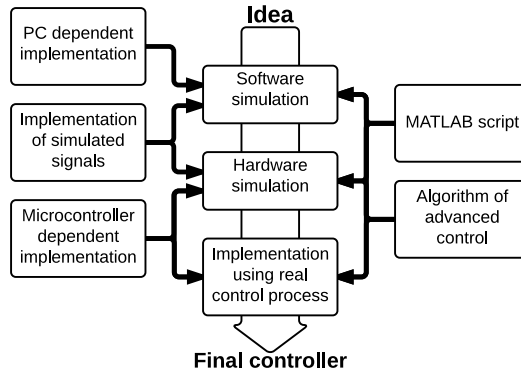


Fig. 1. Proposed design stages for controller implementation

allows to carry the implementation process in stages. First stage allows to test the controller on PC, using simulated signals in place of real ones. Next the microcontroller is used as the target platform, but still the signals are simulated, i.e. the process of control is still not used. Lastly, the control process is connected and used. Throughout the whole process of implementation, the same MATLAB code is used for algorithms' definition, i.e. the algorithmic part of the implementation is completely separated from the target platform dependent code. Proposed designing process is depicted in the Fig. 1.

4.1 Data structures

The main assumption of the proposed approach is to keep the data concerning the process separated from the algorithms implementation. This allows to change algorithms during the program execution without having to wait for the algorithms' variables to "keep up" with the changes. An archiving data structure `ArchiveData` is therefore introduced. Each measurement of the output variable of process and each applied control signal value is stored in this structure, as well as setpoint values. Its dimensionality is strictly connected to the process and is assumed to be constant in time.

Another structure that is used in the framework is a `CurrentControl`, which contains newly calculated control signal values for each algorithm implementation. While the `ArchiveData` is the input of the algorithm, the `CurrentControl` can be treated as the output. The reasoning for this separation is that many algorithms might be tested during one iteration of the controller loop. Therefore separate `CurrentControl` structures allows to choose the best of the control signal values, which will finally be copied to the `ArchiveData` structure.

4.2 Framework

The code structure is divided into three parts: hardware definition, algorithms definition, controller logic. The first part (written in C language) is used to

initialise functions and modules specific for particular microcontroller, e.g. communication modules, Floating Point Unit, digital-analogue and analogue-digital converters. The second one (written in MATLAB) contains definition of parameters of algorithms used and names of the generated functions for further use. The library of algorithms that can be generated is expandable, thus the usage of this approach is not limited to any specific group of control algorithms, as far as they can be implemented in expected form. Last part is written in C, and determines how selected control algorithms will be used, what happens if the time of single iteration of the algorithms is exceeded, etc.

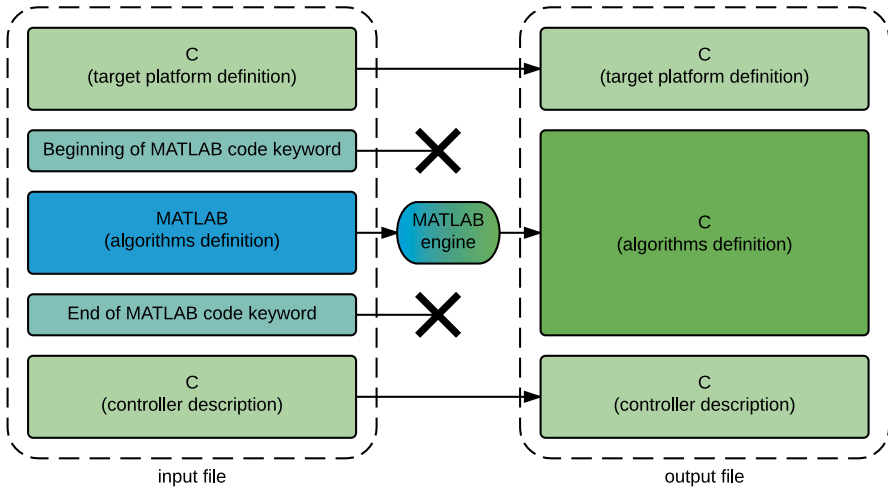


Fig. 2. Transcompiler's general principle of operation

4.3 Transcompiler

The task of the Transcompiler is to generate pure C code from a mixture of MATLAB and C code, which describes control algorithms used and target platform. This process is quite simple, and is briefly depicted in Fig. 2 – the C code of the input file is put into the output file unchanged, while the MATLAB part is executed using the MATLAB engine, and the output (which is now a C language) is put in place of the MATLAB code. This approach assumes that the execution of the MATLAB code generates only functions in form of `void GPC(ArchiveData *ad, CurrentControl *cc);`, where the GPC is an example name of function which implements GPC algorithm. The MATLAB part is separated from the C part by the delimiters `#MPC_BEGIN` and `#MPC_END`. Example of such a code is

```
#MPC_BEGIN
a=[...]; b=[...]; % values given in paper
```

```

N=10; Nu=5; lambda1=1.0; lambda2=0.1;
generate(GPC(a,b,N,Nu,lambda1,[],[],[],[]),'GPC10');
generate(GPC(a,b,N,Nu,lambda2,[],[],[],[]),'GPC01');
#MPC_END

```

which will generate following code

```

void GPC10(ArchiveData *ad, CurrentControl *cc)
{ /* implementation for lambda = 1.0 */}
void GPC01(ArchiveData *ad, CurrentControl *cc)
{ /* implementation for lambda = 0.1 */}

```

5 Application Example

5.1 Hardware

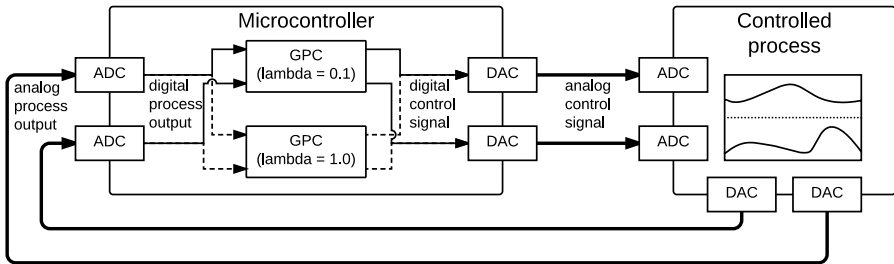


Fig. 3. Connection scheme of the test stand

Both simulated process and controller use two separate STM32F746G-DISCO boards to implement them. This development board contains STM32F746VG microcontroller, which includes Floating Point Unit (i.e. hardware for calculations in floating point arithmetic). It also incorporates high-speed embedded memories with a Flash memory of 1 Mbyte, 320 Kbytes of SRAM and extensive range of I/Os and peripherals. Despite having two digital-analogue converters (DAC), the microcontroller has them permanently connected to another resources. Therefore additional board with two 12-bit DAC is used instead. Amongst many communication interfaces two of them are mainly used – Universal Asynchronous Receiver and Transmitter (which allows, using the programmer as an UART-USB converter, to communicate with PC), and I²C for communication with DAC. Both simulated process and controller use the same hardware configuration. Connections between those two boards are shown on Fig. 3.

5.2 Software – control process

The control process used in this example has two manipulated (U , $n_u = 2$) and two controlled (Y , $n_y = 2$) variables. Equation describing this process is as

follows:

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{1}{0.7s+1} & \frac{5}{0.3s+1} \\ \frac{1}{0.5s+1} & \frac{2}{0.4s+1} \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} \quad (12)$$

A discrete-time representation of this dynamic system is denoted using sampling time of $T_s^{\text{sim}} = 0.003\text{s}$ and Zero Order Hold method. Resulting differential equations are implemented as simulated control process:

$$\begin{aligned} y_1(k^{\text{sim}}) &= 0.004276544 \cdot u_1(k^{\text{sim}} - 1) - 0.004233991 \cdot u_1(k^{\text{sim}} - 2) \\ &\quad + 0.049750831 \cdot u_2(k^{\text{sim}} - 1) - 0.049538070 \cdot u_2(k^{\text{sim}} - 2) \\ &\quad - 1.985773290 \cdot y_1(k^{\text{sim}} - 1) + 0.985815842 \cdot y_1(k^{\text{sim}} - 2) \\ y_2(k^{\text{sim}}) &= 0.005982036 \cdot u_1(k^{\text{sim}} - 1) - 0.005937339 \cdot u_1(k^{\text{sim}} - 2) \\ &\quad + 0.014943890 \cdot u_2(k^{\text{sim}} - 1) - 0.014854495 \cdot u_2(k^{\text{sim}} - 2) \\ &\quad - 1.986546019 \cdot y_2(k^{\text{sim}} - 1) + 0.986590716 \cdot y_2(k^{\text{sim}} - 2) \end{aligned} \quad (13)$$

and analogously the model used for GPC algorithms is determined with sampling time of $T_s = 0.05\text{s}$ and Zero Order Hold method, and looks as follows

$$\begin{aligned} y_1(k) &= b_0^{1,1} u_1(k-1) - b_1^{1,1} u_1(k-2) + b_0^{1,2} u_2(k-1) - b_1^{1,2} u_2(k-2) \\ &\quad - a_1^1 y_1(k-1) + a_2^1 y_1(k-2) \\ y_2(k) &= b_0^{2,1} u_1(k-1) - b_1^{2,1} u_1(k-2) + b_0^{2,2} u_2(k-1) - b_1^{2,2} u_2(k-2) \\ &\quad - a_1^2 y_2(k-1) + a_2^2 y_2(k-2) \end{aligned} \quad (14)$$

$$\begin{aligned} a_1^1 &= -1.77754, & a_2^1 &= +0.78813, & a_1^2 &= -1.78733, & a_2^2 &= +0.79852, \\ b_0^{1,1} &= +0.06894, & b_1^{1,1} &= -0.05835, & b_0^{1,2} &= +0.76759, & b_1^{1,2} &= -0.71468, \\ b_0^{2,1} &= +0.09516, & b_1^{2,1} &= -0.08398, & b_0^{2,2} &= +0.23501, & b_1^{2,2} &= -0.21264 \end{aligned} \quad (15)$$

$y_m(k-p)$ ($m = 1, 2, p = 1, 2$) denotes the m -th output signal of the process at the discrete time instant $k-p$, and $u_n(k-p)$, $n = 1, 2, p = 1, 2$ stands for the n -th input signal at the discrete time instant $k-p$. It is clearly visible that the model used for GPC algorithm is assumed to be known and to be accurate. There are different symbols used for time instants in (13) and (14) to underline that the simulated process updates its signal values with different frequency compared to the controller.

It is worth mentioning, that in both cases there are no notable delays between the calculation of new output signal value (in case of the control process) or control signal value (for controller) and corresponding changes of their analogue representations. What is more, calculation time (of both signals) is also negligible – it is less than 1ms, therefore as soon as the results are ready they are applied to DAC modules and converted to analogue signals.

5.3 Software – control algorithm

GPC algorithms used in this example differ only in model parameters and a value of λ , which denotes a penalty for high increases of control signals' values.

This is just to visualise the easiness of implementation of different algorithms or algorithms with different parameters. Other parameters of those algorithms are the same in both implementations: prediction horizon $N = 10$, control horizon $N_u = 5$. There are implemented constraints for control signals' values, i.e. minimum value is -2000 and maximum 2000 for both signals. Constraints are applied after the control signals are calculated, by trimming them. Setpoint values changed each $2.5s$. The algorithm implementation used for current control signals' calculation was changed in predefined moments, i.e. at: $2.5s$, $7.5s$, $12.5s$, $16s$, $17s$.

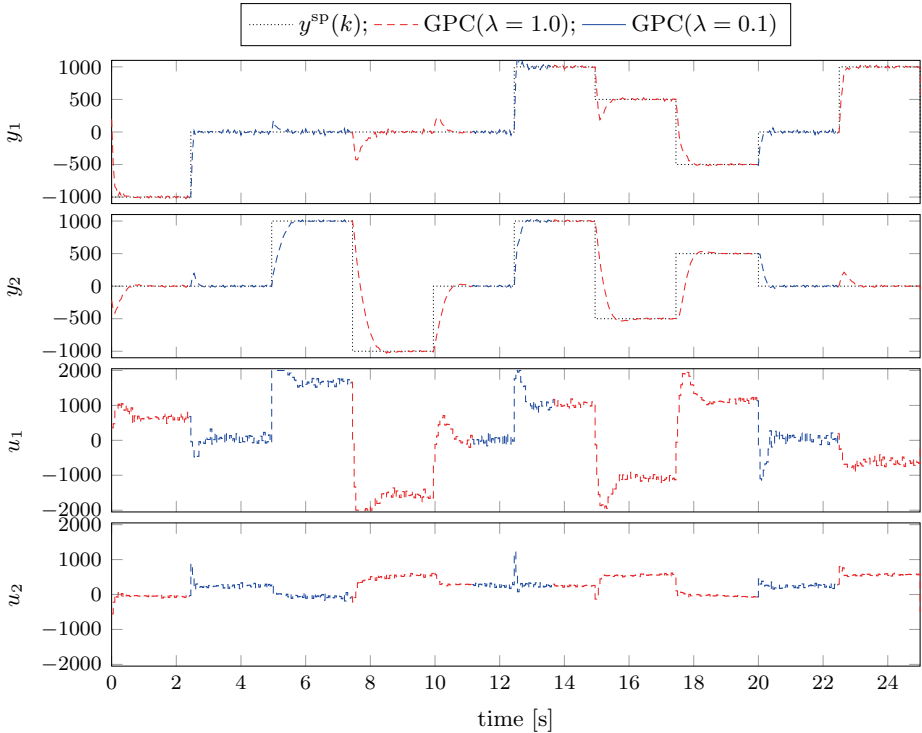


Fig. 4. Output (y_1, y_2) signal values and control (u_1, u_2) signal values acquired during experiment runtime.

6 Experimental Results

The results of experiment visible in Fig. 4 show that the change of algorithms used is indeed seamless, irrespective of the state of the process. This can be of great importance in terms of fault tolerant algorithms.

Apparent noise results from the imperfection of measurements. Even though the control process is simulated, the usage of not software but hardware simulation induces disturbances, similar to the ones seen in real life applications.

7 Conclusions

Presented approach allows to divide the process of controller's implementation between specialists in their fields: embed programming and microcontrollers, control algorithms and MATLAB. Consecutive stages of implementation allows to create reliable and efficient code which can be easily run on microcontrollers. Seamless change of algorithms can be utilised for fault tolerant algorithms, where instant change of model is necessary.

In contrary to general automatic generation of the C code basing on MATLAB code (i.e. translation) it is expected that presented approach is faster, and uses less resources of the microcontroller. Preliminary results are gathered but more thorough research is required. Therefore further work includes mainly comparison of the proposed system with the most popular tools like MATLAB Coder.

References

1. F. Salem and M. I. Mosaad, "A comparison between MPC and optimal PID controllers: Case studies," in *Michael Faraday IET International Summit 2015*, pp. 59–65, 2015.
2. G. C. Goodwin *et al.*, "Application of MPC incorporating Stochastic Programming to Type 1 diabetes treatment," in *2016 American Control Conference (ACC)*, pp. 907–912, 2016.
3. X. Jiang *et al.*, "Application based on fast online MPC in power inverter system," in *Proceedings of the 33rd Chinese Control Conference*, pp. 7673–7678, 2014.
4. P. J. Serkies and K. Szabat, "Application of the MPC to the Position Control of the Two-Mass Drive System," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 9, pp. 3679–3688, 2013.
5. Y. Noda, T. Sumioka, and M. Yamakita, "An application of fast MPC for bike robot," in *2012 Proceedings of SICE Annual Conference (SICE)*, pp. 540–545, 2012.
6. R. Vilanova, V. M. Alfaro, and O. Arrieta, *Robustness in PID Control*, pp. 113–145. London: Springer London, 2012.
7. M. Vukov *et al.*, "Experimental validation of nonlinear MPC on an overhead crane using automatic code generation," in *2012 American Control Conference (ACC)*, pp. 6264–6269, 2012.
8. P. Chaber and M. Ławryńczuk, "Effectiveness of PID and DMC control algorithms automatic code generation for microcontrollers: Application to a thermal process," in *2016 3rd Conference on Control and Fault-Tolerant Systems (SysTol)*, pp. 618–623, 2016.
9. G. Takács *et al.*, "Efficiency and performance of embedded model predictive control for active vibration attenuation," in *2016 European Control Conference (ECC)*, pp. 1334–1340, 2016.
10. P. Chaber and M. Ławryńczuk, "Auto-generation of advanced control algorithms' code for microcontrollers using transcompiler," in *Methods and Models in Automation and Robotics (MMAR), 2016 21st International Conference on*, pp. 454–459, 2016.

Implementation of Dynamic Matrix Control Algorithm Using Field Programmable Gate Array: Preliminary Results

Andrzej Wojtulewicz

Institute of Control and Computation Engineering, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland, andrwoj@gmail.com

Abstract. This paper describes implementation of the Dynamic Matrix Control (DMC) algorithm performed on an Altera Field Programmable Gate Array (FPGA) with the Cyclone IV chip. The DMC algorithm is implemented in its analytical (explicit) version which requires computationally simple matrix and vector operations in real time, no on-line optimisation is necessary. The test-bench application is prepared for fast comparison between C and HDL versions of code. A large number of independent logic cells can provide multi-parallel operations to achieve very fast operations. As a result, the algorithm may be used for controlling very fast dynamic processes characterised by sampling periods of millisecond order. Preliminary results of real experiments are demonstrated. The discussed control structure provides possibility to fast change of algorithm.

Key words: Dynamic Matrix Control, Model Predictive Control, Field Programmable Gate Array.

1 Introduction

Model Predictive Control (MPC) algorithms are interesting alternatives to classical controllers since they are able to give excellent control quality, in particular for constrained and multiple-input multiple-output processes with delays [8, 10]. They have been used in large-scale industrial applications (e.g. in refineries, chemical engineering, paper industry, food industry) for some 40 years [9]. In such cases the necessary sampling periods are quite long, of the order of seconds or even minutes. Hence, the MPC algorithms are implemented using the classical industrial hardware, i.e. industrial computers and in some simple cases even Programmable Logic Controllers (PLC). Currently, the MPC algorithms are used for controlling faster processes, characterised by shorter sampling periods. In such applications the algorithms may be effectively implemented using microcontrollers [2, 7].

This paper presents a universal implementation of the Dynamic Matrix Control (DMC) MPC algorithm using a Field Programmable Gate Array (FPGA). The FPGAs are interesting alternatives to classical microprocessors and microcontrollers because of a few reasons. First of all, the FPGA is able of fast

calculations. Bearing in mind that for fast applications the time necessary to complete calculations performed at one sampling instant must be as short as possible, it is interesting to notice that the FPGA with a relatively low speed clock, e.g. 25MHz, is able to perform calculation with the speed comparable with that of 3GHz desktop computers. This can be achieved thanks to parallel operations. The literature which illustrates this issue is quite rich, e.g. in [6, 5], some solutions based on Handel-C language (generated from a description written in MATLAB) are detailed. Due to low frequency, it is not necessary to use fan cooler which is an important feature in many embedded systems. Secondly, the structure of the FPGA may be flexibly tailored to the specific applications whereas general purpose microprocessors (and microcontrollers) have fixed architecture. Thanks to the structure of the FPGA its programming language, it is possible to use parallel operations. An another example of fast calculation method is to use a dedicated matrix multiply accelerator [4].

2 Dynamic Matrix Control Algorithm

Let u denotes the input of the process (the manipulated variable) and y the output (the controlled variable). Unlike the classical control methods, e.g. the Proportional-Integral-Derivative, the objective of the DMC algorithm [1, 3, 10] is to calculate on-line in real-time not only the value of the manipulated variable for the current sampling instant k , $k = 0, 1, 2, \dots$, but a future control policy of length N_u , which is named a control horizon. Usually, the future increments are calculated

$$\Delta \mathbf{u}(k) = [\Delta u(k|k) \ \dots \ \Delta u(k + N_u - 1|k)] \quad (1)$$

The increments are defined as $\Delta u(k|k) = u(k|k) - u(k - 1)$, $\Delta u(k + p|k) = u(k + p|k) - u(k + p - 1|k)$ for $p = 1, \dots, N_u - 1$. It is assumed that $\Delta u(k + p|k) = 0$ for $p \geq N_u$. The future increments of the manipulated variable are calculated in such a way that the differences between the set-point trajectory $y^{\text{sp}}(k + p|k)$ and the predicted output values $\hat{y}(k + p|k)$ are minimised over the prediction horizon $N \geq N_u$, i.e. for $p = 1, \dots, N$. Hence, for optimisation of the future control policy (1) the following quadratic cost function is usually used

$$J(k) = \sum_{p=1}^N (y^{\text{sp}}(k + p|k) - \hat{y}(k + p|k))^2 + \lambda \sum_{p=0}^{N_u-1} (\Delta u(k + p|k))^2 \quad (2)$$

where $\lambda > 0$ is a weighting coefficient. Although the whole vector $\Delta \mathbf{u}(k)$ is determined at each sampling instant, only its first element is actually applied to the process, i.e. $u(k) = \Delta u(k|k) + u(k - 1)$. At the next sampling instant, $k + 1$, the measurement of the process output is updated, the prediction is shifted one step forward and the whole procedure is repeated.

Predictions $\hat{y}(k + p|k)$ of the process output variable over the entire prediction horizon, i.e. for $p = 1, \dots, N$, are calculated using a dynamic model of the controlled process. It is a unique feature of the DMC algorithm that the process

is modelled by a series of discrete step response coefficients. The step response model shows reaction of the process output to a step excitation input signal, usually a unit step. The output of the classical linear step response model is [10]

$$y(k) = y(0) + \sum_{j=1}^k s_j \Delta u(k-j) \quad (3)$$

Real numbers s_1, s_2, s_3, \dots are step response coefficients of the model. Assuming that the process is stable, after a step change in the input the output stabilises at a certain value s_∞ , $\lim_{k \rightarrow \infty} s_k = s_\infty$. Hence, the model needs only a finite number of step response coefficients: $s_1, s_2, s_3, \dots, s_D$, where D is named a horizon of the process dynamics. The general prediction equation is

$$\hat{y}(k+p|k) = y(k+p|k) + d(k) \quad (4)$$

for $p = 1, \dots, N$. The quantities $y(k+p|k)$ are calculated from the step response model. In the ‘‘DMC type’’ disturbance model the unmeasured disturbance $d(k)$ is assumed to be constant over the prediction horizon. It is estimated from

$$d(k) = y(k) - y(k|k-1) \quad (5)$$

where $y(k)$ is a real (measured) value while $y(k|k-1)$ is calculated from the model. The prediction of the process output variable, i.e. the vector $\hat{\mathbf{y}}(k) = [\hat{y}(k+1|k) \dots \hat{y}(k+N|k)]^T$, may be compactly expressed in the following form [1, 3, 10]

$$\hat{\mathbf{y}}(k) = \mathbf{G} \Delta \mathbf{u}(k) + \mathbf{y}(k) + \mathbf{G}^p \Delta \mathbf{u}^p(k) \quad (6)$$

where $\mathbf{y}(k) = [y(k) \dots y(k)]^T$ is a vector of length N and $y(k)$ is the measured value of the process output signal for the current sampling instant, $\Delta \mathbf{u}^p(k) = [\Delta u(k-1) \dots \Delta u(k-(D-1))]^T$ is a vector of length $D-1$ comprised of increments of the process input signal for the past sampling instants, the matrices

$$\mathbf{G} = \begin{bmatrix} s_1 & 0 & \dots & 0 \\ s_2 & s_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ s_N & s_{N-1} & \dots & s_{N-N_u+1} \end{bmatrix} \quad (7)$$

and

$$\mathbf{G}^p = \begin{bmatrix} s_2 - s_1 & s_3 - s_2 & \dots & s_D - s_{D-1} \\ s_3 - s_1 & s_4 - s_2 & \dots & s_{D+1} - s_{D-1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{N+1} - s_1 & s_{N+2} - s_2 & \dots & s_{N+D-1} - s_{D-1} \end{bmatrix} \quad (8)$$

are of dimensionality $N \times N_u$ and $N \times (D-1)$, respectively. For $p > D$, $s_p = s_D$.

Thanks to using the prediction equation (6), the DMC cost function (2) becomes

$$\begin{aligned} J(k) &= \|\mathbf{y}^{\text{sp}}(k) - \hat{\mathbf{y}}(k)\|^2 + \|\Delta \mathbf{u}(k)\|_{\mathbf{A}}^2 \\ &= \|\mathbf{y}^{\text{sp}}(k) - \mathbf{G} \Delta \mathbf{u}(k) - \mathbf{y}(k) - \mathbf{G}^p \Delta \mathbf{u}^p(k)\|^2 + \|\Delta \mathbf{u}(k)\|_{\mathbf{A}}^2 \end{aligned} \quad (9)$$

where the set-point trajectory

$$\mathbf{y}^{\text{SP}}(k) = \begin{bmatrix} y^{\text{SP}}(k+1|k) \\ \vdots \\ y^{\text{SP}}(k+N|k) \end{bmatrix} \quad (10)$$

is a vector of length N and the weighting matrix $\mathbf{A} = \text{diag}(\lambda, \dots, \lambda)$ is of dimensionality $N_u \times N_u$. Having equated the vector of derivatives of the minimised cost-function (9) to zeros, the optimal future control moves are

$$\Delta \mathbf{u}(k) = \mathbf{K}(\mathbf{y}^{\text{SP}}(k) - \mathbf{y}(k) - \mathbf{G}^{\text{P}} \Delta \mathbf{u}^{\text{P}}(k)) \quad (11)$$

where $\mathbf{K} = (\mathbf{G}^{\text{T}} \mathbf{G} + \mathbf{A})^{-1} \mathbf{G}^{\text{T}}$ is a matrix of dimensionality $N_u \times N$, it is calculated once, off-line. Because at the current sampling instant k only the first element of the vector $\Delta \mathbf{u}(k)$ is used, the control law is

$$\Delta u(k|k) = \mathbf{K}_{n_u}(\mathbf{y}^{\text{SP}}(k) - \mathbf{y}(k) - \mathbf{G}^{\text{P}} \Delta \mathbf{u}^{\text{P}}(k)) \quad (12)$$

where the matrix \mathbf{K}_1 is the first row of the matrix \mathbf{K} .

3 Implementation of DMC Algorithm Using FPGA

It is necessary to point out some features of the FPGA which are important for implementation of the DMC algorithm. Firstly, there are FPGAs that consist of a hardware combination of logic cells matrix and ARM CPUs in one chip. Secondly, the FPGA has a large amount of basic logic cells which can be used to create large systems. The main advantage is that it is possible to use it in a parallel and totally independent way. When an efficient Digital Signal Processor (DSP) is compared to the FPGA, one may consider a theoretical example of a FIR filter with 200 degrees. If the DSP with 8 Multiply and Accumulate units are used, one has to wait 25 clock cycles. By using the FPGA one can program 200 MAC units and it is necessary to wait only one clock cycle for the first stage calculation and one clock cycle to sum the partial results.

The FPGA offers three levels of programming techniques. The first one is a HLS (High Level Synthesis), where the user can provide solution using C/C++ language – Spectra-Q for Altera or Vivado for Xilinx. The second one is a functional block description, where the user has to define behaviour of a unit. The programming language is similar to a simple script, in this case written in HDL (Hardware Description Language). After compilation and synthesis the system gives a solution using logic cells. That kind of unit can be called “black box” because one does not really care how compiler realises the functions. The third solution is very simple for the programmer as it is only necessary to connect basic digital blocks (like AND, OR, etc.) using graphical form or a HDL. This method can be viewed as an assembler for microcontroller and the fastest operations can be achieved.

Using these three methods of programming there is a huge potential to implement advanced computational algorithms on FPGAs. Moreover, there is also

a possibility to include software CPU called NIOS II in Altera FPGA system. NIOS II is RISC microcontroller (Reduced Instruction Set Computing). In one chip it is possible to include several software CPUs. The easiest way to do this is to develop some part of the system (less complex calculations) performed by software CPU in common C language. Next, a dedicated internal interface is used to connect some function blocks developed in hardware language to achieve the whole system. Function blocks can be considered as hardware accelerators. This way of thinking has been applied in this study. The main goal of presented work is to prepare a universal system for fast prototyping new control algorithms. An interesting feature of the described approach is the possibility to run at first the algorithm on the software side (soft CPU in FPGA, code written in C). Next, it may be run using the dedicated FPGA.

Fig. 1 shows the whole system implemented in FPGA. The main part is based on the NIOS II soft processor. There is also SDRAM interface to store temporary data and some calculations. This memory is connected to NIOS II through the Avalon bus. This bus, dedicated for NIOS II, is also used to connect custom block such as input-output unit and the main DMC algorithm blocks. The soft processor is used for time synchronisation, i.e. a hardware timer determines the moments of data sampling and setting new control value. Dedicated hardware blocks for DMC calculations are connected to the Avalon bus through universal input output interface. There are four steps: preparation of the measured signal, the free output response calculation, the control law calculation, conditioning of the control signal u .

In order to guarantee communication with the controlled process (a laboratory stand) a predefined protocol is used. The most basic operation needed is to read a value of the process controlled variable and write a new value of the process manipulated variable. The implemented protocol makes it possible to communicate with single-input single-output processes and with multiple-input multiple-output ones. The last task for software CPU is to send the actual data to MATLAB GUI and read a new set-point value. Communication with the controlled process is based on RS232 interface using standard text messages.

The DMC algorithm needs a step-response model of the process. It is obtained experimentally. Next, the matrices \mathbf{K}_1 and \mathbf{G}^p are calculated off-line. The last issue is to copy all values of the algorithm's parameters to the Very High Speed Integrated Circuits Hardware Description Language (VHDL) file. The matrices are hardcoded directly in the FPGA memory. The function block called MPDATA has two inputs. for selecting row and column of \mathbf{G}^p matrix. The output value was set on output $_v$. Similar solution is used for the \mathbf{K}_1 data.

Fig. 2 shows all blocks. The first one called Y_CONDITION is directly connected to the output y from the NIOS II processor. It can perform some calculations on the current measured value to meet conditions of the algorithm. The user can put some filter or linearisation on measurement values. It has a trigger signal which is also connected to NIOS II and it is used to start calculation of one algorithm iteration. The result is on the y_c output and there is the ready_f signal connected to the next block to indicate the calculations are completed.

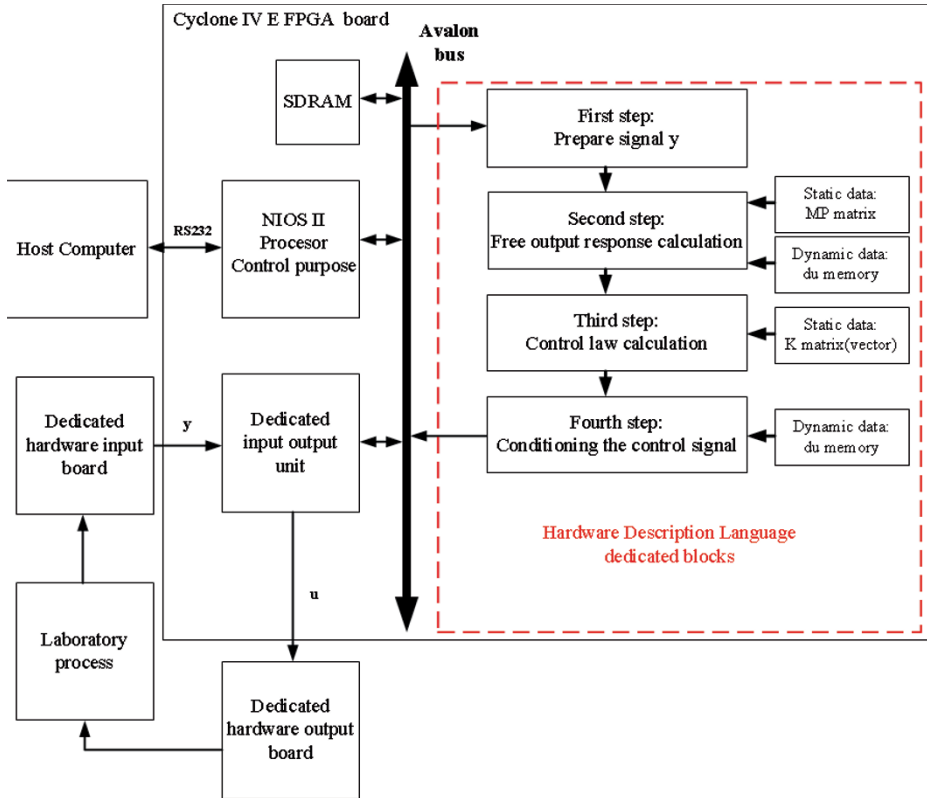


Fig. 1. Block diagram of described system

The second block is Free Output Response Calculation (FORC), where the \mathbf{G}^P matrix and the vector $\Delta \mathbf{u}^P(k)$ are used. The \mathbf{G}^P matrix is given as a hard-coded data. The $\Delta \mathbf{u}^P(k)$ vector is calculated in each iteration of the algorithm. The third block is Control Law Calculation (CLC) where the main calculations are realized. The behavioural description based on C implementation of the DMC algorithm is used. The vector \mathbf{K}_1 and the set-point $y^{SP}(k)$ value must be provided. The first one is hard-coded in HDL language. The second one is taken directly from the NIOS II processor, i.e. from MATLAB control software. The block Conditioning Control Signal (CCS) is used to project the obtained solution onto the admissible set of constraints. This value is sent to the process through NIOS II processor.

A few lines of code for NIOS II are shown below. There are very simple operations on the NIOS II side. At the beginning of one iteration the new process measurement and the set-point value are set. Next, NIOS II waits for a response from hardware blocks which means operation is finished. After this, a new control value is ready and can be sent to the process.

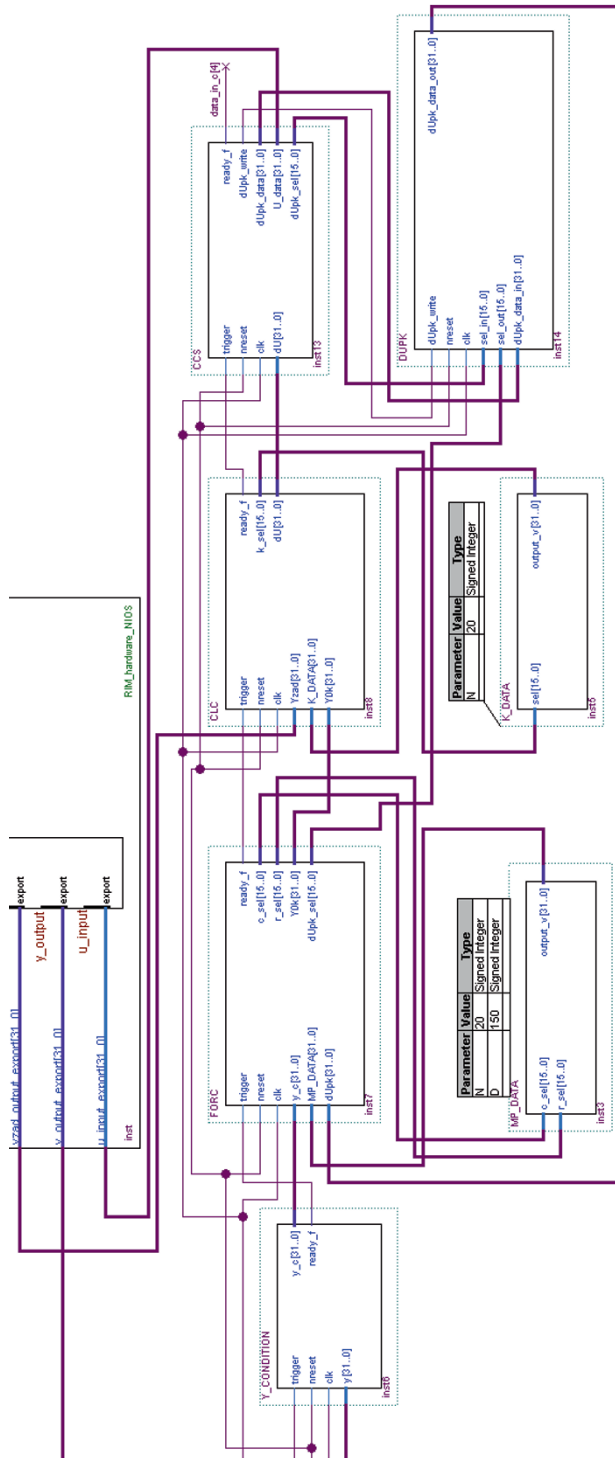


Fig. 2. DMC algorithm function blocks


```

IOWR_ALTERA_AVALON_PIO_DATA(Y_OUTPUT_BASE,y); //put
measure to DMC
IOWR_ALTERA_AVALON_PIO_DATA(YZAD_OUTPUT_BASE,YZAD); //put
setpoint value to DMC
IOWR_ALTERA_AVALON_PIO_DATA(DATA_OUT_BASE,0); //trigger
DMC calculation
while (!IORD_ALTERA_AVALON_PIO_DATA(DATA_IN_BASE)); //wait
for calculation
u_h = IORD_ALTERA_AVALON_PIO_DATA(U_INPUT_BASE); //read
new control value
IOWR_ALTERA_AVALON_PIO_DATA(DATA_OUT_BASE,0); //reset trigger

```

4 Results of Experiments

In the described implementation the development board with Altera Cyclone IV FPGA with 115K logic blocks, 432 M9K memory blocks and 3.888KBits embedded memory is used. The development board is also equipped with 128MB SDRAM, 2MB SRAM, 8MB Flash and 32kb EEPROM. SDRAM is controlled by NIOS II using dedicated interface. The external clock is 25MHz quartz connected to internal PLL unit in FPGA which produce stable 50MHz clock for logic units and NIOS II processor. The whole project uses 15,950 logic elements.

Fig. 3 shows the whole system which consists of the FPGA development board and the laboratory thermal process used [11]. In this work the process has one manipulated variable – the PWM signal connected to the fan F, and one controlled variable – the temperature T. The objective of the DMC algorithm is to adjust the value of the manipulated variable in such a way that the controlled variable follows the changes of its set-point. The connection is made using RS232 and a dedicated protocol implemented for the laboratory stand. Two main instructions are sent to the process: a request for actual value of the temperature of sensor T and a command for the cooling fan F. The heater H is set to constant power without regulation. According to this the process has one input and one output, where the input is flow of air from the fan F and the output is temperature from the sensor T.

To show the experimental results and to provide possibilities of changing the set-point value, a MATLAB GUI is available. Fig. 4 shows example process trajectories demonstrated in the GUI. The trajectories of the manipulated variable (fan F) and the controlled variable (temperature T) are plotted. It is possible to change the required set-point value on-line.

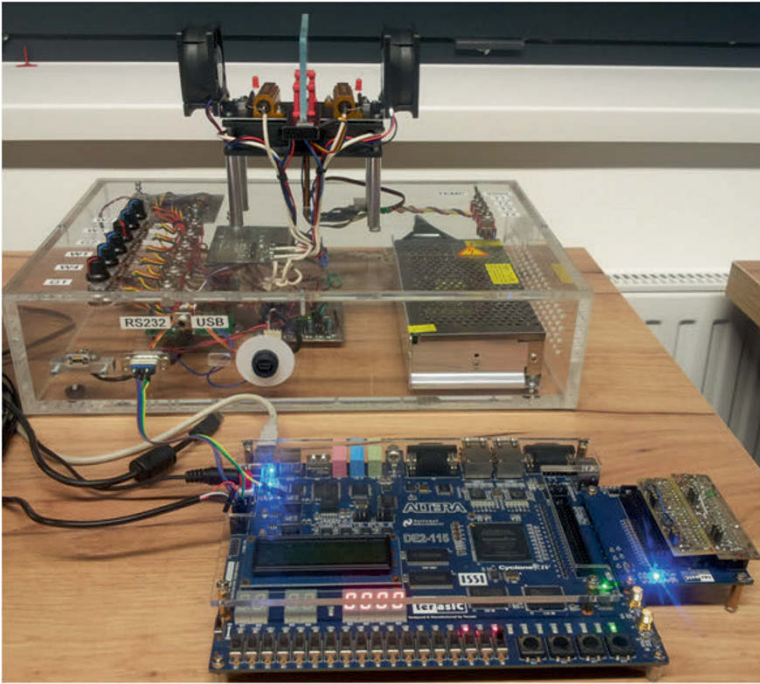


Fig. 3. Test-bench: FPGA development board, laboratory stand

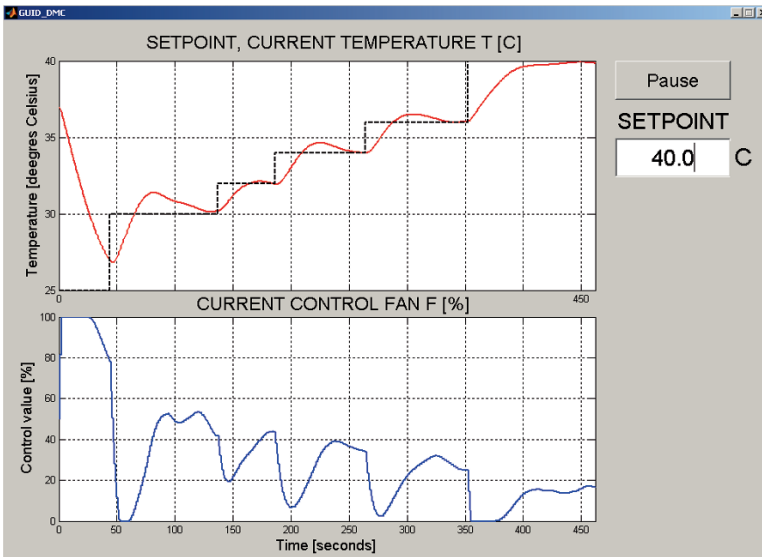


Fig. 4. The example trajectories of the controlled process

5 Conclusions

This work presents implementation of the analytical (explicit) DMC algorithm using the FPGA. The algorithm code is implemented in VHDL language. The software processor NIOS II is used for implementation of the control algorithm in classical way using C language. Implementation of software and hardware versions of the DMC algorithm has been completed. Future work will focus on implementation of other MPC algorithms on the FPGAs, including the MPC algorithms in numerical version, with on-line optimisation. Additionally, it is planned to create a library of MPC algorithms for FPGA which will make it possible to easily and fast prototype new MPC algorithms for numerous practical applications. The final code for the FPGA will be generated automatically, using the libraries and a user definition of the algorithm [12].

References

1. Camacho, E.F., Bordons, C.: *Model Predictive Control*. Springer, London (1999)
2. Chaber, P., Ławryńczuk, M.: Auto-generation of advanced control algorithms' code for microcontrollers using transcompiler. *Proceedings of the 21th IEEE International Conference on Methods and Models in Automation and Robotics, MMAR 2016*, pp. 454–459, Międzyzdroje, Poland (2016)
3. Cutler, C.R., Ramaker, B.L.: Dynamic matrix control – a computer control algorithm. In: *Proceedings of the Joint Automatic Control Conference, San Francisco, USA (1979)*
4. Lan, J., Li, D., Xi, Y., Implementation of Dynamic Matrix Control on FPGA. *Proceedings of the 29th Chinese Control Conference, Beijing*, pp. 5970–5974 (2010)
5. Ling, K.V., Wu, B.F., Maciejowski, J.M.: Embedded model predictive control (MPC) using a FPGA. *IFAC Proceedings Volumes 41 (2)*, 15250–15255 (2008)
6. Ling, K.V., Yue, S.P., Maciejowski J. M.: A FPGA implementation of model predictive control. *American Control Conference* pp. 6 (2006)
7. Liniger, L., Domahidi, A., Morari, M.: Optimization-based autonomous racing of 1:43 scale RC cars. *Optimal Control Applications and Methods 36*, 628–647 (2015)
8. Ławryńczuk, M.: *Computationally Efficient Model Predictive Control Algorithms: A Neural Network Approach, Studies in Systems, Decision and Control*, vol. 3, Springer, Cham (2014)
9. Qin, S.J., Badgwell, T.A.: A survey of industrial model predictive control technology. *Control Engineering Practice 11*, 733–764 (2003)
10. Tatjewski, P.: *Advanced control of industrial processes, structures and algorithms*. Springer, London (2007)
11. Wojtulewicz, A., Chaber, P., Ławryńczuk, M.: Multiple-Input Multiple-Output Laboratory Stand for Process Control Education, 2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR), Międzyzdroje, 2016, pp. 466–471.
12. Petko, M., Staworko, M., Lubieniecki, M.: Automatic software-hardware implementation of process control algorithms using FPGA (in Polish), *Pomiary, Automatyka, Kontrola 5*, 297–300 (2009)

Implementation of DMC algorithm in embedded controller - resources, memory and numerical modifications

Sebastian Plamowski

Institute of Control and Computation Engineering, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland, tel. +48 22 234-76-73
S.Plamowski@ia.pw.edu.pl

Abstract. This paper describes the process of implementation of DMC algorithm in numerical version in embedded environment. The study focuses on the dependences between algorithm parameters and memory and performance requirements. As a result of the investigation the algorithm modifications are presented to achieve compromise between the fast calculations and the memory requirements.

Key words: DMC, predictive control, QP solver, embedded environment.

1 Introduction

Implementation of a computationally complex algorithm is a compromise between resources for memory and CPU usage for calculations. The more memory resources are limited, the more intermediate calculations must be performed during the execution of the algorithm. In the case of calculations being performed on the modern personal computers or dedicated servers the limit resources for memory is usually not a problem. A completely different situation is in the case of implementation of complex algorithms in an embedded environment of programmable logic controllers (PLC) and industrial computers. Industrial computers and PLC controllers are designed to be robust and reliable [6]. Therefore they use proven components not the latest hardware and software solutions. Industrial controllers comparing to the personal computers there are backward about decade. Technological backwardness is caused by two factors. Firstly, the controller processors usually are not cooled. Secondly, controllers use proven hardware and software. Additionally, industrial computers work baseing on the real-time system to be characterized by determinism, which reduces or even eliminates the solutions based on virtual memory. As a result, resources of memory and computing power in an embedded environment are limited and implementation of complex numerical and memory consuming algorithms is a challenge. This article describes how to effectively implement numerical version (with optimization QP) algorithm DMC [1–3,8] as embedded algorithm of industrial computer. The article also presents a software architecture and specifics of the industrial

controller and the resulting limitations. The author shows as restrictions affect the implementation of the algorithm and what modification can be made to run DMC algorithm on the industrial computer.

The paper is organized as follows. In section 2 controller software architecture is presented. Section 3 describes memory requirements for implementation DMC algorithm. Section four presents how to formalize quadratic programming (QP) problem effectively. The concluding remarks is given in Section 5.

2 Software Architecture of Industrial Controller

In the industrial controller, the user can configure the frequency of calls of control tasks, or processes that are responsible for cyclic call computation control structures. Control tasks work in parallel, usually not more than 5 processes and calculations are called from a predetermined (pre-configured) time interval (sampling time) - typically from milliseconds to tens of seconds. The algorithm in the controller is implemented as a function that is called for each instance of the algorithm used in the control structures.

This function is designed so that it provides access to the input configuration parameters of the algorithm and to the state of the algorithm, or the history of the values stored between calls to the function. The function calculates the current output value based on this information. Due to that function design it is necessary to allocate: memory on the parameters of the algorithm, the history of inputs and additional memory is needed for automatic variables (for temporary calculations). It is worth mentioning that in the case of matrix calculations the amount of memory is significant, especially that the allocation is done on the stack.

3 Memory requirements for DMC algorithm

There are different requirements for memory and CPU utilization depending on the type of the algorithm. In the case of the regulator DMC implemented numerically memory requirements are associated with:

- model described by the step response coefficients;
- history of inputs and outputs;
- formalization of the problem QP (matrix calculations).

Demand for memory depends on the algorithm parameters:

- n_{MV} - the number of manipulated variables (MV);
- n_{CV} - the number of controlled variables (CV);
- n_{DV} - the number of disturbance variable (DV);
- D - length of the model answers (number of coefficients);
- N - prediction horizon is defined as the number of sampling periods;
- N_u - control horizon is defined as the number of sampling periods
- N_1 - the shortest delay defined as the number of sampling periods

Table 1: Memory requirements for DMC algorithm

n_{MV}	n_{CV}	n_{DV}	D	N	N_u	Mm	IOM	QPm
6	6	6	200	200	20	60kB	10kB	$M^{MV}=580kB$ $M^{MVP}=5.8MB$ $M^{DVP}=5.8MB$ $H=116kB$ $f=1kB$

Table 1 shows the example of memory requirements for DMC algorithm (assumed $N_1 = 1$).

where:

- Mm - memory for model;
- IOM - memory for history of Inputs and Outputs;
- QPm - memory for formulating the quadratic programming problem;
- M^{MV} - dynamic matrix (4);
- M^{MVP} - matrix for calculation free prediction trajectory dependent on past control values (8);
- M^{DVP} - matrix for calculation free prediction trajectory dependent on past disturbance values (9);
- H - square matrix of quadratic programming problem (2);
- f - vector of quadratic programming problem (3);

Storing the coefficients of the model and the history of inputs and outputs is not difficult from implementation point of view. The main difficulty is in the storage structures used in formulating the quadratic programming problem and effective formulation quadratic programming problem. Computer analysis effort will be presented in the next sections of this article.

4 Formulating of Quadratic Programming Problem

Implementation of the algorithm DMC requires the formulation of quadratic programming problem [5, 8]:

$$\begin{aligned}
 & \min_x \{ J(x) = \frac{1}{2}x^T Hx + f^T x \} \\
 & \text{subject to} \\
 & \quad x_{\min} \leq x \leq x_{\max}, \\
 & \quad Ax \leq b
 \end{aligned} \tag{1}$$

Critical from implementation point of view are calculations of matrix H and vector f , given by the following equations:

$$H = 2((M^{MV})^T Q M^{MV} + R), \tag{2}$$

$$f = -2(M^{MV})^T Q (Y_k^{SP} - Y_k - M^{MVP} \Delta U_k^{MVP} - M^{DVP} \Delta U_k^{DVP}), \tag{3}$$

4.1 Calculation H Matrix

H matrix is calculated basing on the dynamic matrix M^{MV} and matrices of weights Q and R , where:

- Q - quadratic diagonal weights matrix defined for CV signals;
- R - quadratic diagonal weights matrix defined for MV signals;

$$M^{MV} = \begin{bmatrix} S_{N_1} & 0 & 0 & 0 & \dots & 0 \\ S_{N_1+1} & S_{N_1} & 0 & 0 & \dots & 0 \\ S_{N_1+2} & S_{N_1+1} & S_{N_1} & 0 & \dots & 0 \\ S_{N_1+3} & S_{N_2+1} & S_{N_1+1} & S_{N_1} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ S_{N_u+N_1-1} & S_{N_u+N_1-2} & S_{N_u+N_1-3} & S_{N_u+N_1-4} & \dots & S_{N_1} \\ S_{N_u+N_1} & S_{N_u+N_1-1} & S_{N_u+N_1-2} & S_{N_u+N_1-3} & \dots & S_{N_1+1} \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ S_N & S_{N-1} & S_{N-2} & S_{N-3} & \dots & S_{N-N_u+1} \end{bmatrix} \tag{4}$$

Due to the fact that the matrix M^{MV} , Q , and R do not change with each iteration of the algorithm, and the only changes can be made through the tuning process (changing weights and coefficients in step response model) matrix H can be calculated in the user interface and sent to the controller in the completed form.

An important issue is the memory required to store the matrix H . The matrix H is a square matrix of dimensionality $(N_u \times n_{MV}) \times (N_u \times n_{MV})$ and stores numbers in double precision. Due to the fact that the matrix H is a symmetric matrix, it is possible to store only a triangular matrix, which reduces the size of the matrix $(N_u \times n_{MV}) \times (N_u \times n_{MV} + 1)/2$

If the application requires a different weight values for different operating points, then it is required matrix H recalculation in on-line mode. However, the recalculation does not involve counting the entire matrix H , it is possible to implement fast calculation algorithm.

Fast H Matrix Recalculation Algorithm

In the first step weights values should be subtracted, these recalculation should be performed only for diagonal.

$$H = H_{OLD} - R_{OLD} \tag{5}$$

In the second step all matrix H elements (H_{OLD}) should be rescaled.

$$H = H_{\text{OLD}} \frac{Q_{\text{NEW}}}{Q_{\text{OLD}}} \quad (6)$$

In the last step new value of R weights should be updated.

$$H = H_{\text{OLD}} + R_{\text{NEW}} \quad (7)$$

Please note that fast conversion is much more efficient than re-calculation of the whole matrix from the beginning and it doesn't require additional memory for calculations.

4.2 Calculation Vector f

Vector f is calculated basing on the: dynamic matrix M^{MV} , matrices M^{MVP} and M^{DVP} , matrices weights Q and vectors: setpoints Y_k^{SP} , actual process value Y_k and past values of control and disturbance signals ΔU^{MVP} and ΔU^{DVP} . The formulation of these structures is presented below.

$$M^{\text{MVP}} = \begin{bmatrix} S_{1+N_1} - S_1 & S_{2+N_1} - S_2 & \dots & S_{D-1+N_1} - S_{D-1} \\ S_{2+N_1} - S_1 & S_{3+N_1} - S_2 & \dots & S_{D+N_1} - S_{D-1} \\ \vdots & \vdots & \ddots & \vdots \\ S_{N+1} - S_1 & S_{N+2} - S_2 & \dots & S_{N+D-1} - S_{D-1} \end{bmatrix} \quad (8)$$

$$M^{\text{DVP}} = \begin{bmatrix} S_{N_1}^{\text{DV}} & S_{1+N_1}^{\text{DV}} - S_1^{\text{DV}} & S_{2+N_1}^{\text{DV}} - S_2^{\text{DV}} & \dots & S_{D-1+N_1}^{\text{DV}} - S_{D-1}^{\text{DV}} \\ S_{N_1+1}^{\text{DV}} & S_{2+N_1}^{\text{DV}} - S_1^{\text{DV}} & S_{3+N_1}^{\text{DV}} - S_2^{\text{DV}} & \dots & S_{D+N_1}^{\text{DV}} - S_{D-1}^{\text{DV}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ S_N^{\text{DV}} & S_{N+1}^{\text{DV}} - S_1^{\text{DV}} & S_{N+2}^{\text{DV}} - S_2^{\text{DV}} & \dots & S_{N+D-1}^{\text{DV}} - S_{D-1}^{\text{DV}} \end{bmatrix} \quad (9)$$

$$Y_k^{\text{SP}} = \begin{bmatrix} \begin{bmatrix} y_{k+N_1|k}^{(1)\text{sp}} \\ \vdots \\ y_{k+N_1|k}^{(n_{\text{CV}})\text{sp}} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} y_{k+N|k}^{(1)\text{sp}} \\ \vdots \\ y_{k+N|k}^{(n_{\text{CV}})\text{sp}} \end{bmatrix} \end{bmatrix} \quad (10)$$

$$Y_k = \begin{bmatrix} \begin{bmatrix} y_{k+N_1|k}^{(1)} \\ \vdots \\ y_{k+N_1|k}^{(n_{CV})sp} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} y_{k+N|k}^{(1)} \\ \vdots \\ y_{k+N|k}^{(n_{CV})} \end{bmatrix} \end{bmatrix} \quad (11)$$

$$\Delta U_k^{MVP} = \begin{bmatrix} \begin{bmatrix} \Delta u_{k-1|k}^{(1)mv} \\ \vdots \\ \Delta u_{k-1|k}^{(n_{MV})mv} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \Delta u_{k-(D-1)|k}^{(1)mv} \\ \vdots \\ \Delta u_{k-(D-1)|k}^{(n_{MV})mv} \end{bmatrix} \end{bmatrix} \quad (12)$$

$$\Delta U_k^{DVP} = \begin{bmatrix} \begin{bmatrix} \Delta u_{k|k}^{(1)dv} \\ \vdots \\ \Delta u_{k|k}^{(n_{DV})dv} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \Delta u_{k-(D-1)|k}^{(1)dv} \\ \vdots \\ \Delta u_{k-(D-1)|k}^{(n_{DV})dv} \end{bmatrix} \end{bmatrix} \quad (13)$$

Vector f depends on the history of control and disturbance signals (MVs) and (DVs). These signals change at each step, because vector f must be recalculated in each cycle of calculations on the controller. The implementation of the calculations of the vector f is difficult due to the resources limitations of memory and CPU. The first problem, which has already been mentioned is keeping in memory large matrices M^{MVP} and M^{DVP} (each approx. 5.8MB).

It is worth noticing that in the case when multidimensional DMC controller controls low and fast frequency signals the sampling time has to be set adequately to the fast-changing variables. In this case, it may be necessary to use even longer prediction horizon and thus a greater number of factors, which will result in even greater demand for memory resources.

The solution to the problem of storing large matrices M^{MVP} and M^{DVP} can be dynamic counting the elements of the matrix, which minimizes the need

for storage resources but unfortunately increases the effort of calculations. The number of subtraction operations is shown in Table 2.

Table 2: Example: the number of subtraction (*Sub*) operations

n_{MV}	n_{CV}	n_{DV}	D	N	N_u	<i>Sub</i>
6	6	6	200	200	20	1 438 800

In addition, the calculation of the vector f requires matrices multiplication M^{MVP} and M^{DVP} by vectors of past changes of control and disturbance signals (ΔU_k^{MVP} and ΔU_k^{DVP}). The operation must be performed for each cell of the matrices M^{MVP} and M^{DVP} . So the operation matrix multiplication M^{MVP} additionally requires 1 438 800 multiplications and as many addition operations. A similar effort is associated with the operation of multiplication M^{DVP} matrix.

Table 3: Example: the number of subtraction (*Sub*), adding (*Add*) and multiplication (*Multiply*) operations

n_{MV}	n_{CV}	n_{DV}	D	N	N_u	<i>Sub</i>	<i>Add</i>	<i>Multiply</i>
6	6	6	200	200	20	1 438 800	1 438 800	1 438 800

The structures of the matrices: M^{MVP} and M^{DVP} and vectors: ΔU_k^{MVP} and ΔU_k^{DVP} allows us to reduce the number of arithmetic operations in the process of calculation f vector. To do that, the matrices M^{MVP} and M^{DVP} should be split into two submatrices (the same dimension). The approach will be shown for the matrix M^{MVP} , for matrix M^{DVP} algorithm is analogous.

Fast f Vector Calculation Algorithm

In the first step matrix M^{MVP} has to be split into two submatrices M_1^{MVP} and M_2^{MVP} :

$$M^{MVP} = M_1^{MVP} - M_2^{MVP} \tag{14}$$

where:

$$M_1^{MVP} = \begin{bmatrix} S_{1+N_1} & S_{2+N_1} & \dots & S_{D-1+N_1} \\ S_{2+N_1} & S_{3+N_1} & \dots & S_{D+N_1} \\ \vdots & \vdots & \ddots & \vdots \\ S_{N+1} & S_{N+2} & \dots & S_{N+D-1} \end{bmatrix} \tag{15}$$

$$M_2^{\text{MVP}} = \begin{bmatrix} S_1 & S_2 & \dots & S_{D-1} \\ S_1 & S_2 & \dots & S_{D-1} \\ \vdots & \vdots & \ddots & \vdots \\ S_1 & S_2 & \dots & S_{D-1} \end{bmatrix} \quad (16)$$

The rows of M_2^{MVP} matrix are identical, so obviously calculations should be done only for one row.

Reduction of calculations for matrix M_1^{MVP} is presented by the example. Let's assume the following matrix M_1^{MVP} :

$$M_1^{\text{MVP}} = \begin{bmatrix} S_2 & S_3 & S_4 & S_5 & S_6 & S_7 & S_8 & S_9 \\ S_3 & S_4 & S_5 & S_6 & S_7 & S_8 & S_9 & S_{10} \\ S_4 & S_5 & S_6 & S_7 & S_8 & S_9 & S_{10} & S_{11} \\ S_5 & S_6 & S_7 & S_8 & S_9 & S_{10} & S_{11} & S_{12} \end{bmatrix} \quad (17)$$

Based on the equation (3) matrix M_1^{MVP} is multiplied by ΔU^{MVP} vector. The result as a component of free prediction trajectory can be described by equation (18) - for simplification only first MV signal is considered and calculations for $k+3$ moment (performed at current k moment) are presented.

$$y_{k+3|k}^{\text{cf}} = s_4 \Delta u_{k-1|k}^{(1)\text{mv}} + s_5 \Delta u_{k-2|k}^{(1)\text{mv}} + s_6 \Delta u_{k-3|k}^{(1)\text{mv}} + s_7 \Delta u_{k-4|k}^{(1)\text{mv}} + s_8 \Delta u_{k-5|k}^{(1)\text{mv}} + s_9 \Delta u_{k-6|k}^{(1)\text{mv}} + s_{10} \Delta u_{k-7|k}^{(1)\text{mv}} + s_{11} \Delta u_{k-8|k}^{(1)\text{mv}} \quad (18)$$

The same calculations for $k+2$ moment performed at the next $k+1$ moment are described by (19).

$$y_{k+2|k+1}^{\text{cf}} = s_3 \Delta u_{k-1|k+1}^{(1)\text{mv}} + s_4 \Delta u_{k-2|k+1}^{(1)\text{mv}} + s_5 \Delta u_{k-3|k+1}^{(1)\text{mv}} + s_6 \Delta u_{k-4|k+1}^{(1)\text{mv}} + s_7 \Delta u_{k-5|k+1}^{(1)\text{mv}} + s_8 \Delta u_{k-6|k+1}^{(1)\text{mv}} + s_9 \Delta u_{k-7|k+1}^{(1)\text{mv}} + s_{10} \Delta u_{k-8|k+1}^{(1)\text{mv}} \quad (19)$$

Due to the fact that:

$$\Delta u_{k-2|k+1}^{(1)\text{mv}} = \Delta u_{k-1|k}^{(1)\text{mv}} \quad (20)$$

equation (18) can be transformed to (21)

$$y_{k+2|k+1}^{\text{cf}} = s_3 \Delta u_{k-1|k+1}^{(1)\text{mv}} + y_{k+3|k}^{\text{cf}} - s_{11} \Delta u_{k-9|k+1}^{(1)\text{mv}} \quad (21)$$

Summarizing, calculating prediction for a $k+n$ moment can use value calculated in previous step for a $k+n+1$ moment. This fact allows to effectively reduce the number of mathematical operations and finally reduce the time of calculations. However, this approach needs to use additional memory for storing free prediction components between algorithm steps. Calculation cost and memory requirements are presented in tables 4 and 5.

Algorithm reduces the number of mathematical operations several hundred times comparing to the classical approach. The amount of additional memory is on acceptable level.

Table 4: Number of subtraction (*Sub*), adding (*Add*) and multiplication (*Multiply*) operations after optimization

n_{MV}	n_{CV}	n_{DV}	D	N	N_u	<i>Sub</i>	<i>Add</i>	<i>Multiply</i>
6	6	6	200	200	20	2 400	2 400	3 600

Table 5: Memory requirements

n_{MV}	n_{CV}	n_{DV}	D	N	N_u	<i>Memory</i>
6	6	6	200	200	20	4.8kB

5 Conclusions

Described numerical modifications make it possible to effectively implement DMC algorithm in embedded environment of industrial controllers in numerical version (with QP optimization). Described modifications are useful for DMC algorithms with big number of inputs and outputs and long model dynamic. The advantage of described approach is the possibility to reduce time of computation of DMC controller in original version, where each point on prediction horizon is optimized. In the case when time of calculation will be still too long described technique can be combined with Coincidence Point Control technique [4, 7] - this direction will be explored in the future investigation.

References

1. Camacho, E.F., Bordons, C.: Model Predictive Control. Springer, London (1999)
2. Clarke, D.W., Mohtadi, C., Tuffs, P.S.: Generalized predictive control. *Automatica* 23 137–160 (1987)
3. Cutler, C. R., Ramaker, B. L.: Dynamic matrix control – a computer control algorithm. In: Proceedings of the Joint Automatic Control Conference, San Francisco, USA (1979)
4. Haber, R., Bars, R., Schmitz U.: Predictive Control in Process Engineering: From the Basics to the Applications. Wiley, United Kingdom (2011)
5. Maciejowski, J. M.: Predictive control with constraints. Prentice Hall, Englewood Cliffs (2002)
6. Qin, S.J., Badgwell, T.A.: A survey of industrial model predictive control technology. *Control Engineering Practice* 11, 733–764 (2003)
7. Rossiter, J.A., Haber, R.: The Effect of Coincidence Horizon on Predictive Functional Control. *Processes* 2015, 25–45 (2015)
8. Tatjewski, P.: Advanced control of industrial processes, structures and algorithms. Springer, London (2007)

Real-Time Basic Principles Nuclear Reactor Simulator Based on Client-Server Network Architecture with WebBrowser as User Interface

Dymitr Juszczyk, Jarosław Tarnawski, Tomasz Karla, Kazimierz Duzinkiewicz

Department of Control Engineering, Faculty of Electrical and Control Engineering,
Gdansk University of Technology, Gdansk, Poland
dymitrjuszczyk@gmail.com, jaroslaw.tarnawski@pg.gda.pl,
tomasz.karla@pg.gda.pl, kazimierz.duzinkiewicz@pg.gda.pl

Abstract. The real-time simulator of nuclear reactor basic processes (neutron kinetics, heat generation and its exchange, poisoning and burning up fuel) build in a network environment is presented in this paper. The client-server architecture was introduced, where the server is a powerful computing unit and the web browser application is a client for user interface purposes. The challenge was to develop an application running under the regime of real-time, with a high temporal resolution, in an environment which is not a native real-time. The problem of a real-time operation taking into account the variable time of calculations and a communication latency was solved using the developed mechanism of step length adaptation. Results of multiple studies of a numerical compliance with the reference simulator proved correctness of the developed application.

1 Introduction

Simulators can be used for educational and dissemination purposes supporting the classical teaching process. Real-time simulators have the additional property because they offer a familiarization with the dynamics of processes and their temporal interrelationships. The basic principles nuclear reactor (NR) simulator presented in this paper is designed to work in a real-time based on a network environment and a web browser which does not work natively in a real-time regime. The real-time simulator is expected to deliver simulation results at certain moments in time, on-line, with a reference 1:1 to a real-time. A simulation time resolution associated with a simulator step is a very important parameter. Acquisition of input signals, mathematical calculations and presentation of simulation variables in one simulator iteration shall be performed faster than a simulation step. The real-time simulator needs a fast, time-efficient solver able to generate results in a finite, predictable time to produce the results with a prescribed temporal resolution. The deterioration in the accuracy of calculations is expected comparing to non real-time simulators, but a numerical stability must be ensured.

The subject of non real-time simulation of nuclear reactor basic principles can be found among others in [1]. The conception of the real-time simulation with a web browser as a user interface can be found e. g. in [2] but this application is not used with processes requiring a high computing performance like NR processes.

Authors previously developed a cross-platform real-time NR simulator targeted for different hardware-software platforms (like PC or RaspberryPi) [3]. In the following paper [4] authors introduced the algorithm for maintaining a real-time operation in case of delays in simulations in non real-time environments. The usage of this method on different hardware platforms revealed the restrictions on the length of a simulation step and the need for estimating a maximal length of a simulation step for specific hardware-software platforms [5]. Previous works were concerned only about a standalone simulator, without the usage of a network, a web browser or simultaneous access of many users. The comprehensive theory on a real-time simulation with many applications in different areas can be found in the book [6].

2 Physical and control processes in the nuclear reactor and their temporal characteristics

One of the simplest models but well reflecting the nature of the processes in the NR is the point model [7], in which all variables are averaged over the volume of the core. Figure 1 presents processes and control effects of NR included in presented simulators. All of these processes have different time scales. Table 1 summarizes the various processes and their transition times.

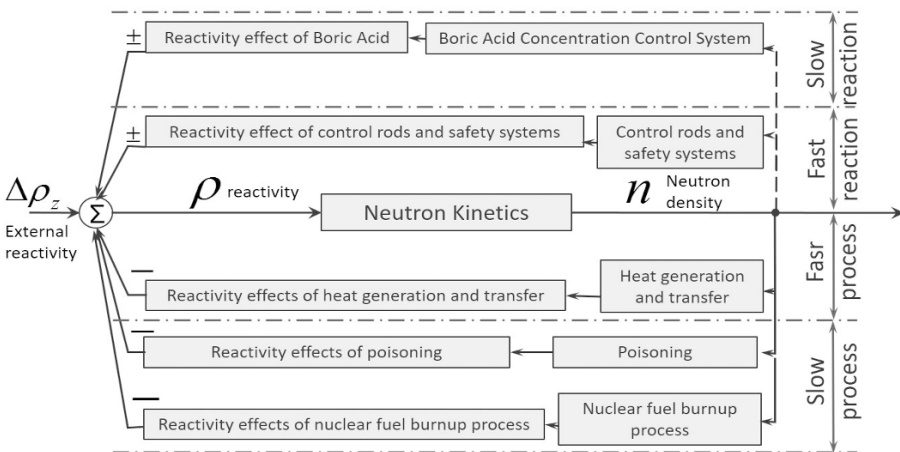


Fig. 1. Main processes occurring in a nuclear reactor affecting its state

Another approach to the modeling of nuclear processes and equipment can be found e.g. in [8], [9]. The comprehensive information about reactor physics and related topics can be found in [10], [11].

Table 1. Processes occurring in the reactor and their duration of transient states

The process	The duration of transient states
Neutron kinetics	1E-5 to 1E-3 seconds
Heat transfers between fuel and coolant/ moderator, Changes in reactivity resulting from changes in temperatures of the fuel and the coolant/ moderator	3 to 6 minutes
Changes in reactivity caused by changing positions of control rods	up to 125 s
Changes in reactivity caused by changing the concentration of boric acid in the moderator	several hours
The xenon poisoning occurring while changing the power level of the reactor	up to 60 hours
The samarium poisoning	up to 60 hours
The nuclear fuel burnup	tens of days and months

3 Simulator design assumptions and requirements

The aim of the study was to develop a real-time basic principle nuclear reactor simulator operating in the regime of real-time in a network environment that can support the interface prepared in a web browser. Specific assumptions include:

- an implementation of mathematic point model of nuclear reactor basic processes,
- a real-time operation in a network environment with the usage of Internet network,
- the ability to present the results to the users and input control signals to the simulator from multiple clients simultaneously,
- the user interface in the form of the web browser application,
- the ability to change while in a simulation following parameters: the level of the control rods, the density of boric acid, the coolant temperature at the inlet to the reactor,
- the numeric compatibility with the reference simulator build in the MATLAB/Simulink environment with the ability of archive data in a form compatible with the MATLAB for further processing,
- the software-hardware independence on the client side,
- the client-server topology with the use of the widely available Ethernet and the TCP protocol.

Fulfilling all the assumptions, especially working in the regime of real-time in a networked environment required development of a mechanism to adapt the computation step that would provide a real-time simulation in environments without real-time kernel even in a case of a large load on the server and the network at the expense of compromising the accuracy of the calculations.

4 The issue of nuclear reactor mathematical model real-time calculations in a network environment

The mathematical model mentioned in the previous chapter consists of 18 differential equations and several algebraic equations. Due to different scales of dynamics the problem called stiffness can be expected during calculations. In [12] it can be read "An ordinary differential equation problem is stiff if the solution being sought is varying slowly, but there are nearby solutions that vary rapidly, so the numerical method must take small steps to obtain satisfactory results". Unfortunately, exactly the kind of situation was faced during the construction of the basic principle nuclear reactor simulator. To calculate the results of the differential equations, the stiff or non-stiff methods can be used. Again quoting [12] "Stiffness is an efficiency issue. If we were not concerned with how much time a computation takes, we would not be concerned about stiffness. Non-stiff methods can solve stiff problems; they just take a long time to do it". Unfortunately, in real-time simulations implementation of both stiff or non-stiff methods with very small steps and long calculations cannot be applied. Methods with a fixed or variable step can be used to solve systems of differential equations. Constant step methods are easier to implement. In this article the method with a variable step was applied not only from the need to step reduce but also to take account of major changes and to avoid stiffness. In a non real-time environment unexpected incidental delays may occur due to e.g. communication, data loading and operating system interrupts. Then there may be situations where the computation time for a single-step of a simulation exceeds a prescribed simulation step. To maintain the synchronization with a simulation time lengthening simulation steps is used - in order to catch up a simulation time. The multi-step catch up simulation time algorithm is described in the following section. The most commonly used methods for solving differential equations are the Euler or Runge-Kutta methods. Higher order methods are numerically stable and more accurate. Taking into account all the above issues the variable step Runge-Kutta (specifically fourth order) method was applied in the simulator.

5 The mechanism of a simulation step adaptation in the web simulator

The real-time supervision mechanism was located on a client side to ensure that the user gets simulation results on time. The time between sending the request of the results from the server and getting answer is counted by the supervision

mechanism. This value is used to determine if current data is on time. The algorithm of step adaptation will be used when the mechanism detects the delay in data delivery. The delay means difference between the simulation time and the real-time. It must be compensated by algorithm. The basis for the development of a step adaptation mechanism is presented in [4]. The idea assumes that in case of a simulation delay, the next simulation step will be as long as the previous cycle time. It is assumed that occurred delay will cause the same communication time extension in the next program cycle. The new simulation step is then extended to multiplicity of the base step to maintain the real-time regime. The principle of step adaptation is shown on fig.2 chart a).

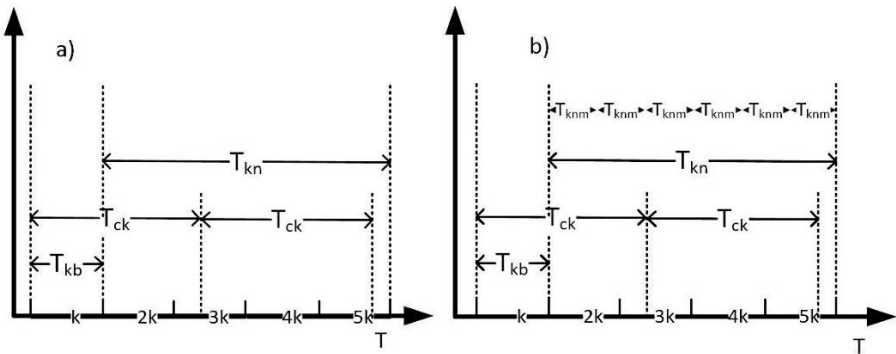


Fig. 2. Charts showing the idea of step adaptation

The base steps are indicated on the x-axis. The communication starts at the beginning of the k period. The cycle of the program should be shorter than the established simulation step T_{kb} . The cycle, however, exceeds simulation step and is T_{ck} , almost two and half-step base. Assuming that the next cycle time also will be T_{ck} , the next calculation point was set at the end of $5k$ period. The new simulation step is the period from the end of the last step to a new calculation point (T_{kn}). A new simulation step has the length of four base steps. Such a step would compensate the delay between the simulation time and the real-time. However, the studies have shown that eliminating the delay in one step has a big impact on the quality of the simulation results and can lead to a numerical instability. To prevent that, the new simulation step is calculated by an operation that divides the base simulation step T_{kn} on the m shorter periods (shown on fig.2 chart b)). This operation will achieve the final step in several cycles of a simulation and reduce the negative effects of its changes. The current step will be increased by T_{knm} period in each simulation cycle up to the final step T_{kn} .

6 The realization of network simulator

6.1 Hardware-Software platform of simulator

After considering all the requirements and objectives software-hardware environment for the implementation of the simulator was selected. The hardware platform for the simulator server was a mainstream Intel i7 PC class computer with the Ubuntu 14.0 operating system. Such configuration has been chosen because of their popularity, availability, reliability and software security in server environments. In addition, the Node.js environment was used which made it possible to ensure the real-time communication with clients that use web browsers. The only requirements on the client side were access to the Internet network and a modern web browser.

The client hardware-software platform can be any device that utilize Ethernet networks (including mobile, e.g. using Wi-Fi) which are equipped with a web browser compliant with modern standards of displaying Web pages (the JavaScript is required).

6.2 Web communication methods used in simulator

It is assumed that only one client device is a real-time maintaining unit in a simulation. The user interface has authorisation system with three types of users. Simulation can be configured and run only by the user with full permissions (administrator). The rest of the users participate in the real-time simulation maintained by the administrator unit. However, non administrator users have the ability to change the control values and have an impact on the simulation.

In order to achieve the two-way real-time communication between clients and the server, WebSocket (WS) technology was used. WS technology offers standard virtual channels, allowing the server to communicate with some or all users. The protocol creates a "Socket" in a browser that have the IP address and the port, maintains a two-way, simultaneous communication between the client and the server. The technology uses the TCP transport layer and allows to bypass the interpretation of messages over HTTP. WS reduces the unnecessary network traffic and shortens the waiting time for the information in the relation to conventional methods of a communication, because if the communication is once established the server can send informations at any time, as soon as they are ready to transfer.

6.3 The user interface of web simulator

The user interface was made as a web browser application. Static elements on the site were designed using HTML5. Animations, an event handling and data operations were developed in the JavaScript. The user interface was designed to present as much relevant information and be able to adjust the displayed objects to the client needs. User interface is presented on fig.3.

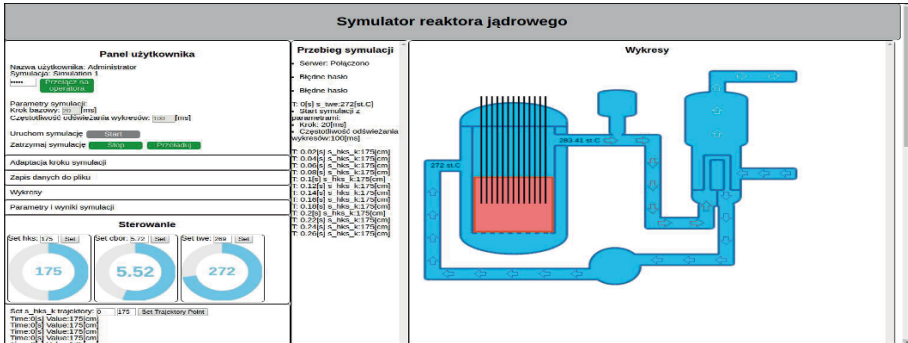


Fig. 3. The user interface

The interface consists of three main columns. The left column of the interface is a control panel, where objects are arranged for the simulation control and interface. The middle column is an area for record of all important actions during the simulation. The right column is an area for displaying charts of all available values. The user can adjust number and type of displayed charts.

6.4 The data archiving

The application has the ability to save data to a file. The user can select the values wanted to store, and then download the results in the zip file after the simulation. The file contains a text in form of a matrix with data, where the columns are separated by commas and rows are separated by semicolons. The columns represent selected variables in the similar order to that in the user interface.

7 Simulator tests

7.1 The verification of simulator

The verification of the correct operation of the simulator and the quality of its results was made by comparing its simulation results with the reference simulator developed in the MATLAB / Simulink. The reference trajectory of the levels for the control rods of the reactor was prepared for both simulators. It assumed a step change in the set point position of control rods at defined levels of 175 cm, 225 cm, 200 cm and 150 cm changing in the sequence every 120 s. The results of the reference simulator have been saved directly into the MATLAB workspace. The simulation on the network simulator was initiated by one of the clients from the web browser and the results were saved to the file, which later also have been imported into the MATLAB workspace where the comparison with the reference simulator was made. The network simulator worked with the mechanism of step adaptation in the simulation with the initial step of the calculation set on 20 ms.

The reference simulator base step was determined to be 10 ms. Waveforms of neutron density were selected to the comparison. The resulting waveforms from both simulators rearranges fig.4.

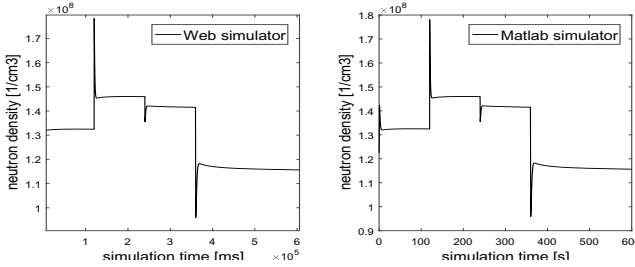


Fig. 4. The results of simulations - values of neutron density

The visual analysis of neutrons density waveforms did not show any significant deviations between the two simulators. The dynamics of the processes were recreated with the acceptable accuracy.

The direct numerical analysis of the results of both simulations was not possible due to the step-variable network simulator, which varied between 20 - 50 ms. To determine a quantitative measure of the quality of the simulation, the network simulator results were interpolated using the linear interpolation to match the number of the measurements of the reference simulator. The simulation errors were calculated using (1):

$$\sigma = \frac{\sum |\Delta X|}{\sum X} * 100\% \quad (1)$$

where: ΔX – the difference between the pair of results from the network and the reference simulator at the specific time, X – the result of the reference simulator at the specific time

The calculated average error was 0.0064%. The largest relative errors could be observed at the time of a step change in the position of control rods. The maximal observed relative error of simulation results was 17.9602%, wherein it could be observed by only one step of the simulation. It is fully justified given the nature of the network simulator, the minor latency and the step change in the reference signal.

In the case of this simulator, in which changes in control signals do not occur too often, such errors have a marginal effect on the quality of the simulation. The resulting low average error proves the numerical correctness of the developed simulator.

7.2 The step length impact on a calculations quality

The quality tests in the network load conditions were performed. During the simulation repeatedly occurring delays were observed, resulting in frequent changes

of the simulation step length. The chart of neutrons density acquired during the test is presented in fig.5. Changes in the waveform of neutrons density not caused by the operation of the control system can be observed in the marked areas. The noise in waveforms can be seen especially in the area of the third minute of simulation.

Due to the impossibility of the stiffness elimination in the real-time simulator the numerical stability analysis with different simulation step lengths were performed. Increasing the simulation step, the extortion in the form of changing the position of the control rods and other factors were applied. Thus it was estimated that for the presented mathematical model and specified solver, numerically stable results can be obtained when the step length is not more than 50 ms.

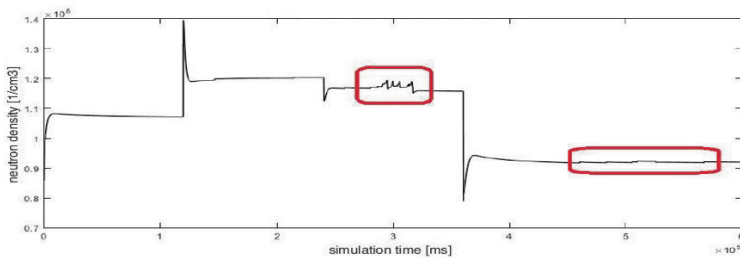


Fig. 5. The neutrons density waveform noise caused by aggressive simulation step changes under high load conditions

7.3 The web environment impact on a quality of simulation

The study involving the observation of the simulation step while communicating with different numbers of users was performed. The study began with four connected users. The application was tested with the maximum number of 15 clients. The negative impact on the communication time with the server was observed, what caused simulation delays when connecting with a larger number of clients at the same time. However, no impact on the step length of the continuously connected clients were found, because the server generates and sends the same version of the results for a group of connected clients in every cycle. In summary, the effect of the number of clients connected applications at the same time has a marginal impact on the quality of the simulation.

8 Summary and conclusions

The concept of the real-time basic principles nuclear reactor simulator based on the network environment is presented in the paper. The main considered problem was to achieve the numerically stable real-time simulation with the high

temporal resolution in the environment which is not a real-time for the plant with the extremely different time scales. The multi-step adaptation algorithm of the simulation step length was proposed for dealing with the real-time. The simulator program in the client-server architecture in the network environment was developed. Several studies were performed to test the developed simulator. The high compliance with the reference simulator was achieved and the maximum length of the step size was determined for ensuring the numerical stability. The only free and open software environments were applied. The user interface of the developed simulator is based on the web browser accessible on almost every operating system and computer hardware including the mobile. As a result of work undertaken in the article very useful education tool with the huge dissemination potential was developed.

References

1. *Nuclear Power Plant Simulator*, <http://www.nuclearpowersimulator.com/>, (access date 15.01.2017)
2. Rui Wu et al., *A Real-time Web-based Wildfire Simulation System*, IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 2016, pp. 4964-4969
3. Karla T., Tarnawski J., Duzinkiewicz K., *Cross-Platform Real-Time Nuclear Reactor Basic Principle Simulator*, 20th International Conference on Methods and Models in Automation and Robotics (MMAR), 2015, pp. 1074-1079. IEEE
4. Tarnawski J., Karla T., Rutkowski T.A., Puchalski B., Duzinkiewicz K., *Soft real-time simulation with adaptive step of computation*, The Scientific Papers of Faculty of Electrical and Control Engineering, Gdansk University of Technology, 2015
5. Tarnawski J., Karla T., *Real-time simulation in non real-time environment*, 21st International Conference on Methods and Models in Automation and Robotics (MMAR), 2016, pp. 577-582
6. Popovici K., Mosterman P. J. (Eds.), *Real-time simulation technologies: principles, methodologies, and applications.*, CRC Press, 2012
7. Karla T., Tarnawski J. Duzinkiewicz K., *Symulator czasu rzeczywistego procesow reaktora jądrowego*, Aktualne Problemy Automatyki i Robotyki, Akademicka Oficyna Wydawnicza EXIT, 2014, pp. 558-569
8. Sokolski P. Rutkowski T.A. Duzinkiewicz K., *Simplified, multiregional fuzzy model of a nuclear power plant steam turbine.*, 21st International Conference on Methods and Models in Automation and Robotics (MMAR), 2016, pp. 379-384
9. Puchalski B., Rutkowski T.A., Duzinkiewicz K., *Multi-nodal PWR reactor model - methodology proposition for power distribution coefficients calculation.*, 21st International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 385-390
10. Oka Y., Suzuki K., *Nuclear Reactor Kinetics and Plant Control*, Springer, 2013
11. Hetrick D. L., *Dynamics of Nuclear Reactors*, The University of Chicago Press, Chicago and London, 1971
12. Moler C., *Stiff Differential Equations*, Technical Articles and Newsletters, MathWorks, <https://www.mathworks.com/company/newsletters/articles/stiff-differential-equations.html>, (access date 15.01.2017)

Object Tracking With the Use of a Moving Camera Implemented in Heterogeneous Zynq System on Chip

Marcin Kowalczyk¹ and Tomasz Kryjak¹

AGH University of Science and Technology, Al. Mickiewicza 30, 30-059 Krakow,
Poland

mkowalcz@student.agh.edu.pl, tomasz.kryjak@agh.edu.pl

Abstract. In this paper a hardware-software implementation of an object tracking system, which uses a moving camera is described. Selected algorithms: mean-shift, particle filter, KLT and so-called tracking by detection were analysed and evaluated. Particular attention was paid to the effectiveness of fast moving object tracking and the ability to implement the algorithm in a heterogeneous computing system. The selected solution was implemented in the Zynq SoC (System on Chip) device from Xilinx. Object position was used to control two servomotors, which constituted a pan-tilt mounting of the camera. Additionally, object position prediction was realised using a Kalman filter. The proposed system is able to process a 1280×720 @ 60 fps video stream in real time and track moving objects.

Keywords: object tracking, moving camera, Kalman filter, hardware-software co-design, Zynq SoC

1 Introduction

Object tracking with the use of a moving vision camera or thermal imaging camera is used, among others, in advanced video surveillance systems and many military applications. This type of systems consist of two main components: an object tracking algorithm and moving camera head control algorithm. In most cases, it is assumed that the aim is to keep the tracked object in centre of the frame.

Designing this type of vision system is a quite difficult task. Firstly, effective tracking requires the use of a method prone to scene lighting changes, rapid object movement, as well as size, shape and orientation changes. Moreover, in the case of a moving camera, image blur could also be a problem. Secondly, an important decision is the choice of the used computing platform. It should allow real-time implementation of the tracking algorithm and also be quite energy effective for many applications. Additionally, an easy integration with the moving camera head should be possible. According to the authors opinion, the above defined requirements are very well met by a Zynq SoC (System on Chip)

device, which combines in one housing reprogrammable logic (FPGA – Field Programmable Gate Array) and a processor system based on a dual-core ARM Cortex A9 unit.

In this paper the concept of using the Zynq SoC device for constructing a prototype of a moving smart-camera able to perform tracking of selected objects is investigated. The main contributions of this paper are:

- design of a fully operational prototype of a moving smart-camera (camera, Zynq computing platform, two servos, PID controller, Kalman filter),
- working system evaluated for two algorithms for 1280×720 @ 60 fps video stream.

The rest of this paper is organized as follows. In Section 2 previous work on object tracking implemented in FPGA devices is discussed. In the next Section 3 the proposed hardware-software system is presented. Its evaluation is shown in Section 4. The paper ends with a short summary and further research direction discussion.

2 Previous Work

The issue of implementing object tracking using a moving smart camera based on a FPGA or Zynq SoC device has been presented in several research papers. In the work [13] an original concept of a smart camera was proposed. It was based on a Spartan 6 FPGA device connected with 8 SRAM (Static Random Access Memory) banks. The camera was mounted in a housing, which allowed movement in two dimensions: rotation (360°) and tilt ($0 - 90^\circ$). Moreover, a tracking algorithm was proposed, which was based on edge movement analysis between consecutive frames. Finally, real-time processing for a 640×480 @ 20 fps video stream was obtained.

In the paper [4] a two camera system able to track circular shape objects was presented. Used algorithm was based on edge detection and circle fitting. The authors reported processing speed of over 1000 fps for a 816×600 pixels video stream. The system has been implemented in a Zynq device on the ZC 706 platform from Xilinx.

In a recently published article [7] a hardware-software tracking system using a pan-tilt camera was described. It used object detection based on colour and edge information. For servomotors control a PID controller was used. The whole system has been implemented in a Zynq device present on the ZedBoard development board. The authors report 58 fps processing speed, but did not provide information about the image resolution.

In the article [5] a vision system for a mobile robot implemented in FPGA was presented. The used algorithm was based on orange object detection. According to the description, the algorithm was realized on the PowerPC processor (with Linux operating system). No unambiguous information about the performance was provided (the used camera supported 640×480 @ 30 fps).

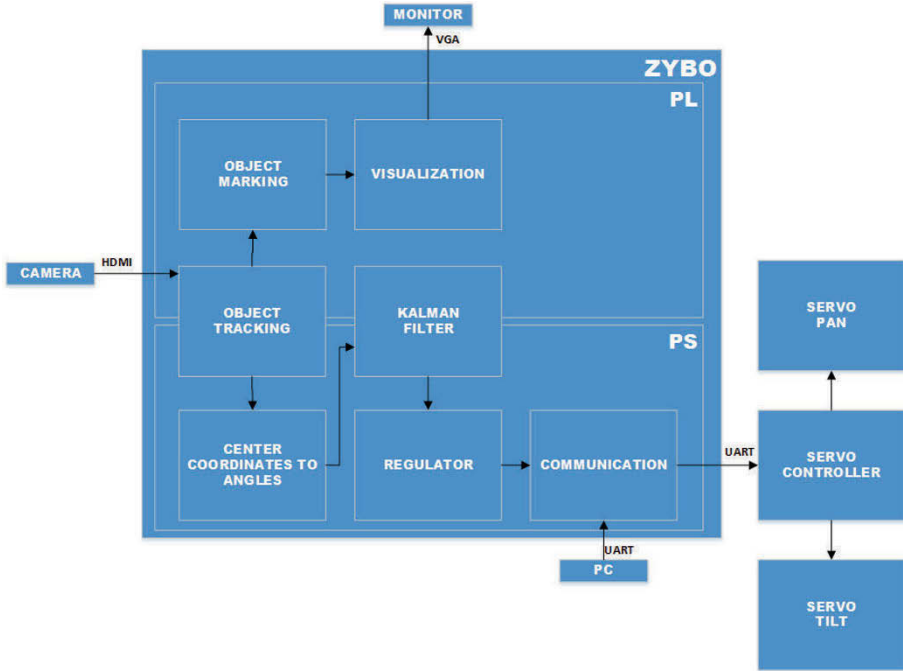


Fig. 1. Scheme of the proposed hardware-software system for object tracking using a moving camera.

In the work [3] a feature point (corners) based tracking algorithm was presented. Also the pyramidal Lucas-Kanade optical flow and Kalman filter were used. All computations were realized on a GPU (Graphics Processing Unit). The authors reported real-time processing for a $656 \times 524 @ 30$ fps video stream.

3 The Proposed Vision System

A general scheme of the proposed smart-camera prototype is presented in Figure 1. It consists of the following components.

- Xiaomi Yi Action YDXJ01XY camera – a $1280 \times 720 @ 60$ fps video stream was processed. It was transmitted to the computing platform via a HDMI (High Definition Multimedia Interface) connection. Additional, on top of the camera a laser pointer was installed. This allowed for better visualization of object tracking results.
- ZYBO development board manufactured by Digilent with Zynq 7010 (PS – processing system (dual ARM core), PL – programmable logic (FPGA)) from Xilinx. Moreover, the HDMI input, VGA output and serial port (USB to UART) were used.
- monitor – for tracking algorithm visualization,

- two digital servomotors PowerHD D-21HV connected into a PT (pan-tilt) configuration.
- servomotor control unit 'MiniMaestro' manufactured by Pololu (communication via UART).

In the following subsections a detailed description of the modules implemented in the Zynq device is provided.

3.1 Object Tracking

At the conceptual stage, the following object tracking algorithm were considered: simple tracking by detection (e.g. object segmentation using colour features), mean-shift [2], particle filter [10], KLT (Kanade, Lucas and Tomasi) [12].

During preliminary work the listed algorithms were evaluated on a sample video sequences containing a moving drone. At this stage the *Matlab* environment was used. The simple tracking by detection was carried out on the basis of the object colour. First RGB colour component ranges were defined and then a pixel was compared with them to check if it should be considered as a part of the object. In the next step, the centroid of the selected pixel was computed. The following equations were used:

$$x_c = \frac{m_{10}}{m_{00}}, \quad y_c = \frac{m_{01}}{m_{00}} \quad (1)$$

$$m_{00} = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} x_{ij}, \quad m_{10} = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} i \cdot x_{ij}, \quad m_{01} = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} j \cdot x_{ij} \quad (2)$$

where: N and M are the image horizontal and vertical sizes, and x_{ij} is a binary pixel.

The point x_c, y_c was assumed as the tracked object's location. During real-life experiments a green object was used, so the following range of RGB components were applied:

$$R \in [0, 30], \quad G \in [40, 255], \quad B \in [0, 30] \quad (3)$$

The FPGA implementation of this method was quite straightforward. To obtain the object mask six comparators and some basic logical operations were required. The computation of m_{00}, m_{01} and m_{10} was done with simple accumulators. For the final division an iterative algorithm was used, as it was needed only once per single frame.

The following algorithm implementations were used: mean-shift [1], particle filter [11] and KLT (build-in in *Matlab*). The results analysis allowed to draw the following conclusions. The simple tracking by detection based on colour features worked correctly when the considered object was clearly distinguishable from the background. Its key advantage was the very simple FPGA implementation. An example tracking result is presented in Figure 2(a). It should be noted that in this version the method should only be used for testing and demonstration of the system. However, the concept could be applied, assuming the use of a more

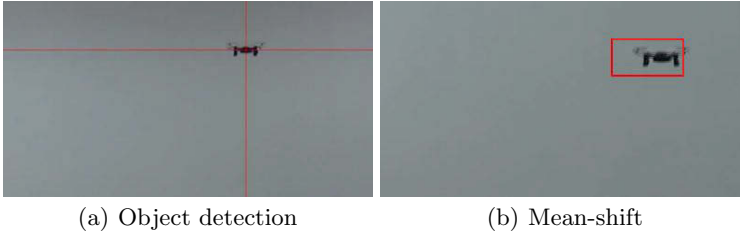


Fig. 2. Sample frames from the tracking algorithms evaluation on the drone sequence.

advanced object detection algorithm e.g. the well-known HOG (Histogram of Oriented Gradients) features and SVM (Support Vector Machine) classifier or even deep convolutional neural networks.

The mean-shift algorithm worked correctly only if the object moved relatively slowly. This was a direct consequence of the method's working principle. The new object location in the current frame is sought in a defined surrounding of the previous one. Moreover, due to the iterative procedure, the hardware implementation of this algorithm is quite complex [8]. An example tracking result is presented in Figure 2(b). Similar observations were also made for the particle filter algorithm. Furthermore, experiments described in the work [9] showed that an effective implementation of this algorithm on the ZYBO platform is impossible due to limited hardware resources.

The last of the evaluated algorithms – KLT – provided the best results. The tracking was correct, even in case of sudden movements. However, the hardware implementation of this algorithm (especially it's multi-scale version) is also quite complex [6] and is planned as part of the future research.

3.2 Centre Coordinates to Angles

The coordinates computed in programmable logic (using the tracking by detection or mean-shift method) had to be converted into angle control errors. This was done on the basis of input image resolution and camera's angle of view. It was assumed that the distance from the image centre is proportional to the angle error. This is generally not true, however for small error values it is a good approximation. Using the exact value would require the use of a look-up table (distance in pixels to angle assignment) or the determination of a mathematical function connecting those two values. Finally, the following equations were used:

$$\Delta_x \varphi = \Delta x \cdot \frac{Z_x}{S_x}, \quad \Delta_y \varphi = \Delta y \cdot \frac{Z_y}{S_y} \quad (4)$$

where:

- $\Delta x, \Delta y$ is the horizontal and vertical distance between the frame centre and current object position (can be positive or negative).
- Z_x, Z_y are the angle of views of the used camera.
- S_x, S_y are the horizontal and vertical frame resolutions.

3.3 Kalman Filter

In order to filter the position of the tracked object and try to predict the object movement when it is covered, the Kalman filter was applied to the output of the tracking algorithm (object location coordinates). For this purpose, the following state-space model of object movement and measurements was assumed:

$$x(k+1) = Ax(k) + \nu(k) \quad (5)$$

$$y(k) = Cx(k) + \omega(k) \quad (6)$$

$$A = \begin{bmatrix} 1 & 0 & h & 0 & \frac{h^2}{2} & 0 \\ 0 & 1 & 0 & h & 0 & \frac{h^2}{2} \\ 0 & 0 & 1 & 0 & h & 0 \\ 0 & 0 & 0 & 1 & 0 & h \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (8)$$

where:

- $x(k)$ is the state vector containing: horizontal position, vertical position, horizontal velocity, vertical velocity, horizontal acceleration, vertical acceleration.
- h is the sampling period.
- $\nu(k) \sim \mathcal{N}(0, V)$ is the process noise.
- $\omega(k) \sim \mathcal{N}(0, W)$ is the observation noise.
- V and W are covariance matrices for respectively $\nu(k)$ and $\omega(k)$.

Equations (5) and (6) describe the uniformly accelerated motion in two directions (horizontal and vertical). The covariance matrices were chosen experimentally.

It was assumed that during object occlusion only the prediction phase of the Kalman filter will be conducted. Such approach was tested for the tracking by detection algorithm. The occlusion was detected using the number of object pixels visible in the current frame. If this value was below a preset threshold, the occlusion state was activated and the position of the tracked object was changed according to the used model.

3.4 Regulator

In the designed system the camera works as a sensor of the control error. In order to guarantee the correct positioning of the considered pan-tilt head, a regulator had to be implemented. Its output signals were used as servomotors setpoints.

Figure 3 shows a block diagram of designed control system where:

- $P(s)$ – Laplace transform of the position of the tracked object.

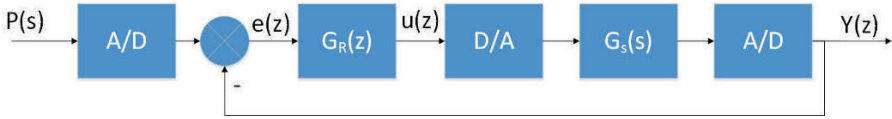


Fig. 3. Block diagram of designed control system.

- $e(z)$ – Z transform of the control error.
- $u(z)$ – Z transform of the regulator's output.
- $G_R(z)$ – transfer function of the used regulator.
- $G_S(s)$ – transfer function of the controlled object (servomechanism).
- D/A – digital to analogue converter (zero-order hold).
- A/D – analogue to digital converter.

In the current version of the system, it is impossible to directly measure the positions of servos. Thus, the used control algorithm had to be based on the output from the previous iteration. Therefore, it was decided to use the incremental PID (Proportional – Integral – Derivative) controller. It is described by the following equation:

$$u(k) - u(k-1) = P \cdot (e(k) - e(k-1)) + I \cdot e(k) + D \cdot (e(k) - 2e(k-1) + e(k-2)) \quad (9)$$

where: $u(k)$ – controller's output, $e(k)$ – control error, P, I, D – coefficients for the proportional, integral and derivative terms.

To be able to test the controller without the risk of damaging the pan-tilt head, it was decided to create a model of the used system. The analogue part of the system with D/A and D/A was converted to a digital state space model:

$$x(k+1) = A^+ x(k) + B^+ u(k) \quad (10)$$

$$y(k) = C^+ x(k) \quad (11)$$

$$A^+ = \begin{bmatrix} (1 + \frac{h}{2T})e^{-\frac{h}{2T}} & he^{-\frac{h}{2T}} \\ -\frac{h}{4T^2}e^{-\frac{h}{2T}} & (1 - \frac{h}{2T})e^{-\frac{h}{2T}} \end{bmatrix} \quad (12)$$

$$B^+ = \begin{bmatrix} -2T(h + 2T)e^{-\frac{h}{2T}} + 4T^2 \\ he^{-\frac{h}{2T}} \end{bmatrix} \quad (13)$$

$$C^+ = [\frac{1}{4T^2} \ 0] \quad (14)$$

where: h – sampling period, T – time constant of used servomechanisms.

Then, simulations were executed in *Matlab* and *Simulink* environments. Based on the model response, the following parameters were chosen:

$$P = 0.4, \quad I = 0.1, \quad D = 0.05 \quad (15)$$

Figure 4 shows the step response of the model (10), (11) with the described regulator for the following parameters of the model: $T = 0.05$, $h = 1/60$. The described regulator was implemented in the ARM processor core available in the Zynq device (c.f. Figure 1 – Regulator).

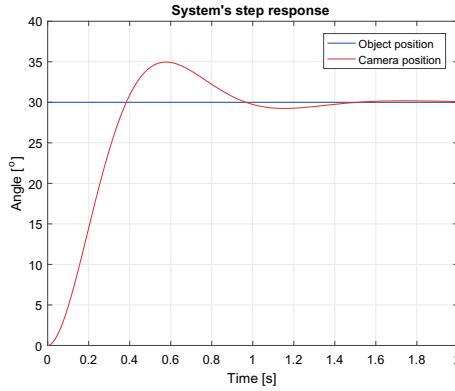


Fig. 4. Step response of system model with designed regulator.

3.5 Communication

To run the system it was necessary to establish communication between a PC, the Zynq device and the servomotor controller 'Maestro' (c.f. Figure 1). UART (Universal Asynchronous Receiver and Transmitter) communication protocol was used. Dedicated ARM core peripherals were utilized for this purpose. Moreover, a simple protocol was proposed. The command sent to the Zynq device from a PC was 5 bytes long. The first byte indicated the type of command and the following ones contained parameters. The system worked in two modes: manual and stand-alone. In the first one, the data received from the PC were forwarded to the 'Maestro' device. So, the head could be controlled directly from an UART terminal. In the second mode, for each received frame from the camera, a new setpoint was sent to the servos controller.

Support of the following commands was implemented:

- changing position of the servomechanisms,
- changing maximal speed of the servomechanisms,
- changing maximal torque of the servomechanisms,
- reading current setpoint from the controller,
- starting stand-alone operation mode (tracking),
- start of sending data to the PC (data acquisition mode).

Also communication between the reprogrammable logic (FPGA, PL) and the ARM processor (PS) was required. The *AXI-Lite* interface was used. When coordinates of the tracked object were available, an interrupt was triggered and the data read by the processor. Then the controller routine was executed and control for the servos provided.

3.6 Object Marking and Visualization

In the presented system both modules were used to analyse the correctness of the tracking algorithms. In the case of simple tracking by detection the centroid

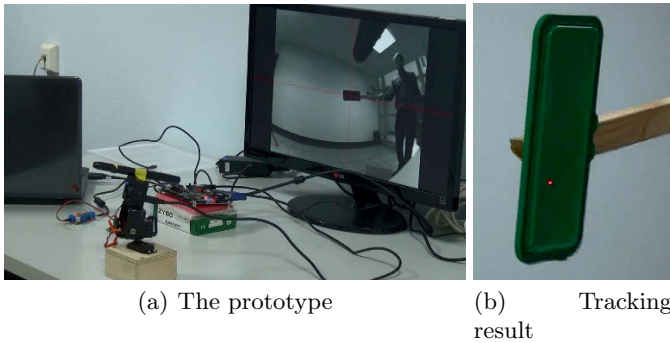


Fig. 5. Real-life evaluation of the prototype.

was visualized by two intersecting lines – cf. Figure 2(a). Whereas, for the mean-shift algorithm a bounding box was used – cf. Figure 2(b). The markers were overlaid on the input video stream, which was then passed to the VGA display controller.

4 System Evaluation

In Figure 5(a) a photograph of the designed prototype is presented. In the foreground, the mobile camera (with two servos) with a laser pointer on top is visible. Behind (right) the ZYBO board and servo controller (left) are present. On the monitor, the tracking by detection result for a green object is shown. Moreover, in Figure 5(b) a sample tracking result is displayed (the laser beam points at the object).

The proposed system worked correctly. Test for two tracking algorithms (by detection and mean-shift) were conducted. A 1280×720 @ 60 fps video stream was analysed in real-time. Additionally, the performance of the Kalman filter in case of object occlusion was evaluated. Incorrect behaviour was observed only in case of slowly occurring occlusion. They resulted from gradual object size shrinking and thus change of the centroid location. This issue will be addressed in the nearest future.

5 Summary

In the paper the concept of a Zynq SoC based moving smart camera was presented. The proposed system, consisting of a camera, two servos, servo controller and computing platform has been positively verified for two tracking algorithms. The obtained results reveal very good properties of heterogeneous platforms for this type of applications. Computationally complex algorithms can be implemented in the reprogrammable part (FPGA) to obtain real-time performance and energy efficiency. On the other hand, relatively simple computations like

the controller or Kalman filter, as well as communication with other components on the system can be implemented in the processor system.

As part of future work it is planned to: implement the KLT and other more advanced tracking algorithms, add servo position and speed sensors to improve the positioning, realize wireless communication with the unit, as well as perform an in-depth analysis of possible regulators.

Acknowledgements The work presented in this paper was supported by the AGH University of Science and Technology project no. 15.11.120.879.

References

1. S. Bernhardt. Mean-Shift Video Tracking. <https://www.mathworks.com/matlabcentral/fileexchange/35520-mean-shift-video-tracking>. Last access: 20.12.2016.
2. D. Comaniciu and V. Ramesh and P. Meer Real-time tracking of non-rigid objects using mean shift Proceedings IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 142–149 (2000)
3. D.D. Doyle, A.L. Jennings, J.T. Black Optical flow background estimation for real-time pan/tilt camera object tracking Measurement 48, pp. 195207 (2014)
4. Z. Du, H. Lu, H. Yuan, W. Zhang, C. Chen, K. Xie A FPGA Based High-Speed Binocular Active Vision System for Tracking Circle-Shaped Target Advances in Multimedia Information Processing – PCM 2015: 16th Pacific-Rim Conference on Multimedia, pp. 365–374 (2015)
5. H. Hagiwara, K. Asami, M. Komori FPGA Implementation of Image Processing for Real-Time Robot Vision System Convergence and Hybrid Information Technology: 5th International Conference pp. 134–141 (2011)
6. W. Jang, S. Oh and G. Kim A hardware implementation of pyramidal KLT feature tracker for driving assistance systems 12th International IEEE Conference on Intelligent Transportation Systems, pp. 1–6 (2009)
7. C. Lyu et al., High-speed target tracking base on FPGA IEEE International Conference on Real-time Computing and Robotics (RCAR), pp. 272-276 (2016)
8. K. Mazur, T. Kryjak An embedded vision-based tracking system for autonomous robot navigation Measurement Automation Monitoring vol. 5 pp. 172–174 (2016)
9. T. Meresiski Implementation of an object tracking algorithm in heterogeneous Zynq device Master Thesis, AGH University of Science and Technology, Krakow, 2016
10. Amir Mukhtar and Likun Xia Target Tracking Using Color Based Particle Filter 5th International Conference on Intelligent and Advanced Systems (ICIAS) (2014)
11. S. Paris. Particle Filter Color Tracker. <https://www.mathworks.com/matlabcentral/fileexchange/17960-particle-filter-color-tracker>. Last access: 20.12.2016.
12. C. Tomasi, T. Kanade Detection and Tracking of Point Features Computer Science Department, Carnegie Mellon University (1991)
13. A. Zawadzki and M. Gorgon Automatically controlled pantilt smart camera with {FPGA} based image analysis system dedicated to real-time tracking of a moving object Journal of Systems Architecture, vol. 61, number 10, pp. 681–692 (2015)

Prototype vision-based system for the supervision of the glass melting process: implementation for industrial environment

Paweł Rotter, Maciej Klemiato

AGH-University of Science and Technology,
Department of Automatics and Biomedical Engineering
rotter@agh.edu.pl, mkl@agh.edu.pl

Abstract. In the article we present an implementation of the system for automatic calculation of relevant descriptors of the glass melting process from the image of the glass surface. The purpose of the system is an automatic analysis of the process course in real time and providing information about batch amount in different zones of the furnace, the symmetry of batch distribution and the symmetry of temperature distribution. The system has been developed in Matlab and implemented with consideration of industrial requirements.

Keywords: Glass melting, image analysis, optical process control

1 Introduction

In the glass melting process, the image of the surface captured by a camera located inside the furnace, in the upper part of the furnace chamber, is an important source of information about the state of the process. The human operator controls the process mostly based on distribution of batch (raw material) floating on the surface of molten glass. Precise control of the process can increase the quality of glass, reduce pollution (NO_x emission) and reduce energy consumption, which is a big part of production costs [1]. An automatic system can be more precise than based on human operator. Computer vision is on common use in glass production at the stage of quality control [2] but there are few systems for the automatic control of the melting process [3], [4]. For example, none of three glassworks where the proposed system was tested (Warta Glass Jedlice SA, Can-Pack Orzesze and Stolzle Czestochowa Sp. z o.o.) was equipped with a system for the automatic analysis of images from the furnace. Moreover, solutions described in the related literature do not consider symmetry of batch distribution or symmetry of temperature of batch and molten glass, which is relevant to the control, according to melting experts.

In [5], [6] we proposed a method for the automatic image-based calculation of the process parameters, such as batch coverage in different zones of the furnace, indicators of batch distribution asymmetry and indicator of temperature asymmetry based

on emission of radiation in visible spectrum. In [7] we presented a method for elicitation of relevant melting criteria based on furnace camera image.

In this article we describe the industrial implementation of the system for optical supervision of the glass melting process, developed in cooperation with Techglass Ltd and Can-Pack Orzesze glassworks. We used Matlab as a rapid prototyping environment to build executable software for the target platform – an industrial PC with Matlab runtime installed, which communicates with SCADA and a furnace camera.

2 Algorithms

In **Fig. 4** (left part of the application window) we present an input image from the furnace camera, taken during reversal, i.e. the moment when one burner stops before the other starts. This is the moment when there is no flame in the furnace, so we can obtain a clear image.

The algorithms require some data provided by the user when the system is placed in a new furnace or camera position is changed. The user is required:

- to indicate several points that belong to batch area and to molten glass area, which are needed for segmentation of the image into three classes: batch, molten glass, and sediment on the camera lens (see example in **Fig. 4**).
- to identify the quadrangle of the glass surface. Its corners are used to extract parameters of the perspective transformation [8] and algorithms operate on the orthoview of this area, i.e. image is mapped to the glass surface’s coordinate system.

See [5] for the details of calibration and segmentation. The segmented image is then used for the calculation of parameters that describe batch distribution and symmetry. *Batch coverage coefficients* describe the percentage of the glass surface covered by batch blanket. They are calculated for three zones Z_1 - Z_3 presented in **Fig. 1**. *Batch symmetry coefficients*, calculated for the same zones, reflect the symmetry of batch distribution with respect to the tank’s symmetry axis. They are based on the batch asymmetry coefficient defined in [5] for area A as:

$$\Phi_A = -\frac{2}{S^2} \mathit{mean}_{y \in A} \int_{-S}^S xb(x, y)dx \tag{1}$$

where x and y are the axes of the glass surface coordinate system (see **Fig. 1**), S is half of the tank’s width and $b(x,y)$ is a binary variable equal to 1 where the batch blanket covers the glass surface at location (x,y) and 0 otherwise. $\Phi > 0$ means that the batch has shifted to the left side, $\Phi < 0$ means that it has shifted to the right side and $\Phi = 0$ corresponds to a symmetrical distribution of the batch blanket. Batch asymmetry coefficient is normalized to $[-1,1]$.

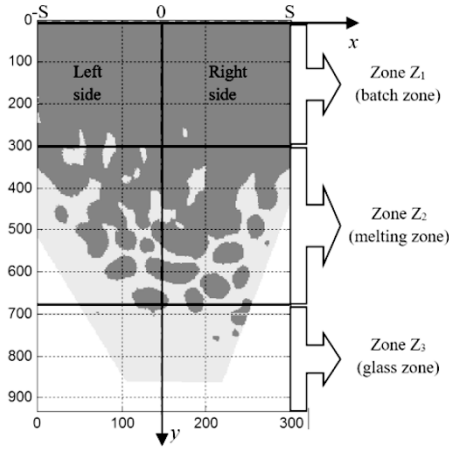


Fig. 1. The coordinate system related to the glass surface

3 The hardware and software considerations

The overall diagram of the glass production control system equipped with the proposed vision module is depicted in Fig. 2. The process – glass melting in the furnace chamber – is controlled by the PLC and SCADA. The vision system requires an additional PC with OPC client for communication with SCADA and camera support for communication with the furnace camera. It is assumed that there is no direct communication between PLC and the vision module for security reasons. This also allows the system to be independent of PLC brand and type.

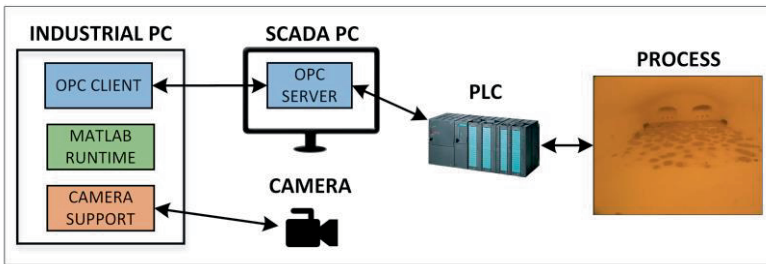


Fig. 2. Diagram of the glass production control system with the vision module.

We used the OPC server included in Wonderware InTouch SCADA system on one side and the OPC client from Matlab OPC Toolbox on the other side to establish communication between the production system and the vision module. The main reason for this connection is the need for information about the time of reversal. This is the only moment when we can acquire a clear furnace image without flame. Therefore, the reversal event triggers capture of the image.

For acquisition and analysis of the image we used *Image Acquisition Toolbox* and *Image Processing Toolbox* from Matlab environment. We also used a *support package* – an add-on that enables us to use Matlab with specific third-party hardware (camera). For general purposes digital USB cameras one should install *OS Generic Video Interface* but there are also several interfaces for more specialized cameras – digital and analog – supporting numerous video standards.

Batch coverage coefficients calculated in each reversal cycle are stored in *SQLite* database for trend plotting and analysis. For database manipulations we used the *Matlab Interface to SQLite* which lets us work with *SQLite* database files without installing and administering a database or driver.

The whole software – the vision algorithms, communication and the user interface – were developed in Matlab and compiled with Matlab Compiler into a standalone application. The application then can be deployed to the target computer with Matlab Runtime installed. This software development workflow is presented on **Fig. 3**. Diagram of software development workflow.

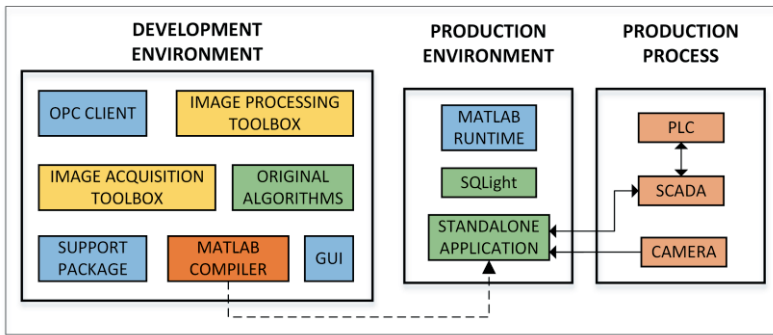


Fig. 3. Diagram of software development workflow.

4 Graphical user interface

The graphical user interface can be done programmatically, using appropriate graphical controls and writing callbacks or with drag-and-drop environment for laying out user interfaces: *GUIDE (GUI Development Environment)* or *App Designer*. The second way is easier but the first gives more control over the application.

The application main window consists of the three sub-windows: camera preview window (left side), batch coverage coefficients for different zones of the glass surface (as described in **Fig. 1**) and batch symmetry coefficient (right side), and the symmetry trend (upper part).

It is natural that the molten glass on the left side of the furnace is melted more when the left burner is working. And the opposite situation occurs when the right burner is working. This affects the value of symmetry coefficient. To eliminate this influence we use mean value of two consecutive symmetry coefficients for displaying the symmetry trend.

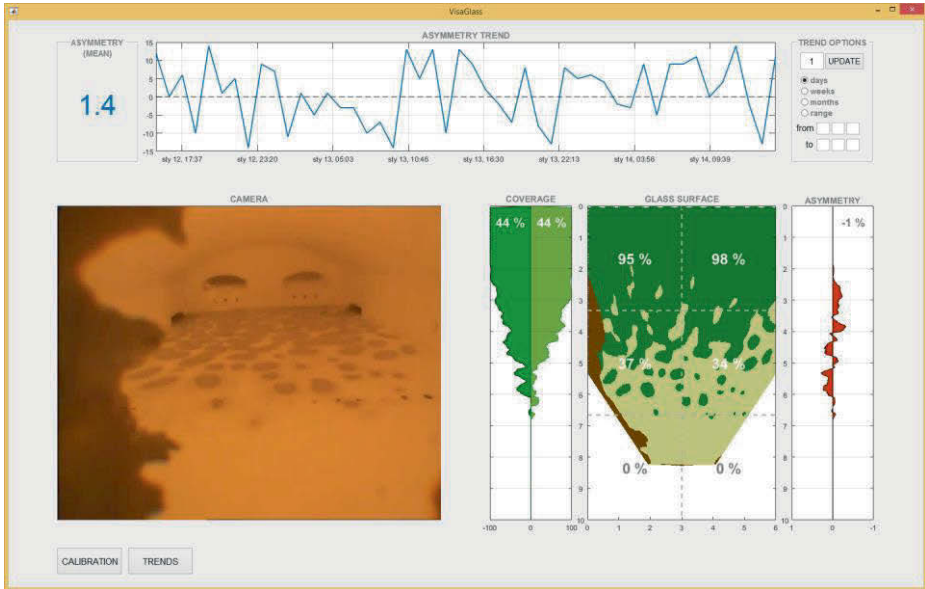


Fig. 4. Application main window

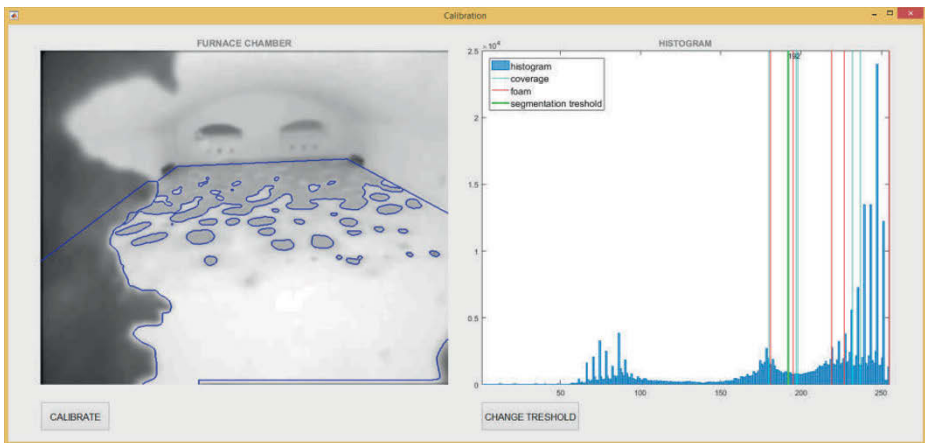


Fig. 5. Calibration sub-window

In the application main window we also display several indicators like:

- symmetry coefficient trend for vertical axis of the furnace (i.e. its width)
- percentage of the batch coverage for the left and right side of the furnace (which is the most important information for the process operator) and for six areas of the furnace (left/right and upper/middle/lower part)
- mean symmetry coefficient within defined time period

Before the first use of the application and after each camera replacing (e.g. after lens cleaning) there is a need of the image calibration to obtain the axes of the glass surface coordinate system. It should be done by the operator using the tool presented in **Fig. 5**.

5 Conclusions

We developed the industrial application for the supervision of the glass melting process using Matlab as a rapid prototyping environment. Using Matlab Compiler allowed us to substantially accelerate developing process and made it much easier.

The main purpose of the application is to help process operator in making decisions about the melting process control, especially setting the parameters of burners and chargers. It is meant as a supervision system, additional to SCADA, which could recommend operators adjusting relevant system parameters to obtain higher symmetry of the batch coverage.

For now the application works in open loop, independently of the process control system. It plays just advising role for operators. However, in the future it could work in closed loop as a part of the PLC control system if it proves its usefulness.

References

1. Ross, C.P., Tincher, G.L.: Glass melting technology: A technical and economic assessment. Glass Manufacturing Industry Council, U.S. Department of Energy Industrial Technologies Program (2004)
2. Nishu, Agrawal, S.: Glass defect detection techniques using digital image processing. A Review. IP Multimedia Communications 1, 65-67 (2011)
3. Siemens: SIGLAS® Optical melt control. (2006)
4. Muller, J., Chmelar, J., Bodi, R., Matustik, F., Viktorin, P.: Automatic batch position control by expert system ESIITM. 23rd International Congress on Glass (ICG), Prague (2013)
5. Rotter, P., Skowiniak, A.: Image-based analysis of the symmetry of the glass melting process. Glass Technology: European Journal of Glass Science and Technology Part A 54, 119-131 (2013)
6. Rotter, P., Skowiniak, A.: Projekt i prototyp systemu komputerowej analizy obrazu dla pieców szklarskich (Design and prototype of the computer vision system for glass furnaces). Pomiary, Automatyka, Kontrola 59, 684-687 (2013)
7. Rotter, P.: Extraction of relevant glass melting parameters based on the pairwise comparisons of sample images from a furnace. Glass Technology: European Journal of Glass Science and Technology Part A 55, 55-62 (2014)
8. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge University Press (2003)

The k-opt algorithm analysis. The flexible job shop case

Wojciech Bożejko¹, Mariusz Uchroński¹, and Mieczysław Wodecki²

¹ Institute of Computer Engineering, Control and Robotics
Wrocław University of Technology
Janiszewskiego 11-17, 50-372 Wrocław, Poland
wojciech.bozejko@pwr.edu.pl
mariusz.uchronski@pwr.edu.pl

² Institute of Computer Science, University of Wrocław
Joliot-Curie 15, 50-383 Wrocław, Poland
mieczyslaw.wodecki@uwr.edu.pl

Abstract. In the work there is considered an NP-hard flexible job shop problem. Its solution lies in allocation of operations to machines and determination of the sequence of their execution. There is also a method of construction of approximate algorithms presented, based on the idea of descent search, determining the allocation of operations. What is more, there were computational experiments conducted to investigate the correlation between the size of the neighborhood and the quality of solutions determined by the algorithm.

1 Introduction

The Job Shop scheduling problem (abbreviated to *JS*) can be summarized as follows. There is a set of tasks and a set of machines given. Each task is a sequence of operations. They should be executed in a sequence, without interruptions, on a machine adequate for each operation, at a specified time. At any time, the machine can perform at most one operation. The problem consists in appointment of a schedule (allocation of each operation in time interval to be executed on a suitable machine) minimizing the time of execution of all tasks. It belongs to a class of strongly *NP*-hard problems and is considered to be one of the most difficult combinatorial optimization problems. For many years, there has been conducted a study on different methods of construction of many, mainly approximate, algorithms. One of the most effective algorithms is tabu search described by Nowicki and Smutnicki [9]. In practice, only simple production systems can be modeled as *JS* problem. This fact does not meet the contemporary requirements of practitioners. Therefore, in the literature there have been many extensions to this problem proposed. One of them is flexibility (parallel machines consideration). In such a model machines, having the same functional properties (but perhaps with different capacities), are grouped in slots (the so-called Flexible Job Shop scheduling problem, in short denoted by *FJS*). In this

case, each operation must be performed on exactly one machine from corresponding to it slot. If we assume that each slot has only one machine, then we obtain the initial JS problem. Based on the above remark, in the literature there have been frequently used two-level structures of algorithms solving the FJS problem:

- Step 1: designation of assignment of operations to machines,
- Step 2: determination of order of operations on each machine.

Both steps are run several times while performance of Step 2 comes down to solving JS problem. This approach have been used, among others, by Brandimarte [5] and Bożejko et al. [2]. Another approach relies in not separating the computation process into two stages. Such integrated algorithms were presented by: Mastrolilli and Gambardella [8], Gao et al. [6], Hmida et al. [7] and Said and Fattahi [10].

In this paper there is presented a method for the construction of approximate algorithms for determining the allocation of operations to machines. They are based on the conception taken from the descent search method. There are allocations generated, in which a change of a machine for a fixed number of operations (algorithm parameter) does not improve the optimized criterion. These are the so called k -optimal allocations, in which k is a parameter of the algorithm.

2 Problem definition

Considered in this work flexible job shop problem can be defined as follows. There is a set of tasks given $\mathcal{J} = \{J_1, J_2, \dots, J_n\}$, which should be executed machines from the set $\mathcal{M} = \{1, 2, \dots, \eta\}$. Machines of the same type are grouped into slots, i.e. subsets of machines with the same functional properties. A task is a sequence of certain operations constituting the so-called technological line. Each operation should be performed without interruption on exactly one machine from the appropriate slot in accordance with the technological line. The problem is not only to allocate operations to machines but also to determine the order of operations for each the machine so as to minimize execution time of all tasks (C_{\max}). Exactly this problem is described thoroughly, among others, in the works [3] and [8].

A task $J_i \in \mathcal{J}$ is a sequence of n_i operation

$$J_i = [o_i^1, o_i^2, \dots, o_i^{n_i}],$$

which in this order will be performed in this order on the respective machines (technological line). Let $\mathcal{O} = \{1, 2, \dots, o\}$ be the set of all operations. The set of machines \mathcal{M} can be broken into m disjoint subsets (slots) $\mathcal{M}^1, \mathcal{M}^2, \dots, \mathcal{M}^m$ ($\mathcal{M} = \bigcup_{i=1}^m \mathcal{M}^i$), wherein $|\mathcal{M}^i| = m_i$, $i = 1, 2, \dots, m$. Operation $v \in \mathcal{O}$ will be performed in the slot \mathcal{M}^{μ_v} (i.e. on one machine from this slot) in time $p_{v,j}$, $j \in \mathcal{M}^{\mu_v}$. The sequence

$$q = (q_1, q_2, \dots, q_o), \tag{1}$$

where $q_i \in \{1, 2, \dots, m_{\mu(i)}\}$, $i \in \mathcal{O}$, represents *allocation* of operations to machines. More specifically, q_i is the number of the machine in the slot $\mu(i)$ to

which there is assigned operation i . By \mathcal{Q} we denote the set of all such sequences (allocations). For a fixed allocation of operations to machines $q \in \mathcal{Q}$ considered in this work flexible job shop problem (FJS) boils down to job shop problem (in short denoted by $FJ(q)$). For this problem, let $\mathcal{O}^k = \{v \in \mathcal{O} : \mu_v = k\}$ ($k \in \mathcal{M}$) be a set of operations executed on k -th machine, π^k some permutation of elements from \mathcal{O}^k (order of operations' execution). By Φ^k we denote the set of all such permutations. Therefore, a sequence of permutation

$$\pi = (\pi^1, \pi^2, \dots, \pi^q),$$

where $\pi^i \in \Phi^i$, $i = 1, 2, \dots, \eta$ is a solution to $FJ(q)$ problem. Let Φ be the set of all such sequences. Using the above definitions and introduced designations, any solution to FJS problem will be represented by a pair (q, π) , where $q \in \mathcal{Q}$ is the assignment of operations to machines, and $\pi \in \Phi$ the order of operations' execution on individual machines.

For allocation $q \in \mathcal{Q}$, by $\mathcal{N}^k(q)$, $k = 1, 2, \dots, o$ we denote k -th neighborhood ($\mathcal{N}^k(q) \subseteq \mathcal{Q}$). The allocation $q' \in \mathcal{Q}$ belongs to neighborhood $\mathcal{N}^k(q)$ if and only if the sequences q and q' differ on exactly k positions. It means, that allocation q' can be obtained from q by changing to other machines (of course, from the respective slots) from allocation of k operation.

Below there is presented an algorithm (based on the descent search method) determining a suboptimal allocation of tasks to machines through search of k -th neighborhoods. By $C_{\max}(FJ(q))$ we denote solution to $FJ(q)$ problem, i.e. the minimum time of tasks execution in the flow shop problem, for a fixed allocation $q \in \mathcal{Q}$.

Algorytm k -opt(q)

```

 $C^\circ \leftarrow C^* \leftarrow C_{\max}(JS(q));$ 
 $C \leftarrow \infty;$ 
 $q^* \leftarrow q;$ 
while  $C^\circ < C_{\max}(JS(q))$  do
  begin
     $C \leftarrow C^\circ;$ 
    generate neighborhood  $\mathcal{N}^k(q);$ 
    designate allocation  $q^\circ \in \mathcal{N}^k(q)$  where
       $C^\circ = C_{\max}(JS(q^\circ)) = \min\{C_{\max}(JS(q')) : q' \in \mathcal{N}^k(q)\};$ 
    if  $C^\circ < C^*$  then
      begin
         $C^* \leftarrow C^\circ;$ 
         $q^* \leftarrow q^\circ;$ 
      end{begin}
       $q \leftarrow q^\circ$ 
    end{while}
  
```

The number of iterations of the algorithm is not polynomial in terms of the number of n tasks or q machines. On the basis of numerical experiments it is

possible to conclude that the algorithm executes an average n iterations (more precisely, this is the number of generated neighborhoods). As a criterion we take minimizing time of all tasks execution C_{\max} , but in the algorithm we can use any other criterion.

Remark 1. If q^* is the allocation of tasks to machines designated by algorithm **k-opt**(q), then the shift of any k operation to other machines (in the appropriate slots) does not generate an allocation of value smaller than execution time of all tasks $C_{\max}(JS(q^*))$.

Solution q^* designated by an algorithm **k-opt**(q) will be called *k-optimal*.

3 Computational experiments

The algorithm **k-opt**(q) has been implemented in C++ and run on the computing node of cluster BEM, which is equipped with multi-core processors Intel Xeon E5-2670 (2.30 GHz) operating under control of the operating system Scientific Linux 6.8 (Carbon). The experiments were carried out not only to compare the results of algorithms **k-opt**(q) ($k = 1, 2, 3$) with current best-known ones but also to examine the effect of the parameter k on the values of designated solutions. To speed up computations for each test instance algorithms **1-opt**(q), **2-opt**(q) and **3-opt**(q) were run independently through the use of threads in C++. In each of these algorithms, to determine the length of lineup of $C_{\max}(JS(q))$, there should be a job shop problem solved. Because it is NP-hard, there is approximate TSAB algorithm used presented in the work [9]. The computations were executed on 31 commonly known examples of test data with different sizes and degree of difficulty. They are divided into two groups:

- (a) ten working examples from the work of Brandimarte [5],
- (b) twenty-one examples provided by Barnes and Chambers [1]

The initial allocation of the operations to machines is generated with the use of the method searching for the global minimum in the table of execution time of operations on the machines. This method has been described in the work [1]. The results of computational algorithms are shown in Tables 1-4. The individual columns present:

- *Best* – currently the best known values of the objective function,
- *NTS* – the results of approximate Neuro-Tabu Search algorithm [4] solving *FJS* problem,
- **k-opt** – the results of the algorithm, in which at the same time there is a re-allocation of k operations to machines,
- *PRD* – the relative error in reference to the best known solution.

Tables 1 and 2 show the values of solution (length of lineup C_{\max}) designated by each algorithm.

Tablica 1: Makespan C_{\max} for instances of the (a) group.

problem	$n \times m$	C_{\max}				
		<i>Best</i>	<i>NTS</i>	1-opt	2-opt	3-opt
Mk01	10 × 6	40	70	43	42	40
Mk02	10 × 6	26	45	42	28	27
Mk03	15 × 8	204	330	321	204	204
Mk04	15 × 8	60	188	174	76	65
Mk05	15 × 4	172	209	202	174	174
Mk06	10 × 15	57	100	96	70	79
Mk07	20 × 5	139	217	202	151	150
Mk08	20 × 10	523	587	578	523	523
Mk09	20 × 10	307	454	355	309	331
Mk10	20 × 15	197	398	388	244	242

Tablica 2: Makespan C_{\max} for instances of the (b) group.

problem	$n \times m$	C_{\max}				
		<i>Best</i>	<i>NTS</i>	1-opt	2-opt	3-opt
mt10c1	10 × 11	927	930	927	927	927
mt10cc	10 × 12	908	930	914	910	913
mt10x	10 × 11	918	930	929	922	922
mt10xx	10 × 12	918	930	922	918	918
mt10xxx	10 × 13	918	930	922	918	918
mt10xy	10 × 12	905	930	914	907	907
mt10xyz	10 × 13	847	930	855	853	853
setb4c9	15 × 11	914	944	914	914	914
setb4cc	15 × 12	907	944	909	907	907
setb4x	15 × 11	925	944	925	925	925
setb4xx	15 × 12	925	944	930	925	925
setb4xxx	15 × 13	925	944	930	925	925
setb4xy	15 × 12	910	944	916	910	910
setb4xyz	15 × 13	903	944	908	903	908
seti5c12	15 × 16	1170	1226	1177	1174	1174
seti5cc	15 × 17	1136	1226	1136	1136	1136
seti5x	15 × 16	1198	1226	1199	1199	1199
seti5xx	15 × 17	1197	1226	1210	1197	1197
seti5xxx	15 × 18	1197	1226	1210	1197	1197
seti5xy	15 × 17	1136	1226	1136	1136	1136
seti5xyz	15 × 18	1125	1226	1128	1127	1127

By far the worst was the *NTS* algorithm. Most of determined by the algorithm solution values were much worse than the upper bounds (column *Best*). The **1-opt** algorithm appeared to be much better (for 6 instances designated solutions were equal to the values of *Best*). In case of two other two algorithms

2-opt and **3-opt** the number of designated best solutions was the same and equal to 15 (although the examples were different). It should be noted that in two cases (Mk09 and setb4xyz) algorithm **2-opt** finds a better solution than **3-opt**. Table 3 and 4 show relative errors in eference to the best solutions. For examples from group (a), Table 3, the average errors in algorithms are only slightly different from one another. The errors of two other algorithms (*NTS* and **1-opt**) are several times bigger. The results presented in Table 4 indicate significant differences in the results. The algorithm **2-opt** turns out to be far better than **3-opt** (average errors are respectively 0.10 and 0.15). This result is quite surprising because it seemed that the algorithm **3-opt** will be the best out of the tested ones. It should be emphasized that, for this data group the mean error of the *NTS* algorithm is 3.71, and is more than 37 times higher than the best error of algorithm **2-opt**. Algorithms **k-opt**, $k > 3$ run much longer and the average relative errors of the solutions designated by them are greater than the best ones listed in Tables 3 and 4.

Tablica 3: Relative deviation for instances of the (a) group.

problem	$n \times m$	PRD			
		<i>NTS</i>	1-opt	2-opt	3-opt
Mk01	10×6	75.00	7.50	5.00	0.00
Mk02	10×6	73.08	61.54	7.69	3.85
Mk03	15×8	61.76	57.35	0.00	0.00
Mk04	15×8	213.33	190.00	26.67	8.33
Mk05	15×4	21.51	17.44	1.16	1.16
Mk06	10×15	75.44	68.42	22.81	38.60
Mk07	20×5	56.12	45.32	8.63	7.91
Mk08	20×10	12.24	10.52	0.00	0.00
Mk09	20×10	47.88	15.64	0.65	7.82
Mk10	20×15	102.03	96.95	23.86	22.84
Average		73.84	57.07	9.65	9.05

Tablica 4: Relative deviation for instances of the (b) group.

problem	$n \times m$	PRD			
		<i>NTS</i>	1-opt	2-opt	3-opt
mt10c1	10 × 11	0.32	0.00	0.00	0.00
mt10cc	10 × 12	2.42	0.66	0.22	0.55
mt10x	10 × 11	1.31	1.20	0.44	0.44
mt10xx	10 × 12	1.31	0.44	0.00	0.00
mt10xxx	10 × 13	1.31	0.44	0.00	0.00
mt10xy	10 × 12	2.76	0.99	0.22	0.22
mt10xyz	10 × 13	9.80	0.94	0.71	0.71
setb4c9	15 × 11	3.28	0.00	0.00	0.00
setb4cc	15 × 12	4.08	0.22	0.00	0.00
setb4x	15 × 11	2.05	0.00	0.00	0.00
setb4xx	15 × 12	2.05	0.54	0.00	0.00
setb4xxx	15 × 13	2.05	0.54	0.00	0.00
setb4xy	15 × 12	3.74	0.66	0.00	0.00
setb4xyz	15 × 13	4.54	0.55	0.00	0.55
seti5c12	15 × 16	4.79	0.60	0.34	0.34
seti5cc	15 × 17	7.92	0.00	0.00	0.00
seti5x	15 × 16	2.34	0.08	0.08	0.08
seti5xx	15 × 17	2.42	1.09	0.00	0.00
seti5xxx	15 × 18	2.42	1.09	0.00	0.00
seti5xy	15 × 17	7.92	0.00	0.00	0.00
seti5xyz	15 × 18	8.98	0.27	0.18	0.18
Average		3.71	0.49	0.10	0.15

4 Summary

In the work there is presented a new method of algorithms construction determining the allocation of operations to machines for flexible job shop problem. As optimized criterion there was the time of completion of all tasks adopted, i.e. C_{\max} . Computational experiments were performed on well-known in the literature examples. Determined by the algorithm **2-opt** solutions are only slightly worse (an average less than 1 %) than the best currently known values.

Literatura

1. Barnes J., Chambers J., Flexible job shop scheduling by tabu search, Graduate program in operations research and industrial engineering. The University of Texas at Austin, 1996.
2. Bożejko W., Uchroński M., Wodecki M., Parallel hybrid metaheuristics for the flexible job shop problem, *Computers & Industrial Engineering* 59 (2010) 323–333.
3. Bożejko W., Uchroński M., Wodecki M., The new golf neighborhood for the flexible job shop problem, *Proceedings of the ICCS 2010, Procedia Computer Science* 1 (2010), Elsevier, 289–296.

4. Bożejko W., Uchroński M., Wodecki M., Parallel neuro-tabu search algorithm for the job shop scheduling problem, Proceedings of ICAISC 2013, Lecture Notes in Artificial Intelligence No. 7895, Springer (2013), 489–499.
5. Brandimarte P., Routing and scheduling in flexible job shop by tabu search, *Annals of Operations Research*, 41, 1993, 157–183.
6. Gao J., Sun L., Gen M., A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problem, *Computers and Operations Research*, 35, 2008, 2892–2907.
7. Hmida A., Haouari M., Huguet M., Lopez P., Discrepancy search for the flexible job shop scheduling problem, *Computers and Operations Research*, 37, 2010, 2192–2201.
8. Mastrolilli M., Gambardella L., Effective neighborhood functions for the flexible job shop problem, *Journal of scheduling*, 3(1), 2000, 3–20.
9. Nowicki, E., Smutnicki, C., (1996): A fast taboo search algorithm for the job shop problem, *Management Science* 42, 797–813.
10. Saidi-Mehrabad M., Fattahi P., Flexible job shop scheduling with tabu search algorithm, *Int. J. Adv. Manuf. Technol.*, 32, 2007, 563–570.

Implementation of Analytical Generalized Predictive Controller for Very Fast Applications Using Microcontrollers: Preliminary Results

Patryk Chaber, Maciej Ławryńczuk

Institute of Control and Computation Engineering, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland, tel. +48 22 234-71-24
pjchaber@gmail.com, M.Lawrynczuk@ia.pw.edu.pl

Abstract. This paper describes implementation of the Generalized Predictive Control (GPC) algorithm on an STM32 microcontroller with the ARM Cortex M7 core. The algorithm is implemented in its analytical (explicit) version which requires computationally simple matrix and vector operations in real time, no on-line optimisation is necessary. As a result, the algorithm may be used for controlling very fast dynamic processes characterised by sampling periods of millisecond order. Results of real experiments are demonstrated for two example processes.

Key words: Generalized Predictive Control, Model Predictive Control, microcontroller, software implementation.

1 Introduction

In Model Predictive Control (MPC) algorithms [1, 12] the control policy is repeatedly calculated on-line from an optimisation problem in which some future differences between the set-point and the predicted process trajectory are minimised. As a result, unlike the classical Proportional-Integral-Derivative (PID) controller, the MPC algorithms are able to control effectively dynamic processes with many manipulated and controlled variables as well as with difficult dynamic properties, e.g. with delays. Furthermore, in MPC it is possible to take into account constraints imposed on process variables which result from physical limits of actuators or some technological requirements.

The MPC algorithms have been used in practice for some 40 years, mainly in large-scale industrial applications (e.g. in chemical engineering, food processing, paper industry [10]). Typically, sampling periods in such applications is of the order of seconds or minutes, which means that for implementation the classical industrial hardware and software platforms are used: Distributed Control Systems (DCS) and Programmable Logic Controllers (PLC). In addition to the aforementioned applications, currently the MPC algorithms become more and more popular in embedded applications. A characteristic feature of such applications is the fact that samplings periods are very short, usually of millisecond order. Applications of MPC in fast embedded systems are possible due to

huge progress in microelectronics (the currently available microcontrollers are very fast, computationally efficient and cheap). It is an interesting phenomenon that MPC algorithms implemented on microcontrollers are used not only in fast embedded system [7, 9, 11], for which they have been developed in mind, but also in typical industrial applications [6]. Recently, a promising field of research concerned with fast development of MPC algorithms for embedded systems is automatic code generation for microcontrollers [2, 3, 5, 8].

Having fast applications in mind, it is necessary to develop MPC algorithms in which the time necessary to calculate on-line the value(s) of the manipulated variable(s) is as short as possible. The calculation time may be reduced when the MPC algorithm is implemented in its analytical (explicit) version. In such an approach all the existing constraints are removed from the on-line optimisation problem. Provided that the considered process is characterised by a linear dynamic model, the optimal control policy may be calculated analytically, without the necessity of on-line optimisation.

This work describes practical implementation of the analytical version of the Generalized Predictive Control (GPC) MPC algorithm [4]. An STM32 microcontroller with the ARM Cortex M7 core running at 216 MHz is used as the hardware platform. The algorithm is applied to two example processes. The influence of the prediction horizon on on-line calculation time is assessed.

2 Generalized Predictive Control Algorithm

2.1 The objective of GPC

This work is concerned with multivariable processes with n_u inputs (manipulated variables) u_1, \dots, u_{n_u} and n_y outputs (controlled variables) y_1, \dots, y_{n_y} . In presentation the vectors $u = [u_1 \dots u_{n_u}]^T$ and $y = [y_1 \dots y_{n_y}]^T$ are used.

The objective of MPC [1, 12] is to calculate on-line in real-time not only the values of the manipulated variables for the current sampling instant k , $k = 0, 1, 2, \dots$, as it is done in the classical control algorithms, e.g in the case of the PID, but a future control policy for a control horizon, N_u . Usually, the future increments of the manipulated variables are successively found

$$\Delta \mathbf{u}(k) = \begin{bmatrix} \Delta u(k|k) \\ \vdots \\ \Delta u(k + N_u - 1|k) \end{bmatrix} \quad (1)$$

The increments are defined as $\Delta u(k|k) = u(k|k) - u(k - 1)$, $\Delta u(k + p|k) = u(k + p|k) - u(k + p - 1|k)$ for $p = 1, \dots, N_u - 1$. It is assumed that $\Delta u(k + p|k) = 0$ for $p \geq N_u$. The future increments of the manipulated variables defined by Eq. (1) are calculated as a result of optimisation of the predicted control errors i.e. the deviations between the predicted values of the process output variables, $\hat{y}(k + p|k)$, and their set-points, $y^{sp}(k + p|k)$. Predicted process behaviour is considered over a prediction horizon, N , i.e. for $p = 1, \dots, N$. Additionally,

unwanted excessive increments of the manipulated variables are penalised during calculations. Hence, the minimised MPC cost-function is usually

$$J(k) = \sum_{p=1}^N \left\| y^{\text{sp}}(k+p|k) - \hat{y}(k+p|k) \right\|_{\mathbf{M}_p}^2 + \sum_{p=0}^{N_u-1} \left\| \Delta u(k+p|k) \right\|_{\mathbf{A}_p}^2 \quad (2)$$

where $\mathbf{M}_p \geq 0$ and $\mathbf{A}_p > 0$ are tuning matrices of dimensionality $n_y \times n_y$ and $n_u \times n_u$, respectively. Although the whole optimal future control policy (1) over the control horizon is calculated at the sampling instant k , only its first n_u elements are actually applied to the process, i.e. $u(k) = \Delta u(k|k) + u(k-1)$. At the next sampling instant, $k+1$, output measurements are updated, the prediction is shifted one step forward and the whole procedure is repeated.

2.2 Modelling and Prediction

In the discussed GPC algorithm the following ARX-style (AutoRegressive with eXogenous input) model is used

$$\mathbf{A}(q^{-1})y(k) = \mathbf{B}(q^{-1})u(k) \quad (3)$$

where the entries of the matrices

$$\mathbf{A}(q^{-1}) = \begin{bmatrix} A_{1,1}(q^{-1}) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & A_{n_y, n_y}(q^{-1}) \end{bmatrix} \quad (4)$$

and

$$\mathbf{B}(q^{-1}) = \begin{bmatrix} B_{1,1}(q^{-1}) & \dots & B_{1, n_u}(q^{-1}) \\ \vdots & \ddots & \vdots \\ B_{n_y, 1}(q^{-1}) & \dots & B_{n_y, n_u}(q^{-1}) \end{bmatrix} \quad (5)$$

are the following polynomials in the backward shift operator q^{-1}

$$A_{m,m}(q^{-1}) = 1 + a_1^m q^{-1} + \dots + a_{n_A}^m q^{-n_A} \quad (6)$$

for $m = 1, \dots, n_y$ and

$$B_{m,n}(q^{-1}) = b_1^{m,n} q^{-1} + \dots + b_{n_B}^{m,n} q^{-n_B} \quad (7)$$

for $m = 1, \dots, n_y$, $n = 1, \dots, n_u$. The order of model dynamics is defined by the integer numbers n_A and n_B , model parameters are denoted by real number coefficients a_i^m and $b_i^{m,n}$. From Eqs. (3), (4), (5), (6) and (7), the consecutive outputs of the model are

$$\begin{aligned} y_1(k) &= \sum_{n=1}^{n_u} \sum_{i=1}^{n_B} b_i^{1,n} u_n(k-i) - \sum_{i=1}^{n_A} a_i^1 y_1(k-i) \\ &\vdots \\ y_{n_y}(k) &= \sum_{n=1}^{n_u} \sum_{i=1}^{n_B} b_i^{n_y, n} u_n(k-i) - \sum_{i=1}^{n_A} a_i^{n_y} y_{n_y}(k-i) \end{aligned} \quad (8)$$

The predictions $\hat{y}(k+p|k)$ of the output variable(s) over the prediction horizon, i.e. for $p = 1, \dots, N$, are calculated from the dynamic model of the process defined by Eqs. (8). In this work the GPC algorithm with the so called Dynamic Matrix Control (DMC) disturbance model is used, which makes it possible to eliminate the necessity of solving Diophantine equations [12]. The predicted value of the m^{th} process output variable for the future sampling instant $k+p$ calculated at the current instant k is calculated from the general prediction equation

$$\hat{y}_m(k+p|k) = y_m(k+p|k) + d_m(k) \tag{9}$$

where the quantity $y_m(k+p|k)$ is the model output whereas $d(k)$ is an estimation of the unmeasured disturbance acting on the process output. It is assumed that disturbance is constant over the whole prediction horizon and its value is estimated as the difference between the real value of the process output measured at the sampling instant k and its value calculated from the model (8). It may be proved that the output predictions may be compactly expressed as [12]

$$\hat{\mathbf{y}}(k) = \underbrace{\mathbf{G}(k)\Delta\mathbf{u}(k)}_{\text{future}} + \underbrace{\mathbf{y}^0(k)}_{\text{past}} \tag{10}$$

where the vectors

$$\hat{\mathbf{y}}(k) = \begin{bmatrix} \hat{y}(k+1|k) \\ \vdots \\ \hat{y}(k+N|k) \end{bmatrix}, \quad \mathbf{y}^0(k) = \begin{bmatrix} y^0(k+1|k) \\ \vdots \\ y^0(k+N|k) \end{bmatrix} \tag{11}$$

are of length $n_y N$ and the matrix

$$\mathbf{G} = \begin{bmatrix} \mathbf{S}_1 & \mathbf{0}_{n_y \times n_u} & \dots & \mathbf{0}_{n_y \times n_u} \\ \mathbf{S}_2 & \mathbf{S}_1 & \dots & \mathbf{0}_{n_y \times n_u} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_N & \mathbf{S}_{N-1} & \dots & \mathbf{S}_{N-N_u+1} \end{bmatrix} \tag{12}$$

is of dimensionality $n_y N \times n_u N_u$. The sub-matrices

$$\mathbf{S}_p = \begin{bmatrix} s_p^{1,1} & \dots & s_p^{1,n_u} \\ \vdots & \ddots & \vdots \\ s_p^{n_y,1} & \dots & s_p^{n_y,n_u} \end{bmatrix} \tag{13}$$

consist of step-response coefficients of the ARX-type model (8). The so called free-response vector $\mathbf{y}^0(k)$ defines influence of the past values of the process variables on the future output predictions.

2.3 On-Line Optimisation of Control Policy

Using the GPC prediction equation (10), the MPC cost function (2) can be expressed in a compact form

$$\begin{aligned} J(k) &= \|\mathbf{y}^{\text{sp}}(k) - \hat{\mathbf{y}}(k)\|_M^2 + \|\Delta\mathbf{u}(k)\|_A^2 \\ &= \|\mathbf{y}^{\text{sp}}(k) - \mathbf{G}(k)\Delta\mathbf{u}(k) - \mathbf{y}^0(k)\|_M^2 + \|\Delta\mathbf{u}(k)\|_A^2 \end{aligned} \tag{14}$$

where the set-point and free trajectories

$$\mathbf{y}^{\text{SP}}(k) = \begin{bmatrix} y^{\text{SP}}(k+1|k) \\ \vdots \\ y^{\text{SP}}(k+N|k) \end{bmatrix}, \quad \mathbf{y}^0(k) = \begin{bmatrix} y^0(k+1|k) \\ \vdots \\ y^0(k+N|k) \end{bmatrix} \quad (15)$$

are vectors of length $n_y N$ and the weighting matrices $\mathbf{M} = \text{diag}(\mathbf{M}_1, \dots, \mathbf{M}_N)$ and $\mathbf{\Lambda} = \text{diag}(\mathbf{\Lambda}_0, \dots, \mathbf{\Lambda}_{N_u})$ are of dimensionality $n_y N \times n_y N$ and $n_u N_u \times n_u N_u$, respectively. In practice the set-point trajectory is constant over the prediction horizon, i.e. $y^{\text{SP}}(k+p|k) = y^{\text{SP}}(k)$.

In order to calculate the optimal values of the decision variables, i.e. the values of the increments of the manipulated variables $\Delta \mathbf{u}(k)$, it is sufficient to equate the vector of gradients of the minimised cost-function (14) to a zero-vector of length $n_u N_u$. The optimal future control moves are

$$\Delta \mathbf{u}(k) = \mathbf{K}(\mathbf{y}^{\text{SP}}(k) - \mathbf{y}^0(k)) \quad (16)$$

where

$$\mathbf{K} = (\mathbf{G}^T \mathbf{M} \mathbf{G} + \mathbf{\Lambda})^{-1} \mathbf{G}^T \mathbf{M} \quad (17)$$

is a matrix of dimensionality $n_u N_u \times n_y N$. Bearing in mind that at the current sampling instant k only the first n_u elements of the vector $\Delta \mathbf{u}(k)$ are actually applied to the process, they are only calculated from

$$\Delta u(k|k) = \mathbf{K}_{n_u}(\mathbf{y}^{\text{SP}}(k) - \mathbf{y}^0(k)) \quad (18)$$

where the matrix \mathbf{K}_{n_u} contains the first n_u rows of the matrix \mathbf{K} .

Because in practice it is usually necessary to impose some constraints on the magnitude and the rate of change of the calculated manipulated variables, the obtained optimal increments (18) may be projected onto the admissible set of constraints [12].

3 Implementation of GPC Algorithm Using Microcontrollers

Fig. 1 shows general connections between the controlled process and the microcontroller on which the GPC algorithm is implemented. Because the process variables are analog-type, measurements of the process output variables are converted into their discrete representations using Analog to Digital Converters (ADC). Similarly, in order to generate analog-type values of the manipulated variables which are calculated by the microcontroller, Digital to Analog Converters (DAC) are used. The PC computer is connected to the microcontroller using the serial Universal Asynchronous Receiver and Transmitter (UART) communication protocol. The PC computer is only used for data acquisition, i.e. to record the data. The analytical GPC algorithm is implemented in C programming language.

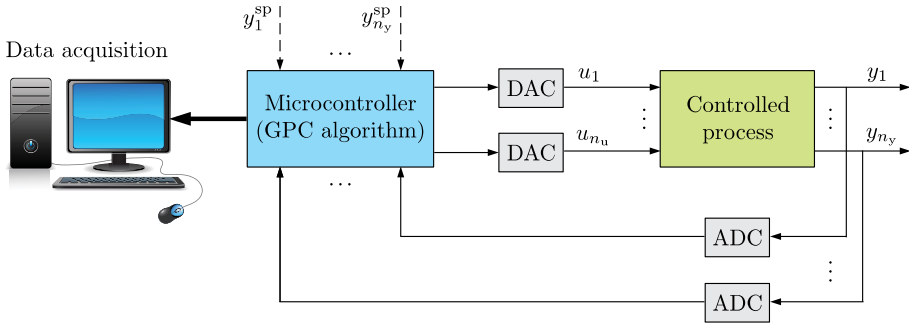


Fig. 1. Connections between the controlled process and the microcontroller with implementation of the GPC algorithm; DAC – Digital to Analog Converter, ACD – Analog to Digital Converter

4 Results of Experiments

The STM32F746 microcontroller (produced by STMicroelectronics) with the Cortex M7 core running at 216 MHz is used. Unlike many microcontrollers currently available on the market, the considered one co-operates with the Floating Point Unit (FPU), which is necessary for implementation of advanced computational algorithms. In order to minimise cost and speed up the prototyping process of the designed GPC algorithm, the development board STM32F746G-DISCO is used. The system offers a rich set of peripherals, including: 64 Mb SDRAM, 128 Mb Flash QSPI, Serial ports (UART, I2C), Ethernet connection and on-board programmer-debugger ST-Link/V2-1. It is also equipped with a 4.3" LCD touch panel, which may be used for communication with the user. A few 12-bit ADC converters are available and used in this project, but on-board DAC converters may be not freely used (they are reserved for other resources) and external ones are used. Fig. 2 shows the development board running the GPC algorithm. As far as the controlled process is concerned, in this work it is emulated on the second development board of the same type. For emulation purposes the input signals are connected by ADC converters and the output signals are converted by DAC ones.

In the first experiment carried out it is assumed that the process has one manipulated and one controlled variable, i.e. $n_u = n_y = 1$. The simulated process and its model are discrete-time representations of the continuous-time dynamic system

$$Y(s) = \frac{1}{0.1s^2 + 0.7s + 1}U(s) \tag{19}$$

The sampling period of the GPC algorithm is 50 ms whereas the process is emulated with the sampling period 5 ms. The tuning coefficients in the minimised cost-function (2) are $M_p = 1$ for $p = 1, \dots, N$ and $A_p = 1$ for $p = 0, \dots, N_u - 1$. The obtained trajectories for $N = 10$ and $N_u = 2$ are depicted in Fig. 3 for a few changes of the set-point.

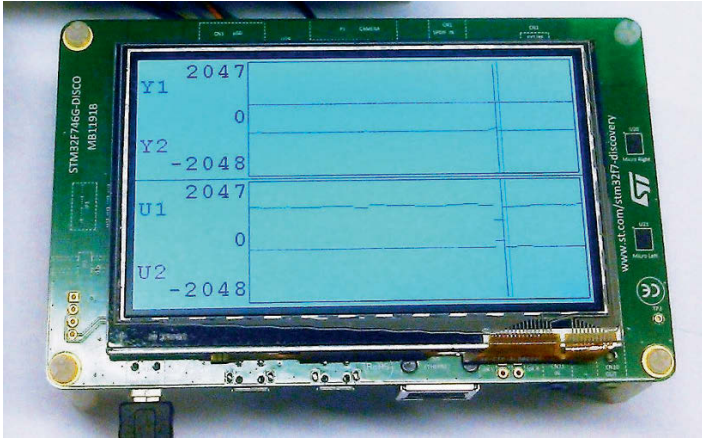


Fig. 2. The development board STM32F746G-DISCO running the GPC algorithm

In the second experiment carried out it is assumed that the process has two manipulated and two controlled variables, i.e. $n_u = n_y = 2$. The simulated process and its model are discrete-time representations of the continuous-time dynamic system

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{1}{0.7s + 1} & \frac{5}{0.3s + 1} \\ \frac{1}{0.5s + 1} & \frac{2}{0.4s + 1} \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} \quad (20)$$

The sampling period of the GPC algorithm is 50 ms whereas the process is emulated with the sampling period 3 ms. The tuning matrices in the minimised cost-function (2) are $\mathbf{M}_p = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ for $p = 1, \dots, N$ and $\mathbf{A}_p = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ $p = 0, \dots, N_u - 1$. The obtained trajectories for $N = 10$ and $N_u = 2$ are depicted in Fig. 4 for a few changes of the set-points.

In the case of both processes the GPC algorithm works correctly, all the changes in the set-point trajectory are followed effectively, new set-points are achieved fast and with no steady-state errors. It is interesting to study the time necessary for calculation the current value(s) of the manipulated variable(s) in one sampling instant, denoted by t_{calc} . Table 1 shows the calculation time as a function of the prediction horizon for selected horizons' lengths whereas Fig. 5 depicts the dependence of the calculation time for all tested horizons $N = 5, 6, \dots, 15$. Considering the obtained results one may easily conclude that implementation of the analytical version of the GPC algorithm is very computationally efficient. Firstly, because for calculation an analytical formula (Eq. (18)) is used, the calculation time depends only on the prediction horizon (and the number of process input and output variables), whereas it is independent of the control horizon. Secondly, for a chosen length of the prediction horizon the calculation time is the same for all sampling instants. It is necessary to point

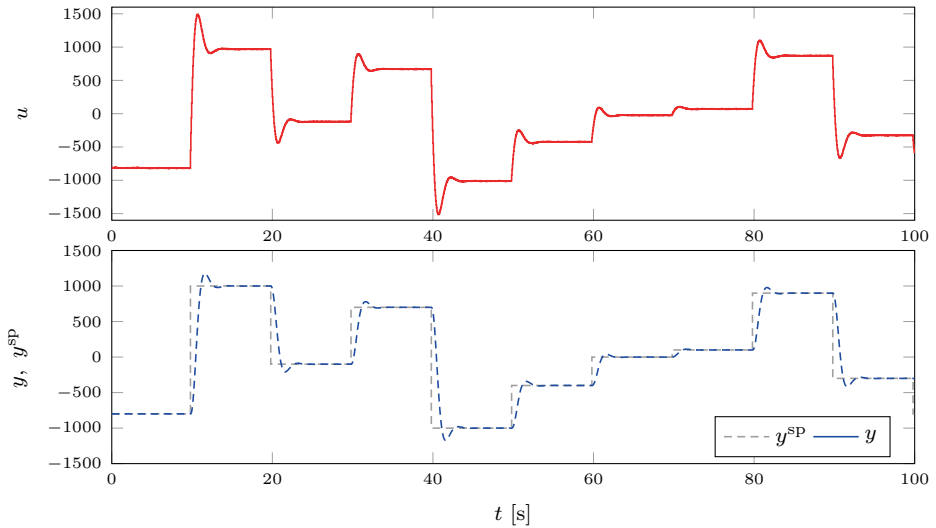


Fig. 3. Experiment no. 1: the manipulated variable (u) and the controlled variable (y) vs. the set-point (y^{SP}) for $N = 10$, $N_u = 2$

Table 1. Calculation time t_{calc} for example prediction horizons (N)

N	$t_{\text{calc}} [\mu\text{s}]$	
	Experiment no. 1 ($n_u = n_y = 1$)	Experiment no. 2 ($n_u = n_y = 2$)
5	120	316
10	161	472
15	201	628

out that the time necessary for calculations is very short. For example, when $N = 10$, in experiment no. 1 ($n_u = n_y = 1$) $t_{\text{calc}} = 161 \mu\text{s}$ and in experiment no. 2 ($n_u = n_y = 2$) $t_{\text{calc}} = 472 \mu\text{s}$. One may easily find the linear relation between the prediction horizon and the calculation time. For the process with one input and one output (experiment no. 1)

$$t_{\text{calc}} = 79.45 + 8.11N$$

and for the process with two inputs and two outputs (experiment no. 2)

$$t_{\text{calc}} = 160.73 + 31.13N$$

It is important to notice that for all considered lengths of the prediction horizon ($5 \leq N \leq 15$) the calculation time (120-628 μs) is much shorter than the sampling time (50 ms).

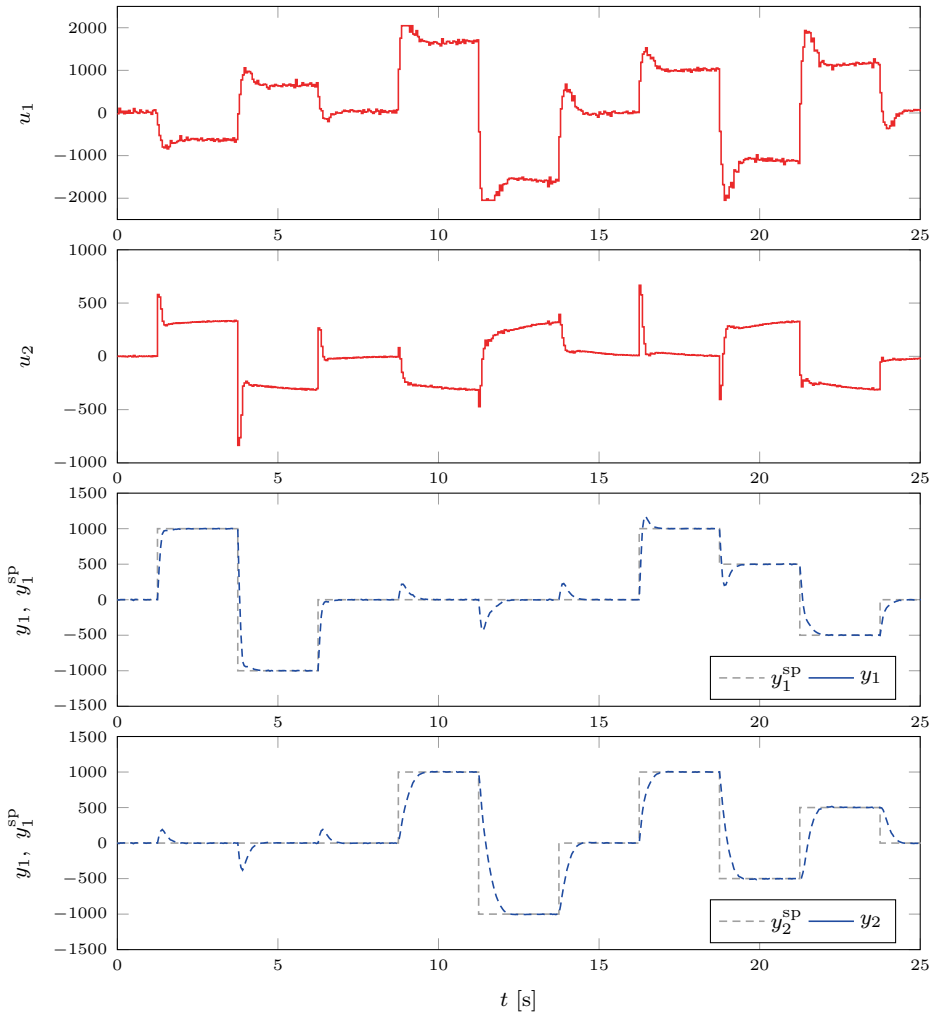


Fig. 4. Experiment no. 2: the manipulated variables (u_1, u_2) and the controlled variables (y_1, y_2) vs. the set-points (y_1^{sp}, y_2^{sp}) for $N = 10, N_u = 2$

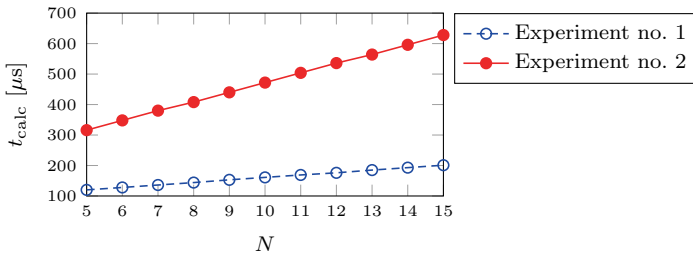


Fig. 5. Calculation time t_{calc} as a function of the prediction horizon (N)

5 Conclusions

This work presents preliminary results of practical implementation of the GPC algorithm on an STM32 microcontroller with the ARM Cortex M7 core with the floating point unit. The GPC algorithm is implemented in its analytical (explicit) version which requires computationally simple matrix and vector operations in real time, no on-line optimisation is necessary. Thanks to that, the time necessary to calculate the current value(s) of the manipulated variable(s) is very short and it depends only on the prediction horizon as well as the number of process inputs and outputs. Because of that, the analytical GPC algorithm implemented on the microcontroller may be used for controlling very fast dynamic processes characterised by sampling periods of millisecond order.

References

1. Camacho, E.F., Bordons, C.: *Model Predictive Control*. Springer, London (1999)
2. Chaber, P., Ławryńczuk, M.: Effectiveness of PID and DMC control algorithms automatic code generation for microcontrollers: application to a thermal process. *Proceedings of the 3rd Conference Control and Fault-Tolerant Systems, SysTol 2016*, pp. 618–623, Barcelona, Spain (2016)
3. Chaber, P., Ławryńczuk, M.: Auto-generation of advanced control algorithms' code for microcontrollers using transcompiler. *Proceedings of the 21th IEEE International Conference on Methods and Models in Automation and Robotics, MMAR 2016*, pp. 454–459, Międzyzdroje, Poland (2016)
4. Clarke, D.W., Mohtadi, C., Tuffs, P.S.: Generalized predictive control. *Automatica* 23 137–160 (1987)
5. Houska, B., Ferreau, H.J., Diehl, M.: An auto-generated real-time iteration algorithm for nonlinear MPC in the microsecond range. *Automatica* 60 2279–2285 (2011).
6. Kufaoor, D.K.M., Aaker, V., Johansen, T.A., Imsland, L., Eikrem, G.O.: Automatically generated embedded model predictive control: Moving an industrial PC-based MPC to an embedded platform. *Optimal Control Applications and Methods* 36, 705–727 (2015)
7. Kunz, K., Huck, S.M., Summers, T.H.: Fast model predictive control of miniature helicopters. *Proceedings of the European Control Conference, ECC 2013*, pp. 1377–1382, Zurich, Switzerland (2013)
8. Kvasnica, M., Rauová I., Fikar M.: Automatic code generation for real-time implementation of model predictive control. *Proceedings of the IEEE International Symposium on Computer-Aided Control System Design, 2010 IEEE Multi-Conference on Systems and Control*, pp. 993–998, Yokohama, Japan (2010)
9. Liniger L., Domahidi A., Morari M.: Optimization-based autonomous racing of 1:43 scale RC cars. *Optimal Control Applications and Methods* 36, 628–647 (2015)
10. Qin, S.J., Badgwell, T.A.: A survey of industrial model predictive control technology. *Control Engineering Practice* 11, 733–764 (2003)
11. Takács G., Batista G., Gulan, M., Rohal'-Ilkiv B.: Embedded explicit model predictive vibration control. *Mechatronics* 36, 54–62 (2016)
12. Tatjewski, P.: *Advanced control of industrial processes, structures and algorithms*. Springer, London (2007)

Robustness of Adaptive Motion Control Against Fuzzy Approximation of LuGre Multi-Source Friction Model

Marcin Jastrzębski, Jacek Kabziński

Institute of Automatic Control, University Lodz of Technology
Stefanowskiego 18/22, 90-924 Lodz, Poland

marcin.jastrzebski@p.lodz.pl, jacek.kabzinski@p.lodz.pl

Abstract. A new modeling technique leading to a simple and reliable dynamic model of multi-source friction is used together with adaptive control algorithm. The model is applied in the electric linear drive system. We give detailed information about preparing the proposed model and the control algorithm. We also investigate effects of the fuzzy model's complexity on the system performance. The presented experiments prove the usefulness of the proposed approach.

Keywords: friction model, fuzzy modeling, servo control, adaptive control.

1 Introduction

Precise servo drives are among the most important components stimulating the development of modern industry, manufacturing, robotics and many other fields. Although the modern motors offer fast and backlash-free thrust force generation, precise encoders provide measurement accuracy to up to single μm , the control algorithm is the most important factor affecting the obtained positioning or tracking quality. It is well recognized that the main factors depressing the servo system performance are the variations of parameters (the load mass for example) and the presence of nonlinear and also variable friction. Therefore the adaptive control technique is one of the most promising approaches to obtain high-quality servos, especially useful for micro-motion devices.

Any adaptive control algorithm is based on the selected friction model. Among several possible dynamic friction models, which are able to capture such phenomena as hysteresis, Dahl effect, 'frictional memory', noninvertible friction characteristics, the most popular one is evidently the LuGre model [1]. The idea and the properties of this model are well described [2] and numerous applications are reported: from automotive applications, ABS system control, through robotics to pneumatic and electric servo control.

The presence of several sources of friction is a typical situation in a servo control problem. Usually, friction is caused by a load machine, several bearings, a motor, etc. Of course, it is possible to use several different models of friction – one for each recognized source, but this will increase the number of parameters drastically and will

make the whole model too complicated. As all friction forces accumulate, working against the motion, it will be impossible to distinguish among the models and to identify all parameters. Therefore in this paper, the simplified, LuGre-type model with one internal state variable and an arbitrary steady-state friction curve [3] is considered.

Even though the adaptive control is able to cope with unknown model parameters, the initial, approximate identification is still necessary, to improve the controller or at least to propose initial conditions for the adaptation. Identification of LuGre model is difficult because its parameters appear nonlinearly in a steady-state characteristic and because the model contains virtual, interior state variable which is, of course, not measurable [4]. Friction modeling and parametric identification become even more complex if the drive system has to compensate several friction forces from several sources.

In the presented paper, the adaptive control algorithm developed in [5] on the basis of a single LuGre model is considered. It is demonstrated that the same controller may be applied also to the multi-source friction problem. As this algorithm requires the approximate value of a certain velocity-dependent coefficient in the internal state variable dynamics, a smart procedure to obtain a fuzzy model of this coefficient is proposed. The obtained fuzzy model is incorporated into the adaptive controller structure and it is verified that the whole system works properly. More over, the robustness against the modeling error is checked and it is confirmed that due to the extremely modeling-friendly structure of the selected approach very simple fuzzy models with a few rules may be applied.

2 Fuzzy Model of Multi-Source Friction

If several (say S) sources of friction affect the system, the complex description of friction may be given as

$$\dot{z}_k = v - \sigma_{0k} \frac{|v|}{f_{ck} + (f_{sk} - f_{ck})g(v, v_{sk})} z_k, \quad (1)$$

$$F_F = \sum_{k=1}^S (\sigma_{0k} z_k + \sigma_{1k} \dot{z}_k) + \sum_{k=1}^S B_k v, \quad (2)$$

where z_k are the internal friction states, B_k are viscous friction coefficients, f_{ck} - Coulomb friction parameters, Stribeck effect is parameterized by f_{sk} and v_{sk} , the parameter σ_{0k} represents stiffness of the bristles, σ_{1k} is the micro damping. The function $g(v, v_s)$ influences the shape of the steady state characteristics, for example $g(v, v_s) = e^{-\left(\frac{v}{v_s}\right)^2}$ [6]. The steady-state friction is

$$F_{FSS}(v_{ss}) = \sum_{k=1}^S [f_{ck} + (f_{sk} - f_{ck})g(v_{ss}, v_{sk})] \operatorname{sgn}(v_{ss}) + \sum_{k=1}^S B_k v_{ss}, \quad (3)$$

where v_{ss} stands for a constant, steady-state velocity and F_{FSS} for a corresponding friction force.

It follows from the analysis of (3) that it is possible to identify the resulting viscous friction coefficient $B = \sum_{k=1}^S B_k$ from the steady-state data, collected for sufficiently big velocities as

$$\Delta F_{FSS} \approx B \Delta v_{SS}, \quad (4)$$

although it is impossible to distinguish among particular coefficients B_k . Changing parameters f_{ck}, f_{sk}, v_{sk} results in various shapes of the characteristics $F_{FSS}(v_{SS})$, but several values of parameters may lead to the same curve $F_{FSS}(v_{SS})$, so identification of all parameters by any curve fitting method is impossible.

These observations motivate to propose a simplified model with one internal state variable, but arbitrary steady-state characteristics:

$$\dot{z} = v - \sigma_0 h(v)z, \quad (5)$$

$$h(v) = \frac{v}{F_{FSS}(v) - Bv}, \quad (6)$$

$$F_F = \sigma_0 z + \sigma_1 \dot{z} + B v. \quad (7)$$

The model (5-7) preserves the viscous friction coefficient and the steady-state friction curve (3), but the coefficient σ_0 (overall stiffness of bristles) and σ_1 (total micro-dumping) must be optimized to reconstruct complicated dynamics described by eq. (1,2). Moreover, the model (5-7) possess the same structure as the single source friction model (1,2) and therefore the controller presented in section 3 may be applied.

The velocity-dependent coefficient (6) may be calculated from the approximated steady-state friction curve, as it was proposed in [5], but this curve is discontinuous for the zero velocity, it is not monotonic, and therefore the modelling errors may be serious.

The coefficient, $h(v)sign(v)$ is monotonic, smooth and $h(v) = 0$, and therefore we propose to obtain and implement the fuzzy model of $h(v)$ itself. The details of the modeling are described in [3,4]. The preparation of the modelling data starts with the identification of B according to (4). Next the velocity/friction force pairs are used to obtain (v, \bar{h}) pairs, according to

$$\bar{h}(v) = \frac{v}{F_{FSS}(v) - Bv} sign(v). \quad (8)$$

Finally, (8) is modeled by a fuzzy inference system with m membership functions μ_j and linear consequents, so the rules are:

$$\text{IF } v \text{ IS } \mu_j \text{ THEN } \bar{h}(v) = \bar{h}_j = p_{0,j} + p_{1,j}v. \quad (9)$$

The initial structure of the Takagi-Sugeno-Kang fuzzy model may be defined by a human expert (by inspection of the shape of the collected data) or automatically, for example as it was described in [7]. Next the fuzzy model is tuned by any standard training algorithm. Because of the regular shape of (8) the results are not sensitive to the selected training technique. In the presented examples fuzzy models with uniformly distributed “generalized bell” membership functions were tuned using ANFIS [8] and none difficulties were observed.

The output of the fuzzy model (9) is given by

$$\bar{h}_{fuz}(v) = \frac{\sum_{j=1}^m \mu_j(v) \bar{h}_j(v)}{\sum_{j=1}^m \mu_j(v)} = \theta^T \xi(v), \quad (10)$$

where

$$\theta^T = [p_{0,1}, p_{1,1}, \dots, p_{0,m}, p_{1,m}], \quad (11)$$

$$\xi^T(v) = [\mu_1(v), v\mu_1(v), \dots, \mu_m(v), v\mu_m(v)] \frac{1}{\sum_{j=1}^m \mu_j(v)}, \quad (12)$$

so it is linear in parameters $[p_{0,1}, p_{1,1}, \dots, p_{0,m}, p_{1,m}]$, and this feature facilitates the model training and enables cooperation with an adaptive control.

The fuzzy rule describing the model for “big” velocity may be simplified, as it is known that

$$h(v) \approx \frac{|v|}{f_c} \text{ for } |v| > 3v_s. \quad (13)$$

The model of $h(v)$ is finally implemented as

$$h_{fuz}(v) = \bar{h}_{fuz}(v) \text{sign}(v) = |\bar{h}_{fuz}(v)|. \quad (14)$$

The proposed approach may be applied without any modifications in case of non-symmetric steady-state friction (hence non-symmetric $h(v)$), but of course the assumed symmetry for positive and negative v simplifies the model and reduces the number of rules.

3 Adaptive motion control

We consider linear motion servo system described by:

$$\dot{x} = v, \quad (15)$$

$$m\dot{v} = F_e - F_F, \quad (16)$$

where: x is the motor position, v – the velocity, m – the forcer mass (with sensors, cables and cart), F_e – the thrust force (control input), F_F – the friction force.

Plugging in the friction force from the LuGre model (5,7) we get the complete plant described by the equations (5, 15) and

$$m\dot{v} = F_e - \sigma_0 z + \sigma_1 \sigma_0 h(v)z - v(\sigma_1 + B), \quad (17)$$

or if we define the coefficients

$$\mu_0 = \sigma_0, \quad \mu_1 = \sigma_0 \sigma_1, \quad \mu_2 = \sigma_1 + B, \quad (18)$$

we get

$$m\dot{v} = F_e - \mu_0 z + \mu_1 h(v)z - \mu_2 v. \quad (19)$$

All parameters m, μ_0, μ_1, μ_2 are constant but unknown. They linearly parameterize right side of (19).

The control objective is to track smooth position trajectory x_d . The tracking error and its dynamic is given by

$$e_x = x - x_d, \quad (20)$$

$$\dot{e}_x = \dot{x} - \dot{x}_d = v - \dot{x}_d. \quad (21)$$

The adaptive control solving the problem formulated above was derived in [5]. The controller consists of:

- the desired velocity and the velocity error

$$v_d = \dot{x}_d - k_1 e_x, \quad e_v = v - v_d \quad (22)$$

with the positive design parameter k_1 ,

- the dual observer of the internal state variable

$$\dot{\hat{z}}_1 = v - \sigma_0 h(v) \hat{z}_1 - \rho_1 e_v, \quad (23)$$

$$\dot{\hat{z}}_2 = v - \sigma_0 h(v) \hat{z}_2 + \rho_2 e_v h(v) \quad (24)$$

with positive design parameters ρ_1, ρ_2 ,

- adaptive laws:

$$\frac{d}{dt} \hat{m} = -\kappa e_v \dot{v}_d, \quad (25)$$

$$\frac{d}{dt} \hat{\mu}_0 = -\gamma_0 e_v \hat{z}_1, \quad (26)$$

$$\frac{d}{dt} \hat{\mu}_1 = \gamma_1 e_v h(v) \hat{z}_2, \quad (27)$$

$$\frac{d}{dt} \hat{\mu}_2 = -\gamma_2 e_v v \quad (28)$$

with positive design parameters $\kappa, \gamma_i, i = 0, 1, 2$,

- the control law:

$$F_e = -k_2 e_v - e_x + \hat{\mu}_0 \hat{z}_1 - \hat{\mu}_1 h(v) \hat{z}_2 + \hat{\mu}_2 v + \hat{m} \dot{v}_d, \quad (29)$$

with the positive design parameter k_2 .

It is proven in [5] that the above controller assures that the tracking errors e_x, e_v converge to zero.

Although the presented controller is able to work properly without any knowledge of parameters m, B, σ_1 it still requires more or less accurate estimates of the velocity

dependent coefficient $h(v)$ and σ_0 to construct the internal state observers (23,24). The effective way of obtaining accurate estimates of these parameters will be illustrated by the examples presented in the following sections and is based on the fuzzy modelling of the coefficient $h(v)$ as it was described in section 2.

4 Numerical Experiments

The concerned model (15,16) describes a linear servo system with a tubular permanent magnet motor. The moving mass is $m=7.04$ kg. The total friction is generated from $s=2$ sources according to (1,2) with parameters presented in table 1.

Table 1. Parameters of two friction sources

i	f_{ci} [N]	f_{si} [N]	v_{si} [mm/s]	B_i [Nm/s]	σ_{0i} [N/m]	σ_{1i} [Ns/m]
1	10	25	2	10	70000	300
2	10	20	20	20	30000	300

The data necessary to model the coefficient $h(v)$ according to the approach presented in section 2 were collected during several runs with the constant velocity. Greater velocities were used to identify the total viscous friction coefficient according to (4). The data prepared according to the formula (8) were used for the fuzzy model training. Although the data were corrupted by a noise and outliers, the influence of the number of rules and membership functions on the modelling accuracy is moderate – satisfactory results are obtained with a small number of rules, due to the modelling-friendly shape of $h(v)$. The training data and resulting plot of 5-rule fuzzy model of $h(v)$ is presented in Fig. 1.

The controller presented in section 3 requires also the approximate value of the parameter σ_0 , which represents stiffness of the bristles. When a slowly varying and a weak force $F_e \ll f_s$ is applied, the so called ‘stick’ motion with $\dot{v} \approx 0$, $v \approx 0$, $z \approx x$ is observed. The system works as a spring and

$$F_e \approx F_F = \sigma_0 z + \sigma_1 \dot{z} + Bv \approx \sigma_0 x + \sigma_1 v + Bv \approx \sigma_0 x. \quad (30)$$

Therefore the stiffness σ_0 can be calculated as the proportional coefficient in (x, F_e) data after a linear approximation

$$\sigma_0 = \frac{\Delta F_e}{\Delta x}. \quad (31)$$

During the conducted experiment the sinusoidal thrust force with the amplitude 40N frequency 0.1rad/s was generated. The obtained coefficient was $\sigma_0 = 55000\text{N/m}$ (Fig.2).

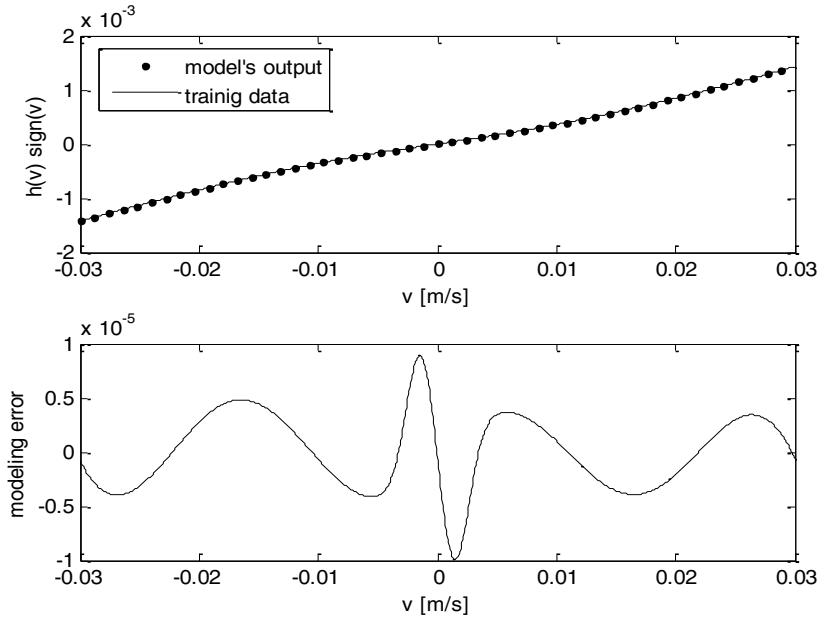


Fig. 1. Coefficient $h(v)$ for multisource friction, its fuzzy model and modeling error

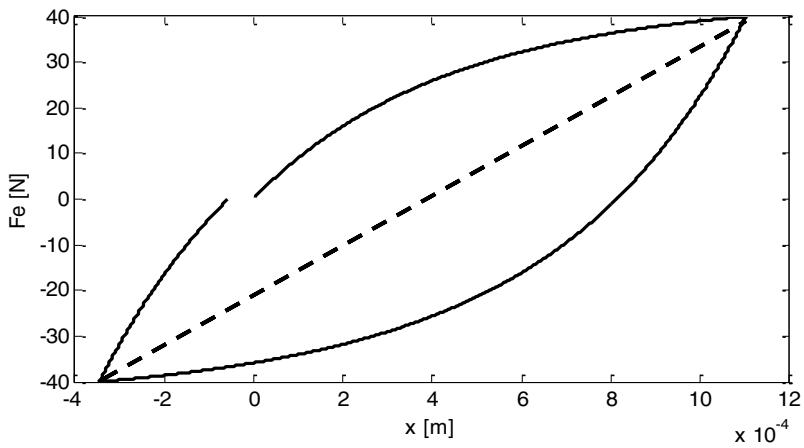


Fig. 2. Static position versus force and the linear approximation (dashed line)

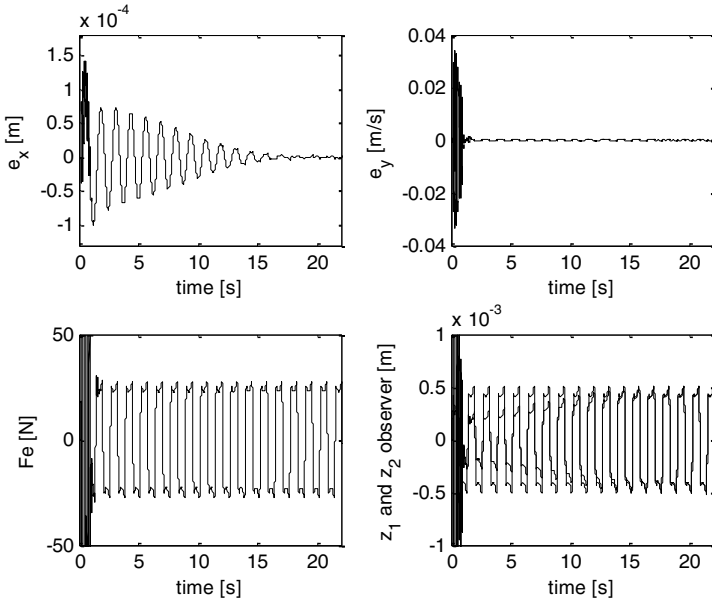


Fig. 3. Tracking errors e_x , e_y , control F_e and observers' output z_1 and z_2

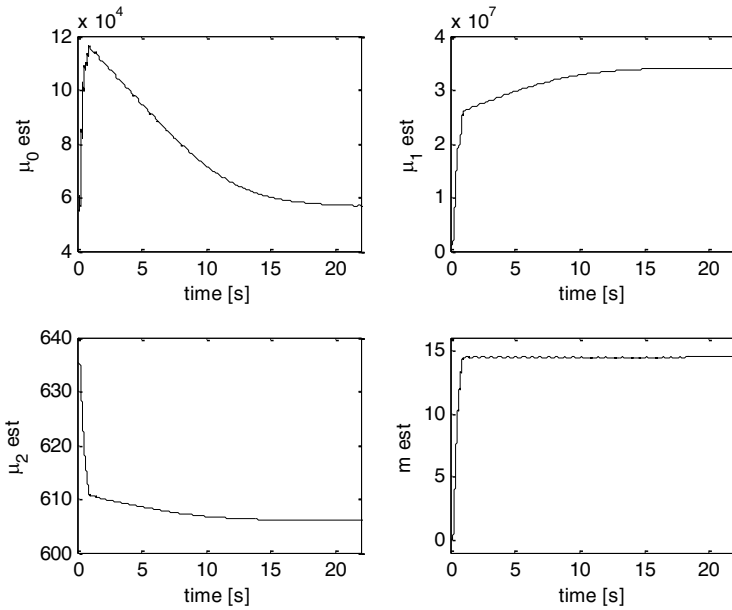


Fig. 4. Time history of adapted parameters

The control system was supposed to track the desired trajectory $x_d = 5 \sin(5t)$ [mm]. The controller parameters were: $k_1=3$, $k_2=3$, $\rho_1=30$, $\rho_2=1$, $\gamma_0=10^{11}$, $\gamma_1=10^{16}$, $\gamma_2=10^5$, $\kappa=10^4$. Great values of γ_i result from very small signals in the adaptive laws (e_v , $h(v)$, $\hat{z}_{1,2}$ are about 10^{-4} , 10^{-3}) and great values of the estimated parameters ($\sim 10^4$, 10^7). In Fig.3, the time history of tracking errors e_x , e_v , control F_e and observer states z_1 and z_2 are presented. In Fig.4, the time history of adapted parameters is demonstrated. The influence of the fuzzy model accuracy is presented in Fig. 5. The quasi-steady-state tracking errors, plotted for the fuzzy model with 2, 5, 10 rules, confirm robustness against the simplicity of the model. Two rules are sufficient to get the average modelling error $<1\%$ (Fig.1). Increasing the number of rules to 10 will decrease the average modelling error twice, but the derivative of the modelling error will be bigger. The maximal position tracking error is almost the same as for any number of rules from 2 till 10 (Fig.5), and the tracking error variance even increases with the number of rules. Therefore it is not recommended to increase the number of rules unnecessarily. Of course the number of the modelling data and the data quality (outliers, noise etc.) will also influence the optimal number of rules, but the general principle “use the simplest possible model” is fully applicable to the discussed problem, due to a very “modelling-friendly” shape of the coefficient $h(v)$ and to the efficiency of the proposed adaptive control.

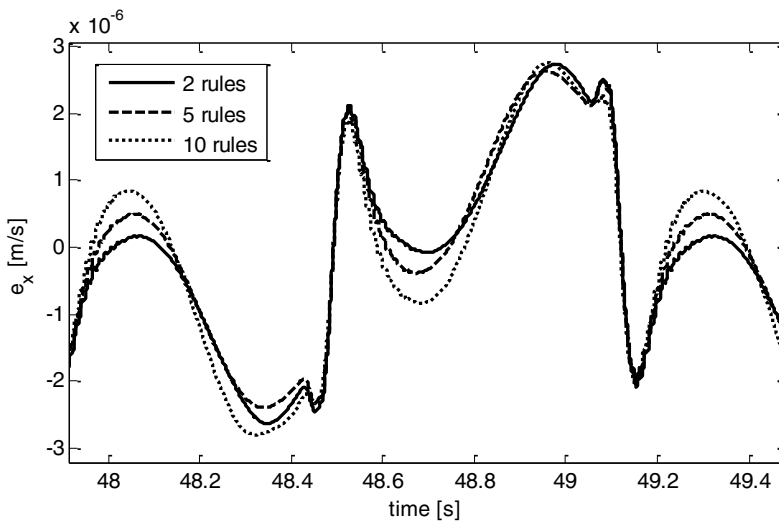


Fig. 5. Position tracking error in quasi-steady state for different complexity of models

5 Conclusion

In the paper, the adaptive control algorithm developed on the simple fuzzy model of friction was considered. The proposed fuzzy model was used for multi-source friction with the internal dynamics. The main part of the proposed model was based on the

measurements recorded during the steady-state drive operation, that are the most precise and reliable type of data. We have successfully demonstrated the usefulness of adaptive control utilizing obtained smooth model to control position of a linear permanent magnet motor, operating in millimetric motion. The robustness against the modeling error was checked. Numerical results confirmed that due to the extremely modeling-friendly structure of the selected approach very simple fuzzy models with a few rules may be applied.

6 References

1. Canudas de Wit C, Lischinsky P (1997) Adaptive friction compensation with partially known dynamic friction model, *Int. J. Adapt. Control Signal Process.*, vol. 11, no. 1, pp. 65–80
2. Åström K J, Canudas de Wit C (2008) Revisiting the LuGre friction model. Stick-slip motion and rate dependence, *IEEE Control Syst. Mag.*, vol. 28, no. 6, pp. 101–114
3. Kabziński J, Jastrzębski M (2015) Fuzzy modeling of complex, multi-source, dynamic friction, XXV International Conference on Information, Communication and Automation Technologies (ICAT), Sarajevo
4. Kabziński J, Jastrzebski M (2015) Fuzzy modeling of LuGre-type friction, CYBCONF 2015: 2nd IEEE International Conference on Cybernetics, Gdynia
5. Kabzinski J, Jastrzebski M (2014) Practical implementation of adaptive friction compensation based on partially identified LuGre model, 19th International Conference On Methods and Models in Automation and Robotics (MMAR), pp.699-704
6. Armstrong-Helouvry B (1991) *Control of Machines with Friction*. Kluwer Academic Publishers, Boston
7. Kabzinski J (2012) Fuzzy friction modeling for adaptive control of mechatronic systems, *Artificial Intelligence Applications and Innovations, IFIP Advances in Information and Communication Technology*, vol. 381, pp.185-195, Springer Berlin Heidelberg
8. Jang, J-S R (1993) ANFIS: Adaptive-Network-based Fuzzy Inference Systems, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 3, pp. 665-685

Part VI
Non-Integer Order Calculus

Relationships between the reachability of positive standard and fractional discrete-time and continuous-time linear systems

Tadeusz Kaczorek

Białystok University of Technology
Faculty of Electrical Engineering
Wiejska 45D, 15-351 Białystok

e-mail: kaczorek@ee.pw.edu.pl

Abstract: The relationships between the reachability of positive standard and fractional discrete-time and continuous-time linear systems are addressed. It is shown that: 1) The fractional positive discrete-time and linear systems are reachable in one step if and only if the corresponding positive standard system is reachable in one step; 2) If the positive standard discrete-time linear system with single input is unreachable, then the corresponding fractional positive system is also unreachable; 3) The fractional positive continuous-time linear system is reachable if and only if the corresponding continuous-time positive standard system is reachable.

Keywords. Key words: fractional, standard, positive, linear, discrete-time, continuous-time, system, reachability.

1 Introduction

The notion of controllability and observability of linear systems have been introduced by Kalman [1, 2]. Those notions are the basic concepts of the modern control theory [3-10]. They have been extended to positive and fractional linear and nonlinear systems [11-19]. The mathematical fundamentals of fractional calculus are given in the monographs [20-22]. The positive fractional linear systems have been introduced in [16, 19].

In the paper [23] it has been shown that the fractional discrete-time and continuous-time linear systems are controllable if and only if the standard discrete-time and continuous-time systems are controllable. Similar problems for the observability have been analyzed in [18].

In this paper the relationships between the reachability of the positive standard and fractional discrete-time and continuous-time linear systems will be analyzed.

The paper is organized as follows. In subsection 2.1 the basic definitions and theorems concerning standard and fractional discrete-time and continuous-time linear systems are recalled. The positivity of standard and fractional discrete-time and continuous-time linear systems is considered in subsection 2.2. The relationship between the reachability of the positive standard and fractional discrete-time linear systems is analyzed in section 3 and of continuous-time linear systems in section 4. Concluding remarks are given in section 5.

The following notation will be used: $\mathfrak{R}^{n \times m}$ is the set of $n \times m$ real matrices, $\mathfrak{R}_+^{n \times m}$ is the set of $n \times m$ real matrices with nonnegative entries and $\mathfrak{R}_+^n = \mathfrak{R}_+^{n \times 1}$, Z_+ is the set of nonnegative integers, M_n is the set of $n \times n$ Metzler matrices (real matrices with nonnegative off-diagonal entries), I_n is the $n \times n$ identity matrix.

2 Preliminaries

2.1 Reachability of linear systems

Consider the standard discrete-time linear system

$$x_{i+1} = Ax_i + Bu_i, \quad i \in Z_+ = \{0, 1, \dots\}, \quad (1)$$

where $x_i \in \mathfrak{R}^n$, $u_i \in \mathfrak{R}^m$ are the state and input vectors and $A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times m}$.

Definition 1. [3, 4] The system (1) is called reachable in q steps if there exists an input sequence u_0, u_1, \dots, u_{q-1} , $q \leq n$ which steers the state of the system from $x_0 = 0$ to the given final state $x_f \in \mathfrak{R}^n$, $x_q = x_f$.

Theorem 1. [3, 4] The system (1) is reachable in q steps if and only if

$$\text{rank}[B \quad AB \quad \dots \quad A^{q-1}B] = n \quad (2)$$

Consider the standard continuous-time linear system

$$\dot{x}(t) = Ax(t) + Bu(t), \quad (3)$$

where $x(t) \in \mathfrak{R}^n$, $u(t) \in \mathfrak{R}^m$ are the state and input vectors and $A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times m}$.

Definition 2. [16] The system (3) is called reachable in the time $[0, t_f]$, $t_f > 0$, if there exists an input $u(t) \in \mathfrak{R}^m$ for $t \in [0, t_f]$ which steers the state of the system from $x(0) = 0$ to the given final state x_f , i.e. $x(t_f) = x_f$.

Theorem 2. [16] The system (3) is reachable in the time $[0, t_f]$ if and only if

$$\text{rank}[B \quad AB \quad \dots \quad A^{q-1}B] = n \text{ for } q \leq n. \tag{4}$$

Now let us consider the fractional discrete-time linear system

$$\Delta^\alpha x_{i+1} = Ax_i + Bu_i, \quad 0 < \alpha < 1, \quad i \in Z_+, \tag{5}$$

where

$$\Delta^\alpha x_i = \sum_{j=0}^i (-1)^j \binom{\alpha}{j} x_{i-j}, \quad \binom{\alpha}{j} = \begin{cases} 1 & \text{for } j = 0 \\ \frac{\alpha(\alpha-1)\dots(\alpha-j+1)}{j!} & \text{for } j = 1, 2, \dots \end{cases} \tag{6}$$

is the fractional α -order difference of x_i and $x_i \in \mathfrak{R}^n$, $u_i \in \mathfrak{R}^m$ are the state and input vectors and $A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times m}$.

Substitution of (6) into (5) yields

$$x_{i+1} = (A + I_n \alpha)x_i + \sum_{j=2}^{i+1} c_j x_{i-j+1} + Bu_i, \quad i \in Z_+, \tag{7}$$

where

$$c_j = (-1)^{j+1} \binom{\alpha}{j}, \quad j = 2, 3, \dots \tag{8}$$

Definition 3. [19] The system (5), (6) is called reachable in q steps if there exists an input sequence u_0, u_1, \dots, u_{q-1} , $q \leq n$ which steers the state of the system from $x_0 = 0$ to the given final state $x_f \in \mathfrak{R}^n$.

Theorem 3. [19] The system (5), (6) is reachable in q steps if and only if

$$\text{rank}[B \quad \Phi_1 B \quad \dots \quad \Phi_{q-1} B] = n, \quad q \leq n, \tag{9}$$

where

$$\Phi_{j+1} = \Phi_j(A + I_n \alpha) + \sum_{k=2}^{j+1} c_k \Phi_{j-k+1}, \quad \Phi_0 = I_n, \tag{10}$$

$$c_k = (-1)^{k+1} \binom{\alpha}{k}, \quad k = 2, 3, \dots$$

Consider the fractional continuous-time linear system

$$\frac{d^\alpha x(t)}{dt^\alpha} = Ax(t) + Bu(t), \quad 0 < \alpha < 1, \tag{11}$$

$$\frac{d^\alpha x(t)}{dt^\alpha} = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{\dot{x}(\tau)}{(t-\tau)^\alpha} d\tau, \quad \dot{x}(\tau) = \frac{dx(\tau)}{d\tau} \tag{12}$$

is the Caputo fractional derivative of order α of $x(t)$, $\Gamma(x)$ is the Euler gamma function, $x(t) \in \mathfrak{X}^n$, $u(t) \in \mathfrak{X}^m$ are the state and input vectors and $A \in \mathfrak{X}^{n \times n}$, $B \in \mathfrak{X}^{n \times m}$.

Definition 4. [19] The system (11), (12) is called reachable in the time $[0, t_f]$, $t_f > 0$, if there exists an input $u(t) \in \mathfrak{X}^m$ for $t \in [0, t_f]$ which steers the state of the system from $x(0) = 0$ to the given final state x_f , i.e. $x(t_f) = x_f$.

Theorem 4. [19] The system (11), (12) is reachable in the time $[0, t_f]$ if and only if the reachability matrix

$$R(t_f) = \int_0^{t_f} \Phi(\tau) B B^T \Phi^T(\tau) d\tau, \quad \Phi(\tau) = \sum_{k=0}^{\infty} \frac{A^k \tau^{(k+1)\alpha-1}}{\Gamma[(k+1)\alpha]} \tag{13}$$

is nonsingular (positive definite), T denotes the transpose.

The input $u(t)$ which steers the state of the system from $x(0) = 0$ to $x_f = x(t_f) \in \mathfrak{X}^n$ is given by

$$u(\tau) = B^T \Phi^T(t_f - \tau) R^{-1}(t_f) x_f, \quad \tau \in [0, t_f]. \tag{14}$$

2.2. Positivity of linear systems

Definition 5. [11, 14] The discrete-time linear system (1) is called (internally) positive if $x_i \in \mathfrak{X}_+^n$, $i \in Z_+$ for any initial condition $x_0 \in \mathfrak{X}_+^n$ and all $u_i \in \mathfrak{X}_+^m$, $i \in Z_+$.

Theorem 5. [11, 14] The discrete-time linear system (1) is positive if and only if

$$A \in \mathfrak{R}_+^{n \times n}, B \in \mathfrak{R}_+^{n \times m}. \tag{15}$$

Definition 6. [11, 14] The continuous-time linear system (3) is called (internally) positive if $x(t) \in \mathfrak{R}_+^n, t \geq 0$ for any initial condition $x(0) \in \mathfrak{R}_+^n$ and all $u(t) \in \mathfrak{R}_+^m, t \geq 0$.

Theorem 6. [11, 14] The continuous-time linear system (3) is positive if and only if

$$A \in M_n, B \in \mathfrak{R}_+^{n \times m}. \tag{16}$$

Definition 7. [19] The fractional discrete-time linear system (5), (6) is called (internally) positive if $x_i \in \mathfrak{R}_+^n, i \in Z_+$ for any initial condition $x_0 \in \mathfrak{R}_+^n$ and all inputs $u_i \in \mathfrak{R}_+^m, i \in Z_+$.

Theorem 7. [19] The fractional discrete-time linear system (5), (6) is positive if and only if

$$A_\alpha = [A + I_n \alpha] \in \mathfrak{R}_+^{n \times n}, B \in \mathfrak{R}_+^{n \times m}. \tag{17}$$

Definition 8. [19] The fractional continuous-time linear system (11), (12) is called (internally) positive if $x(t) \in \mathfrak{R}_+^n, t \geq 0$ for any initial condition $x(0) \in \mathfrak{R}_+^n$ and all inputs $u(t) \in \mathfrak{R}_+^m, t \geq 0$.

Theorem 8. [19] The fractional continuous-time linear system (11), (12) is positive if and only if

$$A \in M_n, B \in \mathfrak{R}_+^{n \times m}. \tag{18}$$

3 Reachability of standard and fractional positive discrete-time linear systems

Definition 9. [14, 16, 19] The standard positive system (1) is called reachable in q steps if there exists an input sequence $u_i \in \mathfrak{R}_+^m, i = 0, 1, \dots, q-1$ which steers the state of the system from $x_0 = 0$ to the given final state $x_f \in \mathfrak{R}_+^n, x_q = x_f$.

Theorem 9. [14, 16, 19] The standard positive system (1) is reachable in q steps if and only if the reachability matrix

$$R_q = [B \quad AB \quad \dots \quad A^{q-1}B] \tag{19}$$

contains n linearly independent monomial columns, i.e. columns which have only one positive entry and the remaining entries are zero.

Theorem 10. [19] The standard positive system (1) is reachable in q steps only if the matrix

$$[B \ A] \quad (20)$$

contains n linearly independent monomial columns.

Definition 10. [19] The fractional positive system (5), (6) is called reachable in q steps if there exists an input sequence $u_i \in \mathfrak{R}_+^m$, $i = 0, 1, \dots, q-1$ which steers the state of the system from $x_0 = 0$ to the given final state $x_f \in \mathfrak{R}_+^n$, $x_q = x_f$.

Theorem 11. [19] The fractional positive system (5), (6) is reachable in q steps if and only if the reachability matrix

$$\bar{R}_q = [B \ \Phi_1 B \ \dots \ \Phi_{q-1} B] \quad (21)$$

contains n linearly independent monomial columns.

Theorem 12. [19] The fractional positive system (5), (6) is reachable in q steps only if the matrix

$$[B \ A + I_n \alpha] \quad (22)$$

contains n linearly independent monomial columns.

Example 1. Consider the standard discrete-time linear system (1) with the matrices

$$A = \begin{bmatrix} 0 & 2 \\ 1 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (23)$$

and the fractional discrete-time linear system (5), (6) for $\alpha = 0.5$ and with the same matrices (23).

Both systems are positive since the matrices (23) have nonnegative entries and

$$A_\alpha = A + I_n \alpha = \begin{bmatrix} 0 & 2 \\ 1 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} 0.5 = \begin{bmatrix} 0.5 & 2 \\ 1 & 1.5 \end{bmatrix} \in \mathfrak{R}_+^{2 \times 2}. \quad (24)$$

The positive standard system (23) is reachable in $q = 2$ steps since the matrix

$$[B \quad AB] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (25)$$

contains two linearly independent monomial columns, i.e. it is a monomial matrix.

The positive fractional system with (23) is not reachable in $q = 2$ steps since the matrix

$$[B \quad A_\alpha B] = \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \quad (26)$$

has only one monomial column.

From comparison of Theorems 8 – 12 it follows that in general case for $q > 1$ the conditions for reachability of standard and fractional positive systems are different. In particular case for $q = 1$ the reachability depends only on the matrix B and it is independent of the matrices A and $A_\alpha = A + I_n \alpha$. If the matrix B contains n linearly independent monomial columns then the both systems are reachable in one step ($q = 1$). Therefore, we have the following theorem.

Theorem 13. The fractional positive system (5), (6) is reachable in one step ($q = 1$) if and only if the standard positive system (1) is reachable in one step.

Example 2. Consider the standard and fractional positive linear systems (1) and (5), (6) with the matrices

$$A = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \quad (27)$$

The standard positive system with (27) is unreachable for any $q > 0$ since the matrix

$$[B \quad AB] = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \quad (28)$$

has only one monomial column.

The fractional positive system is also unreachable in $q = 2$ steps since the matrix

$$[B \quad A_\alpha B] = \begin{bmatrix} 0 & 0 \\ 1 & \alpha \end{bmatrix}, \quad 0 < \alpha < 1 \quad (29)$$

has only one linearly independent monomial column.

Note that for $n=2$, $m=1$ and $q=2$ if the standard positive system is unreachable then the fractional positive system is also unreachable.

It is easy to prove the following theorem.

Theorem 14. If the standard positive system for $n > 2$, $m=1$ and $q=n$ is unreachable, then the corresponding fractional positive system is also unreachable.

Example 3. Consider the positive standard and fractional linear systems with the matrices

$$A = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}. \quad (30)$$

The standard system with (30) is reachable in $q=3$ steps since the reachability matrix

$$R_3 = [B \quad AB \quad A^2B] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (31)$$

is a monomial matrix, i.e. it has 3 linearly independent monomial columns.

The reachability matrix of the fractional system

$$\begin{aligned} \bar{R}_3 &= [B \quad \Phi_1 B \quad \Phi_2 B] = [B \quad A_\alpha B \quad (A_\alpha^2 + c_2 I_3)B] \\ &= \begin{bmatrix} 1 & 0 & \alpha^2 \\ 0 & 1 & 2\alpha \\ 0 & 0 & * \end{bmatrix}, \quad (* - \text{positive number}) \end{aligned} \quad (32)$$

has only one monomial column. Therefore, the corresponding fractional positive system is unreachable.

In general case for $n > 3$ and $m=1$ we have the following theorem.

Theorem 15. If the matrix A has the Frobenius form

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & a_0 \\ 1 & 0 & \cdots & 0 & a_1 \\ 0 & 1 & \cdots & 0 & a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & a_{n-1} \end{bmatrix}, a_k \geq 0, k = 0, 1, \dots, n-1 \text{ and } B = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (33)$$

then the standard positive system is reachable in n steps but the corresponding fractional positive system is unreachable.

Proof. The reachability matrix of the standard system with matrices (33)

$$R_n = [B \ AB \ \cdots \ A^{n-1}B] = I_n \quad (34)$$

and the positive system is reachable in $q = n$ steps.

It is easy to check that the reachability matrix

$$\bar{R}_n = [B \ \Phi_1 B \ \cdots \ \Phi_{n-1} B] \quad (35)$$

of the fractional system contains only one first monomial column. Therefore, the fractional positive system is unreachable. \square

4 Reachability of standard and fractional positive continuous-time linear systems

Definition 11. [14] The standard positive system (3) is called reachable in the time $[0, t_f]$, $t_f > 0$, if there exists an input $u(t) \in \mathfrak{R}_+^m$ for $t \in [0, t_f]$ which steers the state of the system from $x(0) = 0$ to the given final state $x_f \in \mathfrak{R}_+^n$, i.e. $x(t_f) = x_f$.

Theorem 16. [14, 17] The standard positive system (3) is reachable in the time $[0, t_f]$ if and only if the reachability matrix

$$R(t_f) = \int_0^{t_f} e^{A\tau} B B^T e^{A^T \tau} (\tau) d\tau \in \mathfrak{R}_+^{n \times n} \quad (36)$$

is a monomial matrix.

The input $u(t) \in \mathfrak{R}_+^m$, $t \in [0, t_f]$ which steers the state of the system from $x(0) = 0$ to the given final state $x_f \in \mathfrak{R}_+^n$ is given by

$$u(\tau) = B^T e^{A^T(t_f-\tau)} R^{-1}(t_f) x_f \in \mathfrak{R}_+^m, \tau \in [0, t_f]. \tag{37}$$

Definition 12. [19] The fractional positive system (11), (12) is called reachable in the time $[0, t_f]$, $t_f > 0$, if there exists an input $u(t) \in \mathfrak{R}^m$ for $t \in [0, t_f]$ which steers the state of the system from $x(0) = 0$ to the given final state $x_f \in \mathfrak{R}_+^n$, i.e. $x(t_f) = x_f$.

Theorem 17. The fractional positive system (11), (12) is reachable in the time $[0, t_f]$ if and only if the reachability matrix

$$\bar{R}(t_f) = \int_0^{t_f} \Phi(\tau) B B^T \Phi^T(\tau) d\tau \in \mathfrak{R}_+^{n \times n}, \tag{38}$$

is a monomial matrix.

The input $u(t)$ which steers the state of the system from $x(0) = 0$ to $x_f = x(t_f) \in \mathfrak{R}^n$ is given by

$$u(\tau) = B^T \Phi^T(t_f - \tau) \bar{R}^{-1}(t_f) x_f \in \mathfrak{R}_+^m, \tau \in [0, t_f]. \tag{39}$$

Proof. It is well-known [14] that $\bar{R}^{-1}(t_f) \in \mathfrak{R}_+^{n \times n}$ if and only if the matrix (38) is monomial. Substituting (39) into

$$x(t_f) = \int_0^{t_f} \Phi(t_f - \tau) B u(\tau) d\tau \tag{40}$$

we obtain

$$\begin{aligned} x(t_f) &= \int_0^{t_f} \Phi(t_f - \tau) B B^T \Phi^T(t_f - \tau) \bar{R}^{-1}(t_f) x_f d\tau \\ &= \int_0^{t_f} \Phi(\tau) B B^T \Phi^T(\tau) d\tau \bar{R}^{-1}(t_f) x_f = x_f. \end{aligned} \tag{41}$$

Therefore, the input (39) steers the state of the system from $x(0) = 0$ to $x(t_f) = x_f$. \square

Theorem 18. The fractional positive continuous-time linear system (11), (12) is reachable in the time $[0, t_f]$ if and only if the standard positive continuous-time linear system (3) is reachable in the same time $[0, t_f]$.

Proof. Note that the reachability matrices (36) and (38) of the standard positive system (3) and of fractional positive system (11), (12) differ only by the transition matrices e^{At} for standard system and $\Phi(t)$ (defined by (13)) for fractional system. Using the well-known Cayley-Hamilton theorem or the Lagrange-Sylvester formula [3, 13] it is possible to write the transition matrices in the forms

$$e^{At} = \sum_{k=0}^{n-1} c_k(t) A^k \tag{42}$$

and

$$\Phi(t) = \sum_{k=0}^{n-1} \bar{c}_k(t) A^k, \tag{43}$$

where $c_k(t)$ and $\bar{c}_k(t)$ for $k = 0, 1, \dots, n-1$ are nonzero linearly independent functions of time t [5, 24].

Therefore, the reachability matrix (38) is monomial if and only if the reachability matrix (36) is monomial. By Theorems 16 and 17 the fractional positive system (11), (12) is reachable in the time $[0, t_f]$ if and only if the standard positive system (3) is reachable in the time $[0, t_f]$. \square

Example 4. Consider the standard and fractional positive systems (3) and (11), (12) with the same matrices

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{44}$$

Using Lagrange-Sylvester formula [3, 13] and (44) we obtain

$$e^{At} = c_0(t)I_2 + c_1(t)A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + (e^t - 1) \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, c_0(t) = 1, c_1(t) = e^t - 1 \tag{45}$$

and

$$\Phi(t) = \sum_{k=0}^{\infty} \frac{A^k t^{(k+1)\alpha-1}}{\Gamma[(k+1)\alpha]} = \bar{c}_0(t)I_2 + \bar{c}_1(t)A, \quad \bar{c}_0(t) = \frac{t^{\alpha-1}}{\Gamma(\alpha)},$$

$$\bar{c}_1(t) = \sum_{k=1}^{\infty} \frac{t^{(k+1)\alpha-1}}{\Gamma[(k+1)\alpha]}. \quad (46)$$

Substituting (45) and (44) into (36) we obtain

$$R(t_f) = \int_0^{t_f} e^{A\tau} B B^T e^{A^T \tau}(\tau) d\tau = \int_0^{t_f} \begin{bmatrix} e^{2\tau} & 0 \\ 0 & 1 \end{bmatrix} d\tau = \begin{bmatrix} \frac{1}{2}(e^{2t_f} - 1) & 0 \\ 0 & t_f \end{bmatrix} \in \mathfrak{R}_+^{2 \times 2}. \quad (47)$$

The matrix (47) is monomial for $t_f > 0$ and by Theorem 16 the standard positive system with the matrices (44) is reachable in the time $[0, t_f]$.

Similarly, substituting (46) and (44) into (38) we obtain

$$\bar{R}(t_f) = \int_0^{t_f} \Phi(\tau) B B^T \Phi^T(\tau) d\tau = \int_0^{t_f} \begin{bmatrix} [c_0(\tau) + c_1(\tau)]^2 & \\ & c_0^2(\tau) \end{bmatrix} d\tau \in \mathfrak{R}_+^{2 \times 2}. \quad (48)$$

The matrix (48) is monomial for $t_f > 0$ and by Theorem 17 the fractional positive system with the matrices (44) and $0 < \alpha < 1$ is reachable in the time $[0, t_f]$.

5 Concluding remarks

Relationships between the reachability of positive standard and fractional discrete-time and continuous-time linear systems have been addressed. It has been shown that:

- 1) The fractional positive discrete-time linear systems are reachable in one step if and only if the corresponding standard positive system is reachable in one step (Theorem 13).
- 2) If the standard positive discrete-time linear system with single input ($m=1$) is unreachable then the corresponding fractional system is also unreachable (Theorem 14).

- 3) If the matrices A and B has the Frobenius form (3.15) then the standard positive system is reachable in n steps but the corresponding fractional positive system is unreachable (Theorem 15).
- 4) The fractional positive continuous-time linear system is reachable in the time $[0, t_f]$ if and only if the corresponding continuous-time standard positive system is reachable in the same interval (Theorem 18).

The considerations have been illustrated by numerical examples of positive discrete-time and continuous-time linear systems. An extension of these considerations to standard and fractional positive time-varying linear systems is an open problem.

Acknowledgment

This work was supported by National Science Centre in Poland under work No. 2014/13/B/ST7/03467.

References

1. Kalman R.: Mathematical description of linear systems, SIAM J. Control, vol. 1, no. 2, pp. 152-192, 1963.
2. Kalman R.: On the general theory of control systems, Prof. First Intern. Congress on Automatic Control, Butterworth, London, pp. 481-493, 1960.
3. Antsaklis P., Michel A.: Linear Systems, Birkhauser, Boston, 2006.
4. Kaczorek T.: Linear Control Systems, Vol. 1, J. Wiley, New York, 1999.
5. Kaczorek T.: Vectors and Matrices in Automation and Electrotechnics, WNT, Warsaw, 1998 (in Polish).
6. Kailath T.: Linear Systems, Prentice Hall, Englewood Cliffs, New York, 1980.
7. Klamka J.: Controllability of Dynamical Systems, Kluwer, Academic Press, Dordrecht, 1991.
8. Rosenbrock H.: State-space and multivariable theory, J. Wiley, New York, 1970.
9. Wolovich W.: Linear multivariable systems, Springer-Verlag, New York, 1974.
10. Żak S.H.: Systems and Control, Oxford University Press, New York, 2003.

11. Farina L., Rinaldi S.: *Positive Linear Systems: Theory and Applications*, J. Wiley & Sons, New York, 2000.
12. Kaczorek T.: Constructability and observability of standard and positive electrical circuits, *Electrical Review*, vol. 89, no. 7, pp. 132-136, 2013.
13. Kaczorek T.: Controllability and observability of linear electrical circuits, *Electrical Review*, vol. 87, no. 9a, pp. 248-254, 2011.
14. Kaczorek T.: *Positive 1D and 2D systems*, Springer-Verlag, London, 2002.
15. Kaczorek T.: Positive linear systems consisting of n subsystems with different fractional orders, *IEEE Trans. Circuits and Systems*, vol. 58, no. 6, pp. 1203-1210, 2011.
16. Kaczorek T.: Reachability and controllability to zero tests for standard and positive fractional discrete-time systems, *Journal Européen des Systemes Automatisés, JESA*, vol. 42, no. 6-8, pp. 769-787, 2008.
17. Kaczorek T.: Reachability and observability of fractional positive electrical circuits, *Computational Problems of Electrical Engineering*, vol. 23, no. 2, pp. 28-36, 2013.
18. Kaczorek T.: Relationship between the observability of standard and fractional linear systems, 2016.
19. Kaczorek T.: *Selected Problems of Fractional Systems Theory*, Springer-Verlag, Berlin, 2011.
20. Oldham K., Spanier J.: *The Fractional Calculus: Integrations and Differentiations of Arbitrary Order*, Academic Press, New York, 1974.
21. Ostalczyk P.: *Epitome of the Fractional Calculus, Theory and its Applications in Automatics*, Technical University of Lodz Press, Lodz, 2008 (in Polish).
22. Podlubny I.: *Fractional Differential Equations*, Academic Press, San Diego, 1999.
23. Klamka J.: Relationship between controllability of standard and fractional linear systems, Submitted to KKA 2017.
24. Gantmacher F.R.: *The Theory of Matrices*, Chelsea Pub. Comp., London, 1959.

Remarks on descriptor fractional-order systems with l -memory and its stability in Lyapunov sense

Ewa Pawłuszewicz

Białystok University of Technology,
Faculty of Mechanical Engineering
Wiejska 45C, 15-351 Białystok, Poland

Abstract. Fractional order linear descriptor systems with finite memory are studied. The formula for trajectory of such system is given. The Lyapunov-Krasovskii approach is used to analyze the stability of the considered systems.

Keywords: Grunwald-Letnikov-type h difference operator, l -memory, descriptor system, stability

1 Introduction

Analysis of experiments results shows that there is a large class of systems where behaviors of real phenomena are not properly explained by using classical calculus. It has been found that these systems not only contain non-local dynamics involving memory but also can be described using fractional-order operators, see [1], also [3, 11, 17, 18].

To the most popular non-integer operators, among the others, are fractional order Riemann-Liouville derivative and fractional order Caputo derivative. The first one can be used successfully in practical issues related to a non-zero initial conditions, see for example [6, 11] and references therein. The most reason is that in many cases the past values of real phenomena should be memorized, see for example in [1, 6, 17]. In practice, the memory of the considered phenomena has influence on the present values of the process and on its future. Initialized fractional order Riemann-Liouville derivative is defined in the following way, see [1, 6]

$$\begin{aligned} {}^{RL}D^\alpha f(t) &= \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{1+\alpha-n}} d\tau, \quad t > a \\ f(t) &:= \begin{cases} \phi(t), & \text{for } t_0 < t \leq a; \\ 0, & \text{for } t \leq t_0. \end{cases} \end{aligned} \quad (1)$$

with $n-1 < \alpha \leq n$, $n \in \mathbb{N}$, and $f \in \mathcal{C}^{n-1}$. In (1) function $\phi(t)$ represents the initial history of the process described by f . In practice approach an approximation or discretization of fractional order Riemann-Liouville derivative should

be introduced and used. It is well known that the one of method among the others, termed as *short memory principle*, see [17], is based on definition of the fractional order Grünwald-Letnikov definition, see for example [12, 16, 17]. The idea of the short memory principle is to consider the behavior of function f only on the interval $[t - L; t]$ where L denotes the length of the memory, i.e. to take L in

$${}^{RL}D^\alpha f(t) \approx_{{}^{RL}D^\alpha f(t)} {}^{RL}D^\alpha f(t)$$

for any $t > t_0 + L$ and $\alpha > 0$.

Our goal is to study, basing on [15], a descriptor linear systems of fractional order with the initialization given by additional function φ that vanishes on a time interval with finitely many points. This way a finite set of initial values (not necessary zero), called l -memory, is obtained.

Generalizations of n -th order differences to their fractional forms are used. In [7] there was adopted a more general fractional h -difference Riemann-Liouville operator. On one hand h represents a sample step, on the other - when h tends to zero, the solutions of the fractional difference equation may be seen as approximations to the solutions of corresponding Riemann-Liouville fractional equations. In [8] it was shown that the Grünwald-Letnikov-type fractional h -difference operator can be expressed by the Riemann-Liouville-type fractional h -difference operator. So, systems under our consideration with these types of operators are studied simultaneously.

The work is motivated by results discussed in [15, 20] and [19]. In Section 2 the notation and facts concerning fractional order difference operators and idea of l -memory are presented. Also some properties of l -memory are discussed. Section 3 presents fractional order linear systems with initialization (with l -memory). The formula for their trajectory is given. In Section 4 asymptotic stability in Lyapunov-Krasovskii sense of the considered systems is study.

2 Preliminaries

If $h > 0$ and $t_0 \in \mathbb{R}$ then we put $(h\mathbb{Z})_{t_0} := \{t_0, t_0 + h, t_0 + 2h, \dots\}$. Consider a function $x : (h\mathbb{N})_{t_0} \rightarrow \mathbb{R}$. The forward h -difference operator is classically defined as $(\Delta_h x)(t) = \frac{x(t+h) - x(t)}{h}$.

Let Γ denotes the Euler function, i.e.

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt$$

where $z \in \mathbb{C}$, $\Re(z) > 0$, and

$$\binom{\alpha}{j} = \begin{cases} 1 & \text{for } k = 0, \\ \frac{\alpha(\alpha-1)\dots(\alpha-j+1)}{k!} & \text{for } k \in \mathbb{N}. \end{cases}$$

be the classical binomial coefficient. Then the fractional h -sum of order α , $\alpha > 0$ is defined as, see [2]

$$({}_{t_0}\Delta_h^{-\alpha}x)(t) = \frac{1}{\Gamma(\alpha)} \sum_{k=\frac{t_0}{h}}^{\frac{t}{h}-t_0} (t - (k + 1)h)_h^{(\alpha-1)} x(kh)h \tag{2}$$

where $t_h^{(\alpha)} := h^\alpha \frac{\Gamma(\frac{t}{h}+1)}{\Gamma(\frac{t}{h}+1-\alpha)}$ is the h -factorial function. In [8] it was shown that equivalently fractional h -sum (2) for $t = t_0 + (\alpha+n)h$, $n \in \mathbb{N}_0$ can be equivalently expressed as

$$({}_{t_0}\Delta_h^{-\alpha}x)(t) = h^\alpha \sum_{j=0}^n (-1)^j \binom{-\alpha}{j} x(t_0 - jh).$$

The Grünwald-Letnikov-type fractional h -difference operator ${}_{t_0}\Delta_h^\alpha$ of order α for a function $x : (h\mathbb{N})_{t_0} \rightarrow \mathbb{R}$ is defined by, see [8]

$$({}_{t_0}^{GL}\Delta_h^\alpha x)(t) := \sum_{j=0}^{\frac{t-t_0}{h}} a_j(\alpha) x(t - jh)$$

where

$$a_j^{(\alpha)} := (-1)^j \binom{\alpha}{j} \frac{1}{h^\alpha}.$$

Proposition 1. [8] *Let $t_0 = (\alpha - 1)h$. Then*

$$\nabla_h({}_{t_0}\Delta_h^{-(1-\alpha)}x)(nh) = ({}_{t_0}^{GL}\Delta_h^\alpha y)(nh)$$

where $y(nh) := x(nh + t_0)$ or $x(t) = y(t - t_0)$ for $t \in (h\mathbb{N})_{t_0}$ and $(\nabla_h\varphi)(nh) = \frac{\varphi(nh) - \varphi(nh-h)}{h}$.

Following [15], for any $l \in \mathbb{N}_0$ and $t_0 \in \mathbb{R}$ let us define the set

$$\Omega_l(t_0) := \{t_0, t_0 - h, \dots, t_0 - lh\}.$$

Let $D := D(\Omega_l(t_0), \mathbb{R}^{n(l+1)})$ denote the state space of continuous functions $\phi : \Omega_l(t_0) \rightarrow \mathbb{R}^{n(l+1)}$. Then

$$\|\phi\|_D := \sup_{\Theta \in \Omega_l(t_0)} \|\phi(\Theta)\| \quad \text{and} \quad D^\gamma := \{\phi \in D : \|\phi\|_D < \gamma, \gamma \in \mathbb{R}\} \subseteq D.$$

Definition 1. [15] *Let $l \in \mathbb{N}_0$, $a \in \mathbb{R}$ and $\varphi : \mathbb{R} \rightarrow \mathbb{R}^n$. The vector*

$$M(l, t_0, \varphi, h) := \begin{bmatrix} \varphi(t_0) \\ \varphi(t_0 - h) \\ \vdots \\ \varphi(t_0 - lh) \end{bmatrix} \tag{3}$$

of values of the function φ on the set $\Omega_l(t_0)$, is called a finite l -memory at t_0 .

Note that $M(l, t_0, \varphi, h) \in \mathbb{R}^{n+nl}$ and $M(l, t_0, \varphi, h) \in D$. If $l_1, l_2 \in \mathbb{N}_0, l_2 \geq l_1$, then $\Omega_{l_1}(t_0) \subset \Omega_{l_2}(a)$. Moreover,

$$[I_{nl_1}, \mathbf{0}_{nl_1 \times n(l_2-l_1)}] M(l_2, t_0, \varphi, h) = M(l_1, a, \varphi, h),$$

where $\mathbf{0}_{nl_1 \times n(l_2-l_1)}$ denotes the zero matrix of dimension $nl_1 \times n(l_2 - l_1)$ and I_{nl_1} is the identity matrix of dimension $nl_1 \times nl_1$.

We assume that $\|M(l, t_0, \varphi, h)\|_D \in D^\infty$.

Definition 2. [15] Let $t_0 \in \mathbb{R}, h > 0, \alpha \in \mathbb{R}$. The fractional h -difference of order α of a given function $x : \mathbb{R} \rightarrow \mathbb{R}^n$, starting at t_0 , with respect to the l -memory $M(l, t_0, \varphi, h)$, is defined in the following way

$$({}_{t_0}\Delta_{h,l}^\alpha x)(t) := \sum_{j=0}^{\lfloor \frac{t-t_0}{h} + l \rfloor} a_j^{(\alpha)} x(t - jh),$$

where $x(t) = \varphi(t)$ for any $t \leq t_0$.

Proposition 2. Let $x : (hN)_{t_0} \rightarrow \mathbb{R}^n, \alpha \in \mathbb{R}_+$ and $l \in \mathbb{N}_0$. Suppose that there exists a real positive constant K such that $|x(t)| \leq K$ for any $t \geq t_0$. Then the difference ε between values of the Grünwald–Letnikov–type fractional h -operator $({}_{t_0}^{GL}\Delta_h^\alpha x)(t)$ and of the fractional h -difference with respect to the l -memory $({}_{t_0}\Delta_{h,l}^\alpha x)(t)$ is less or equal to $\frac{K}{h^\alpha} \left| \binom{\alpha-1}{l} \right|$, i.e.

$$|\varepsilon| \leq \frac{K}{h^\alpha} \left| \binom{\alpha-1}{l} \right|.$$

Proof. Since $\sum_{j \leq m} (-1)^j \binom{\alpha}{j} = (-1)^m \binom{\alpha-1}{m}$ for any integer m , see [10], then

$$\begin{aligned} |\varepsilon| &= |({}_{t_0}^{GL}\Delta_h^\alpha x)(t) - ({}_{t_0}\Delta_{h,l}^\alpha x)(t)| = \left| \sum_{j=0}^l a_j^{(\alpha)} x(t - jh) \right| \\ &\leq K \left| \sum_{j=0}^l a_j^{(\alpha)} \right| \leq \frac{K}{h^\alpha} \left| (-1)^l \binom{\alpha-1}{l} \right| = \frac{K}{h^\alpha} \left| \binom{\alpha-1}{l} \right|. \quad \square \end{aligned}$$

As an immediate consequence of Proposition 2 we have the following.

Corollary 1. Let $x : (hN)_{t_0} \rightarrow \mathbb{R}^n, \alpha \in \mathbb{R}_+$ and $l \in \mathbb{N}_0$. Suppose that there exists a real positive constant K such that $|x(t)| \leq K$ for any $t \geq t_0$. Then the length of l -memory $M(l, t_0, \varphi, h)$ in the fractional h -difference ${}_{t_0}\Delta_{h,l}^\alpha$ of the function x should be such that

$$\left| \binom{\alpha-1}{l} \right| \geq \frac{h^\alpha |\varepsilon|}{K}$$

where ε is the difference between values of the Grünwald–Letnikov–type fractional h -operator $({}_{t_0}^{GL}\Delta_h^\alpha x)(t)$ and of the fractional h -difference with respect to the l -memory $({}_{t_0}\Delta_{h,l}^\alpha x)(t)$.

3 Fractional order systems with initialization

Although, the order α in Definition 2 could be any real number, for our purpose we use $\alpha > 0$.

Definition 3. Let $t_0 \in \mathbb{R}$, $l \in \mathbb{N}_0$ and $a = t_0 - lh$. A discrete-time descriptor linear fractional-order system with l -memory, denoted by $\Sigma_{(\varphi,l)}$, is a system given by the following set of equations:

$$E ({}_t\Delta_{h,l}^\alpha x) (t + h) = Ax(t), \quad t = t_0 + kh, k \in \mathbb{N}_0 \tag{4}$$

$$x(t) = \varphi(t)u_a(t), \quad t \leq t_0 \tag{5}$$

where $E \in \mathbb{R}^{n \times n}$ is a singular constant matrix, $A \in \mathbb{R}^{n \times n}$, is a nonsingular constant matrices, x is the state vector and $u_a : \mathbb{R} \rightarrow \{0, 1\}$ denotes the Heaviside step function.

Definition 4. l -memory $M(l, t_0, \varphi, h)$ given by (3) is a consistent l -memory (associated with t_0) for control system $\Sigma_{(\varphi,l)}$ if system (4)-(5) has at least one solution.

We assume that the matrix pair (E, A) is regular, i.e. $\det(\lambda E - A) \neq 0$ for some $\lambda \in \mathbb{C}$. The solution of l -memory initial value problem (4)-(5) corresponding to values of the function φ , denoted shortly by $\psi(k, M)$, describes the trajectory of the system $\Sigma_{(\varphi,l)}$.

Applying the definition of Grünwald-Letnikov h -difference fractional operator to (4) we obtain

$$Ex(t_0 + (k+1)h) = h^\alpha \left(A - Ea_1^{(\alpha)} \right) x(t_0 + kh) - \sum_{j=1}^{k+l} a_{j+1}^{(\alpha)} h^\alpha Ex(t_0 - (j-k)h) \tag{6}$$

with $x(t) = \varphi(t)u_a(t)$ for $t \leq t_0$. Denoting

$$G := h^\alpha(A - Ea_1^{(\alpha)}), \quad E_j := -a_{j+1}^{(\alpha)}h^\alpha E$$

for any $j \in \mathbb{N}$ and $E_0 = 0$ for $j = 0$, equation (6) can be rewritten as

$$Ex(t + h) = Gx(t) - \sum_{j=1}^{k+l} E_j x(t - jh) \tag{7}$$

where $t = t_0 + kh$, $k \in \mathbb{N}_0$. Additionally, if $k = l = 0$ we put $\sum_{j=1}^0 E_j x(t - jh) = 0$. Following [15] and [9] let us introduce the sequences of matrices

$$\begin{aligned} \Phi_0 &= [E \ 0_{n \times n} \ \dots \ 0_{n \times n}] \\ \Phi_1 &= G\Phi_0 + [E_0 \ E_1 \ \dots \ E_l] \\ \Phi_2 &= G\Phi_1 + [E_1 \ E_2 \ \dots \ E_{l+1}] \end{aligned}$$

and for $k \geq 2$

$$\Phi_{k+1} = G\Phi_k + \sum_{j=1}^{k-1} E_j\Phi_{k-j} + [E_k \ E_{k+1} \ \dots \ E_{k+l}].$$

Theorem 1. *Let $l \in \mathbb{N}_0$, $h > 0$ and $\varphi : \mathbb{R} \rightarrow \mathbb{R}^n$. The solution to the dynamics equation of system $\Sigma_{\varphi,l}$ corresponding to the function φ is given by formula*

$$Ex(t_0 + kh) = \Phi_k \tilde{x}(t_0)$$

where $\tilde{x}(t_0) = M(l, t_0, \varphi, h)$ is the extended initial state.

Proof. Proof by the mathematical induction with respect to k is the same as the proof of the similar result in the case when $E = I$ in [15].

If $E \in \mathbb{C}^{n \times n}$ the range of E is denoted by $\mathcal{R}(E)$, and the null space of E , $\{x : Ex = 0\}$, by $\mathcal{N}(E)$. Recall that $\dim \mathcal{R}(E) + \dim \mathcal{N}(E) = n$. The index of E , denoted by $Ind(E)$, is defined as the least nonnegative integer ν such that $\mathcal{N}(E^\nu) = \mathcal{N}(E^{\nu+1})$. It is known (see [4]) that if (E, G) with $Ind(E) = \nu$, $\dim \mathcal{R}(E)^\nu = s_1$ and $\dim \mathcal{N}(E^\nu) = s_2$, is the regular pair then there exists a nonsingular matrix T such that

$$T^{-1}ET = \begin{bmatrix} C & 0 \\ 0 & N \end{bmatrix}, \quad T^{-1}GT = I - \lambda E = \begin{bmatrix} I - \lambda C & 0 \\ 0 & I - \lambda N \end{bmatrix}$$

where C is $s_1 \times s_1$ nonsingular matrix and N is $s_2 \times s_2$ - nilpotent matrix with $\nu = Ind(N)$. Letting $x := T \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$, $z_1 \in \mathbb{R}^{s_1 \times 1}$, $z_2 \in \mathbb{R}^{s_2 \times 1}$ and $s_1 + s_2 = n$, equation (7) can be decomposed as follows

$$Cz_1(t+h) = (\lambda C - I)z_1(t) - \sum_{j=1}^{k+l} a_{j+1}^{(\alpha)} h^\alpha Cz_1(t-jh)$$

$$Nz_2(t+h) = (\lambda N - I)z_2(t) - \sum_{j=1}^{k+l} a_{j+1}^{(\alpha)} h^\alpha Nz_2(t-jh)$$

Recall that (see for example [4]) for any matrix $E \in \mathbb{R}^{n \times n}$ (not necessary nonsingular) there exists a unique matrix *Drazin inverse* of E denoted as E^D , i.e. matrix E^D such that $E^D E = E E^D$, $E^D E E^D = E^D$ and $E^{\nu+1} E^D = E^\nu$ where $\nu = Ind(E)$. The Driazin inverse is unique and given by

$$E^D = T \begin{pmatrix} C^{-1} & 0 \\ 0 & 0 \end{pmatrix} T^{-1}.$$

Let $\hat{E} = (\lambda E - G)^{-1}E$, $\hat{G} = (\lambda E - G)^{-1}G$ and $\hat{E}_j = (\lambda E - G)^{-1}E_j$.

Theorem 2. *i.) Equation (7) with memory $M(s_1, t_0, \varphi, h)$ given by (3) has the unique solution if and only if there exist a scalar $\lambda \in \mathbb{C}$ such that $\det(\lambda E - G) \neq 0$.*

ii.) This solution is given by

$$\psi(k, M(s_1, t_0, \varphi, h)) = x(t_0 + kh) = \Psi_k \tilde{x}(t_0) \tag{8}$$

where $\tilde{x}(t_0) = M(l, t_0, \phi, h)$ and

$$\begin{aligned} \Psi_0 &= [\hat{E}^D \hat{E} \ 0_{n \times n} \ \dots \ 0_{n \times n}] \\ \Psi_1 &= \hat{G} \Psi_0 + [\hat{E}_0^D \hat{E}_0 \ \hat{E}_1^D \hat{E}_1 \ \dots \ \hat{E}_l^D \hat{E}_l] \\ \Psi_2 &= \hat{G} \Psi_1 + [\hat{E}_1^D \hat{E}_1 \ \hat{E}_2^D \hat{E}_2 \ \dots \ \hat{E}_{l+1}^D \hat{E}_{l+1}] \end{aligned} \tag{9}$$

for $k \geq 2$

$$\Psi_{k+1} = \hat{G} \Psi_k + \sum_{j=1}^{k-l} \hat{E}_j^D \hat{E}_j \Psi_{k-j} + [\hat{E}_k^D \hat{E}_k \ \hat{E}_{k+1}^D \hat{E}_{k+1} \ \dots \ \hat{E}_{k+l}^D \hat{E}_{k+l}]. \tag{10}$$

iii.) The memory $M(s_1, t_0, \varphi, h)$ is a consistent l -memory for dynamics (7) of the system $\Sigma_{(\phi, l)}$ if and only if $M(s_1, t_0, \varphi, h) = \hat{E}^D \hat{E}(M(s_1, t_0, \varphi, h))$, i.e. $M(s_1, t_0, \varphi, h) \in \mathcal{R}((\hat{E}^D)^k) = \mathcal{R}(\hat{E}^D \hat{E})$ where $k = \text{Ind}(\hat{E}^D)$.

Proof. i.) " \Leftarrow " The proof is the same as the one given in [4] for trackable system. " \Rightarrow " Thesis is obtain as the consequence of Theorem 1 and the similar reasoning as the proof of close result given in [4] for trackable system.

ii.) The fact is a direct consequence of the Driazin inverse of matrix E and Theorem 1.

iii.) The proof is a direct consequence of the similar fact given in [4] for trackable system and follows from the fact that Driazin inverse of E and formula for trajectories of nonsingular systems with l -memory given in [15]. \square

Later on the l -memory of the system $\Sigma_{(\varphi, l)}$ given by (3) later (for simplicity of notation) will be denoted shortly by M . The solution of system (7) given by (8) defines the trajectory of this system.

Example 1. Let $E = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}$ and $A = \begin{bmatrix} 0 & \frac{1}{2} \\ 1 & 1 \end{bmatrix}$. For simplicity, let us take $h = 0, 25$ and $\alpha = 0, 5$. Then for $\lambda = 1$ we have $\hat{E} = \frac{4}{3} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$ and $\hat{E}^D = \frac{3}{4} \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}$. Since $G = \frac{1}{4} \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ then also $\hat{G} = \frac{1}{3} \begin{bmatrix} 1 & 4 \\ 0 & -3 \end{bmatrix}$. Moreover, $\hat{E}_0 = -\frac{1}{3} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$, $\hat{E}_1 = \frac{1}{6} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$ and $\hat{E}_0^D = \begin{bmatrix} -3 & -3 \\ 0 & 0 \end{bmatrix}$, $\hat{E}_1^D = \begin{bmatrix} 6 & 6 \\ 0 & 0 \end{bmatrix}$ and so on.

Using formulas (9)-(10) we can do calculations recursively. In this case we obtain $\Psi_0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$, $\Psi_1 = \begin{bmatrix} \frac{4}{3} & \frac{4}{3} & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ and so on. Taking $l = 1$ we need to start memory in four dimensional space for $\tilde{x}(t_0)$. Again, for simplicity, let us take $\tilde{x}(t_0) = [0 \ 1 \ 1 \ 1]^T$. Note that the initial state is $[0 \ 1]^T$, while from the memory we have $[1 \ 1]^T$. It is easy to see that $\psi(1, M) = [3, 33 \ 0]^T$. Similarly we can get $\psi(2, M) = [12, 33 \ 0]^T$ and next values for points lying on trajectory of the considered system.

4 Lyapunov Stability

Observe that $x = 0$ is an equilibrium state of system (7), so also on (4)-(5).

Definition 5. *The equilibrium of system (7) is*

- i.) *stable if for any $\varepsilon > 0$ and $k \in \mathbb{N}_0$ there exists $\delta = \delta(\varepsilon) > 0$ such that if $M \in D^\delta$ then $\psi(k, M) \in D^\varepsilon$.*
- ii.) *asymptotically if it is stable and for any $k \in \mathbb{N}_0$ there exists $\delta > 0$ such that if $M \in D^\delta$ then $\lim_{k \rightarrow \infty} \psi(k, M) = 0$.*

Let us recall that a continuous function $\phi : [0, \rho] \rightarrow [0, \infty)$ is said to *belong to class \mathcal{K}* (or *be class- \mathcal{K} function*) if $\phi(0) = 0$ and ϕ is strictly increasing.

A real valued functional $V : D \rightarrow \mathbb{R}$ is said to be *positive definite* if and only if $V(0) = 0$ and there exists $\alpha \in \mathcal{K}$ such that $\alpha(\|\psi(k, M)\|_D) \leq V(\psi(M))$. V is *decreascent* if and only if $V(0) = 0$ and there exists $\beta \in \mathcal{K}$ such that $V(\psi(k, M)) \leq \beta(\|\psi(k, M)\|_D)$.

Let $\bar{V} : \mathbb{N}_0 \rightarrow \mathbb{R}$ and for $k \in \mathbb{N}_0$ we define

$$\bar{V}(k) := V(\psi(k, M)) \quad \text{and} \quad \Delta \bar{V}(k) := \bar{V}(k + 1) - \bar{V}(k).$$

The idea results presented in Theorems 3 and 4 and theirs proofs follows from ideas result given in [20] for fractional difference systems with two orders and in [19] for linear discrete time systems delays.

Theorem 3. *If there exist a continuous positive defined and decreascent functional $V : D \rightarrow \mathbb{R}_+$ and functions α, β of the class \mathcal{K} such that $\Delta \bar{V}(k) \leq 0$ for all $k \in \mathbb{N}_0$, then the equilibrium of the system (4)-(5) is stable.*

Proof. Let $\psi(k, M)$ be solution of (7). Since V is positive definite and decreasing, there exist functions $\alpha, \beta \in \mathcal{K}$ such that

$$\alpha(\|\psi(k, M)\|_D) \leq V(\psi(k, M)) \leq \beta(\|\psi(k, M)\|_D)$$

for all $k \in \mathbb{N}_0$.

Let $\varepsilon > 0$. One can choose $\delta = \delta(\varepsilon)$ such that $\alpha(\delta) < \beta(\varepsilon)$. Since V is positive definite, then $\alpha(\|\psi(k, M)\|_D) \leq V(\psi(k, M))$ where $M \in D^\delta$. Similarly as in [20], from assumption we conclude that since $V(\psi(k + 1, M)) - V(\psi(0, M)) = \sum_{j=0}^k \bar{V}(j) \leq 0$ and $\alpha \in \mathcal{K}$, then

$$\alpha(\|\psi(k, M)\|_D) \leq V(\psi(0, M)) \leq \alpha(\|\psi(0, M)\|_D) \leq \alpha(\delta) < \beta(\varepsilon)$$

and $(\|\psi(k, M)\|_D) < \varepsilon$ for any $k \in \mathbb{N}_0$. Hence $\psi(k, M) \in D^\varepsilon$. \square

Theorem 4. *If there exist a continuous positive defined and decreascent functional $V : D \rightarrow \mathbb{R}_+$ and a function μ of the class \mathcal{K} such that*

$$(\Delta \bar{V}(k) \leq -\mu(\|\psi(k, M)\|_D))$$

for all $k \in \mathbb{N}_0$ and $M \in D$ satisfying (7), then the equilibrium of the system (4)-(5) is asymptotically stable.

Proof. Since conditions of Theorem 3 are fulfilled then the equilibrium of (4)-(5) is stable.

Using the same reasoning as in [20] one can show that if $\varepsilon > 0$ and $M \in D^\varepsilon$ that stability of the system implies that one can choose ε_0 and $\delta_0 = \delta(\varepsilon_0) > 0$ such that $\beta(\delta_0) < \mu(\varepsilon_0)$. Assuming that $M \in D^{\delta_0}$ we have

$$V(\psi(k+1, M)) - V(\psi(0, M)) = \sum_{j=0}^k \bar{V}(j) \leq - \sum_{j=0}^k \mu(\|\psi(j, M)\|_D).$$

and positivity definite of V :

$$\sum_{j=0}^k \mu(\|\psi(j, M)\|_D) \leq V(\psi(0, M)).$$

if $\mu \in \mathcal{K}$ then $\mu(\|\psi(k, M)\|_D) \leq V(\psi(0, M))$ for $k \in \mathbb{N}_0$ and since V is decreasing and $M \in D^\delta$, then $\mu(\|\psi(k, M)\|_D) \leq V(\psi(0, M)) \leq \alpha(\|M\|_D) < \alpha(\delta_0)$ for all $k \in \mathbb{N}_0$. In consequence $\|\psi(k, M)\|_D < \varepsilon_0$. So, $\psi(k, M) \in D^{\varepsilon_0}$. \square

5 Conclusions

In the work discrete-time descriptor linear fractional order systems with l -memory (initialization) have been considered. The l -memory is the vector that contains information about the previous values of the system. On the other hand it can be consider as the extended initial state. For this type of systems, basing on result given for discrete-time linear fractional order control systems with initialization in [15], the formula for trajectories was given. Next, basing on [20] and [19], asymptotic stability in Lyapunov-Krasovskii sense of considered systems was touch. Since the Grünwald-Letnikov-type fractional h -difference operator can be expressed by the Riemann-Liouville-type fractional h -difference operator (see [8]) systems presented in this work with these types of operators can be studied simultaneously.

The obtain results are only the first steps in studding of linear descriptor fractional order systems with initialization. Interesting results for fractional order descriptor systems without initialization and for $h = 1$ can be found for example in [13, 14] and for discrete-time integer order descriptor linear systems in [5].

Acknowledgments

The work is supported by University Work No. S/WM/1/2016 of Bialystok University of Technology, by Polish Ministry of Science and Higher Education (MNiSW).

References

1. B.Bandyopadhyay, S.Kamal, Stabilization and control of fractional order systems: a sliding mode approach, Lecture Notes in Electrical Engineering 317^o, Springer International Publishing, 55-90 (2015).
2. Bastos N.R.O., Ferreira R.A.C., Torres D.F.M., Necessary optimality conditions for fractional difference problems of the calculus of variations, *Discrete Contin. Dyn. Syst.*, 29(2) (2011), 417–437.
3. M. Busłowicz. Robust stability of positive discrete-time linear systems of fractional order, *Bull. Pol. Acad. Sci. Tech. Sci.*, 58 (4), 567–572 (2010).
4. S.L.Campbell, Singular systems of differential equations, *Research Notes in Mathematics*, Pitman Publishing (1980).
5. D.I.J.Debeljkovic, L.M.Buzurovic, G.V.Simeunovic, Stability of linear discrete descriptor systems in the sense of Lyapunov, *International Journal of Information and Systems Sciences*, 7(4), 303-322 (2011).
6. M.Du, Z.Wang, Correcting the initialization of models with fractional derivatives via history-depend conditions, *Acta Mech. Sin.*, 320-325 (2016).
7. R.A.C.Ferreira, D.F.M.Torres, Fractional h-difference equations arising from the calculus of variations, *Appl. Anal. Discrete Math.*, 5(1) (2011), 110–121.
8. Girejko E., Mozyrska D., Wyrwas M., Comparison of *h*-difference fractional operators, *Advances in the theory and applications of non-integer order systems*, Eds. W. Mitkowski, J. Kacprzyk, and J. Baranowski, LNEE 257, 191–197 (2013),.
9. S.Guermah, M. Bettayeb, S. Djennoune, Controllability and the obserability of lineardiscrete-time fractional order systems, *International Journal of Applied Mathematics and Computer Sciences*, vol.18(2), 213-222 (2008)
10. Graham R.L., Knuth D.E., Patashnik O. Concrete Mathematics: A Fondation for Computer Science. Addison–Wesley (1994).
11. T.T.Hartley, C.F.Lorenzo, Control of initialized fractional-order systems, *NASA/TM-2002-211377/Rev1 Raport*, Glenn Research Center, 1-40 (2002).
12. T. Kaczorek. *Selected problems of fractional systems theory*. Springer, Berlin (2011).
13. T.Kaczorek, Minimum energy control of fractional descriptor positive discrete-time linear systems, *Int.J.Appl.Math.Comput.Sci.* 24(4), 735-743 (2014)
14. T.Kaczorek, Positivity and stability of fractional descriptor time-varying discrete-time linear systems, *Int.J.Appl.Math.Comput.Sci.* 26(1), 5-13 (2016).
15. D.Mozyrska, E.Pawłuszewicz, Fractional discrete-time linear control systems with initialisation, *Int. J. Cont.*, 85 (2), 213-219 (2013).
16. P.Ostalczyk *Ephitome of fractional calculus*, Wyd. Politechnika Łódzka 2008.
17. I. Podlubny, *Fractional differential systems*, Academic Press, San Diego 1999.
18. D. Sierociuk and D. Dzielinski, Fractional Kalman filter algorithm for the states parameters and order of fractional system estimation, *Int. J. Appl. Math. Comp. Sci.*, 16 (1), 129–140 (2006).
19. S.B.Stojanovic, D.L.Debeljkovic, I.Mladenovic. A Lyapunov-Krasovskii methodology for asymptotic stability of discrete time delay systems, *Serbian Journal of Electrical Engineering*, 4(1), 109-117, (2007).
20. M.Wyrwas, E.Girejko, D.Mozyrska, E.Pawłuszewicz, Stability of fractional difference systems with two orders, *Advances in the theory and applications of non-integer order systems*, Eds. W. Mitkowski, J. Kacprzyk, and J. Baranowski, LNEE 257, 41-52, Springer (2015).

Modeling of heat transfer process with the use of non integer order, discrete, transfer function models

Krzysztof Oprzędkiewicz and Edyta Gawin

AGH University, A. Mickiewicza 30,30-59 Krakow, Poland
State Higher Vocational School in Tarnow, A. Mickiewicza 8, 33-100 Tarnow, Poland
kop@agh.edu.pl
e_gawin@pwsztar.edu.pl

Abstract. The paper is intended to present a possibility of modeling a heat transfer process in one dimensional plant with the use of new, non integer order, discrete models. The proposed models have the form of hybrid transfer functions, containing both integer and non integer order parts. The integer order part represents a pure delay, the non integer order part is modeled with the use of CFE approximant. The proposed models were compared to step responses of real experimental plant with the use of typical MSE cost function. Results of experiments show that a good accuracy can be achieved for relatively low order of model.

Keywords: fractional order systems, CFE approximation, heat transfer process

1 An Introduction

One of main areas of application a Fractional Order Calculus in automation is modeling of processes with dynamics hard to describe with the use of another approaches. Fractional order modeling has been considered by many Authors, for example: [1], [2],[4], [7], [9], [10]. Particularly, modeling of heat transfer processes with the use of non integer order approach has been considered among others in papers: [12], [13], [14].

The goal of the paper is to propose new non integer order models of heat transfer process. These models have a form of hybrid, discrete transfer functions $G^+(z^{-1})$, containing both integer order and non integer order parts. Previous investigations run by authors (see for example [11], [12]) point that the hybrid models are more accurate in the sense of MSE (Medium Square Error) cost function than "pure" non integer order models. The integer order part in the proposed models is a "pure" delay, the non integer order part describes the inertia. The proposed, discrete transfer functions are relatively easy to digital implementation (for example at PLC), because the delay is expressed as the multiple of sample time and the non integer order part is approximated with the

use of CFE approximation. This gives reasonable summarized order of the whole model.

The paper is organized as follows: at the beginning any elementary ideas from non integer order calculus are remembered, next proposed hybrid models are given. They are next verified with the use of experimental results. Finally main conclusions are presented.

2 Preliminaries

2.1 Elementary ideas

The presentation of elementary ideas will be started with define a non integer order, integro-differential operator. It is expressed as follows (see for example [6]):

Definition 1. *The non integer order integro - differential operator*

$${}_a D_t^\alpha f(t) = \begin{cases} \frac{d^\alpha f(t)}{dt^\alpha} & \alpha > 0 \\ 1 & \alpha = 0 \\ \int_a^t f(\tau)(d\tau)^{-\alpha} & \alpha < 0 \end{cases} . \quad (1)$$

where a and t denote time limits to operator calculating, $\alpha \in \mathbb{R}$ denotes the non integer order of the operation.

The fractional-order, integro-differential operator (1) can be described by different definitions, given by Grünvald and Letnikov (GL definition), Riemann and Liouville (RL definition) and Caputo (C definition). The digital modeling of FO operator can be most naturally done with the use of GL definition and it will be presented here:

Definition 2. *The Grünvald-Letnikov definition of the FO operator ([2],[15])*

$${}^GL D_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{j=0}^{\lfloor \frac{t}{h} \rfloor} (-1)^j \binom{\alpha}{j} f(t - jh). \quad (2)$$

In (2) $\binom{\alpha}{j}$ denotes the binomial coefficient:

$$\binom{\alpha}{j} = \begin{cases} 1, & j = 0 \\ \frac{\alpha(\alpha-1)\dots(\alpha-j+1)}{j!}, & j > 0 \end{cases} \quad (3)$$

2.2 The CFE approximation

An implementation of operator (1) at each digital platform (PLC, microcontroller) requires us to apply an integer order, finite dimensional, discrete approximant. The most known are PSE (Power Series Expansion) and CFE (Continuous Fraction Expansion). They allow us to estimate a non integer order element with the use of digital FIR or IIR filter. The PSE approximant bases directly on discrete version of GL definition (2) and it has the form of FIR filter containing only zeros. However its digital implementation to keep a good quality requires us to apply long memory buffer (high order of the filter). The CFE approximant has the form of IIR filter containing both poles and zeros. It is faster convergent and easier to implement because its order is relatively low, typically not higher than 5.

The discretization of fractional order element s^α , $\alpha \in \mathbb{R}$ can be done with the use of the so called generating function $s \approx \omega(z^{-1})$. The new operator raised to power α has the following form (see for example [3]):

$$\begin{aligned}
 (\omega(z^{-1}))^\alpha &= \left(\frac{1+a}{h}\right)^\alpha CFE\left\{\left(\frac{1-z^{-1}}{1+az^{-1}}\right)^\alpha\right\}_{M,M} = \\
 &= \frac{P_{\alpha M}(z^{-1})}{Q_{\alpha M}(z^{-1})} = \left(\frac{1+a}{h}\right)^\alpha \frac{CFE_N(z^{-1}, \alpha)}{CFE_{N,D}(z^{-1}, \alpha)} = \frac{\sum_{m=0}^M w_m z^{-m}}{\sum_{m=0}^M v_m z^{-m}}. \tag{4}
 \end{aligned}$$

In (4) a is the coefficient depending on approximation type (for example: $a=0$ for Euler approximation, $a=1$ for Tustin approximation), h denotes the sample time, M is the order of approximation. Numerical values of coefficients w_m and v_m and different values of parameter a can be calculated for example with the use of MATLAB function given by Petras in [16]. This MATLAB function was applied in experiments described in the next section. If the Tustin approximation is considered ($a=1$) then $CFE_{N,D}(z^{-1}, \alpha) = CFE_N(z^{-1}, -\alpha)$ and the polynomial $CFE_{N,D}(z^{-1}, \alpha)$ can be given in the direct form (see [3]). Examples of polynomial $CFE_{N,D}(z^{-1}, \alpha)$ for $M = 1, 3, 5$ are given in table 1.

3 The experimental system

Experiments were executed with the use of the experimental heat plant shown in figure 1. It has the form of a thin copper rod 260[mm] long. The rod is heated with the use of an electric heater of the length $\Delta x_0 = 36[mm]$ localized at one end of rod. An output temperature is measured with the use of Pt100 sensors long $\Delta x = 5[mm]$ located in points: 75[mm], 130[mm] and 190[mm]. The input signal of the system is the standard current signal from range 0 – 20[mA]. It is amplified to the range 0 – 1.5[A] and next it is the input signal for the heater. Signals from the Pt100 sensors are read directly by analog input module in the PLC. Data from PLC are read with the use of SCADA. The whole system is connected via PROFINET. The temperature distribution with respect to both time and length coordinates is shown in the figure 2.

Table 1. Coefficients of CFE polynomials $CFE_{N,D} z^{-1}, \alpha$ for Tustin approximation with respect to [3]

Order M	w_m	v_m
$M=1$	w_1 $-\alpha$	v_1 α
	w_0 1	v_0 1
$M=3$	w_3 $-\frac{\alpha}{3}$	v_3 $\frac{\alpha}{3}$
	w_2 $\frac{\alpha^2}{3}$	v_2 $\frac{\alpha^2}{3}$
	w_1 $-\alpha$	v_1 α
	w_0 1	v_0 1
$M=5$	w_5 $-\frac{\alpha}{5}$	v_5 $\frac{\alpha}{5}$
	w_4 $\frac{\alpha^2}{5}$	v_4 $\frac{\alpha^2}{5}$
	w_3 $-\left(\frac{\alpha}{5} \quad \frac{2\alpha^3}{35}\right)$	v_3 $-\left(\frac{-\alpha}{5} \quad \frac{-2\alpha^3}{35}\right)$
	w_2 $\frac{2\alpha^2}{5}$	v_2 $\frac{2\alpha^2}{5}$
	w_1 $-\alpha$	v_1 α
	w_0 1	v_0 1

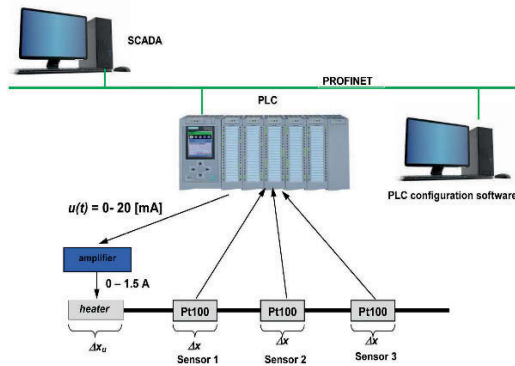


Fig. 1. The experimental system

The fundamental mathematical model describing the heat conduction in the plant is the partial differential equation of the parabolic type with the homogeneous Neumann boundary conditions at the ends, the homogeneous initial condition, the heat exchange along the length of rod and distributed control and observation. This equation with integer orders of both differentiations was presented by many papers, for example in [12]. The previous investigations of Authors show, that non integer order, state space models are more accurate in the sense of MSE cost function (see [13], [14]) that integer order models.

An alternative, simpler approach during modeling the considered plant is to apply transfer function model with delay. In the integer order form it has the

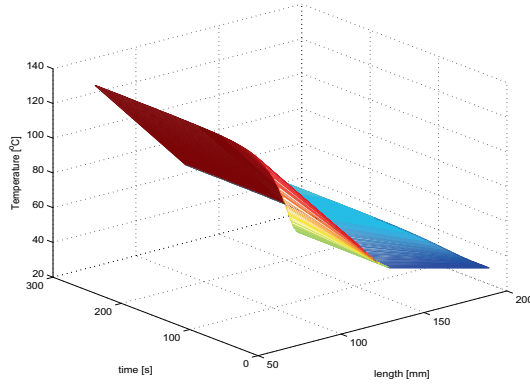


Fig. 2. The spatial-time temperature distribution in the plant

shape of well known Kűpfműller transfer function:

$$G_1(s) = \frac{ke^{-\tau_1 s}}{T_1 s + 1} \tag{5}$$

$$G_2(s) = \frac{ke^{-\tau_2 s}}{(T_{21} s + 1)(T_{22} s + 1)}$$

4 The proposed, discrete, non integer order transfer function models with delay

The transfer function models with delay (5) can be also expressed in FO form:

$$G_{FO1}(s) = \frac{ke^{-\tau_1 s}}{T_\alpha s^\alpha + 1} \tag{6}$$

$$G_{FO2}(s) = \frac{ke^{-\tau_2 s}}{(T_\alpha s^\alpha + 1)(T_\beta s^\beta + 1)} \tag{7}$$

Let us apply the CFE approximant (4) to transfer functions (6) and (7) and assume that the delays τ_1 and τ_2 are multiplies of sample time h . Then we obtain:

$$G_{FO1}^+(z^{-1}) = \frac{z^{-N_1} Q_{\alpha M}(z^{-1})}{T_\alpha P_{\alpha M}(z^{-1}) + Q_{\alpha M}(z^{-1})} \tag{8}$$

$$G_{FO2}^+(z^{-1}) = \frac{z^{-N_2} Q_{\alpha M}(z^{-1}) Q_{\beta M}(z^{-1})}{(T_\alpha P_{\alpha M}(z^{-1}) + Q_{\alpha M}(z^{-1}))(T_\beta P_{\beta M}(z^{-1}) + Q_{\beta M}(z^{-1}))} \tag{9}$$

where P and Q denote numerator and denominator of CFE approximant (4), M is the order of CFE and $\tau = N_{1,2}h$. An important advantage of use CFE is that it requires use a relatively low order M (typically $M < 10$) to obtain a good accuracy. The step responses of the both models can be calculated with the use of MATLAB function *step*.

5 Experimental results

Experiments were done with the use of experimental system shown in the previous section, the discrete time model was examined with the use of typical MSE cost function (10) (see for example [5]):

$$MSE = \frac{1}{K_s} \sum_{k=1}^{K_s} (y(kh) - y^+(kh))^2. \quad (10)$$

where K_s is a number of all collected samples from all sensors, y is the experimental time response measured in discrete time moments kh , y^+ is the time response of the discrete, tested model, calculated along the same time grid. Parameters of the both proposed models calculated via optimization the cost function (10) are given in the tables 2 and 3. The tables 2 and 3 compare the proposed, hybrid models to integer-order models (5) with the use of the same cost function. For all experiments the order of CFE approximation was equal $M = 5$, the sample time was equal $h = 1[s]$. Step responses for plant and each non-integer order model are shown in figures 3 and 4.

Table 2. Optimal parameters and cost function MSE (10) for model (6) and the integer-order models (5)

Sensor No	α ,	T_α ,	N_1 ,	MSE	T_1 ,	N_1 ,	MSE
1	1.2183	38.0751	8	0.0944	29.0422	10	0.1133
2	1.5433	99.1564	16	0.0093	43.6374	23	0.0301
3	1.6967	195.5599	27	0.0033	82.7001	30	0.0182

Table 3. Optimal parameters and cost function MSE (10) for model (7) and the integer-order models (5)

Sensor No	α ,	β ,	T_α	T_β	N_2 ,	MSE	T_{21}	T_{22}	N_2	MSE
1	1.4704	0.7138	24.6555	43.8919	3	0.0434	28.3752	4.4113	6	0.0857
2	1.5703	0.8997	57.8378	48.3222	8	0.0043	40.1456	10.5969	15	0.0089
3	1.7666	1.1558	173.3362	64.1773	10	0.0021	60.6850	18.9781	26	0.0033

From the tables 2, 3 and figures 3, 4 it can be concluded that the proposed models are able to more precisely describe the temperature distribution in the considered experimental plant than known integer order transfer function models with delay.

6 Conclusions

Final conclusions from the paper can be formulated as underneath:

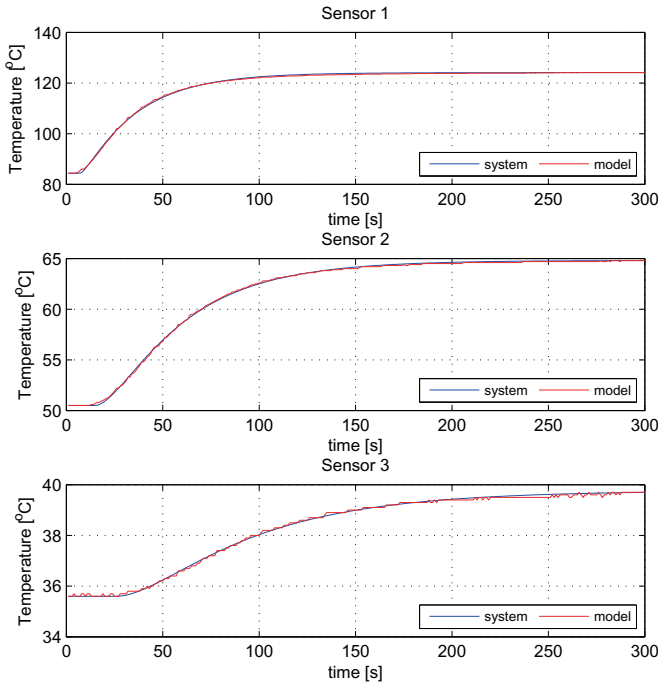


Fig. 3. Step responses of plant and model 1

- The proposed hybrid FO models with delay are more precise in the sense of MSE cost function than "classic" models with delay,
- The main advantage of the proposed models is that their good accuracy can be achieved for relatively low order of model. It is equal a sum of CFE approximant order (equal 5) and order of IO part describing delay as the multiple of the sample time.
- Models discussed in this paper are going to be PLC implemented and next applied to construct Model Based Control systems or Model Based Fault Detection systems.

Acknowledgments. The paper was sponsored by AGH University grant no 11.11.120.815 .

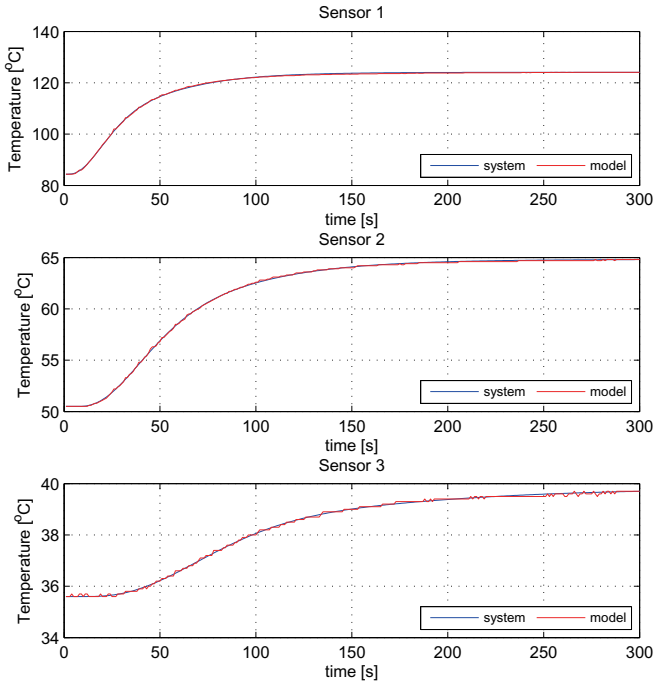


Fig. 4. Step responses of plant and model 2

References

1. Burnecki K. (2012), Identification, validation and prediction of fractional dynamic systems, Oficyna Wydawnicza Politechniki Wrocławskiej Wrocław, pp. 1-125.
2. Caponetto R., Dongola G., Fortuna I., Petras I. (2010), Fractional Order Systems. Modeling and Control Applications, World Scientific Series on Nonlinear Science, Series A, vol. 72, World Scientific Publishing.
3. Chen Y.Q., Moore K.L. (2002) Discretization Schemes for Fractional-Order Differentiators and Integrators, IEEE Transactions on Circuits and Systems - I: Fundamental Theory and Applications, vol. 49, No 3, March 2002
4. Dzielinski, A.; Sierociuk, D.; Sarwas, G. (2010) Some applications of fractional order calculus. Bulletin of the Polish Academy of Sciences: Technical Sciences, 2010, 58.4: 583-592.
5. Isermann, R., Muenchhof, M.(2011) Identification of Dynamic Systems. An Introduction with Applications. Springer, 2011.
6. Kaczorek T.(2011), Selected Problems in Fractional Systems Theory, Springer-Verlag.
7. Klimek M. (2010), Existence-uniqueness result for a certain equation of motion in fractional mechanics. Bulletin of The Polish Academy of Sciences Technical Sciences, Vol. 58, No. 4, 573-581.

8. Mitkowski W., Oprzędkiewicz K. (2007) Optimal sample time estimation for the finite-dimensional discrete dynamic compensator implemented at the soft PLC platform in: 23rd IFIP TC 7 conference on System modelling and optimization : Cracow, Poland, July 23-27, 2007 : book of abstracts, eds. Adam Korytowski, Wojciech Mitkowski, Maciej Szymkat ; AGH University of Science and Technology. Faculty of Electrical Engineering, Automatics, Computer Science and Electronics. Krakow, pp. 77-78.
9. Mitkowski W., Kacprzyk J., Baranowski J., Editors, (2013), *Advances in the Theory and Applications of Non-integer Order Systems*, 5th Conference on Non-integer Order Calculus and Its Applications, Cracow, Poland. *Lecture Notes in Electrical Engineering* 257, Springer, pp. 1-325.
10. Mitkowski W., Skruch P. (2013), Fractional-order models of the supercapacitors in the form of RC ladder networks. *Bulletin of The Polish Academy of Sciences Technical Sciences*, Vol. 61, No. 3, pp. 581-587.
11. Oprzędkiewicz K (2012) A Strejc model-based, semifractional (SSF) transfer function model, *Automatics* ; AGH UST 2012 vol. 16 no. 2, pp. 145-154. link to the full text: <http://journals.bg.agh.edu.pl/AUTOMAT/2012.16.2/automat.2012.16.2.145.pdf>
12. Oprzędkiewicz K., Mitkowski W., Gawin E. (2015) Application of fractional order transfer functions to modeling of high order systems. *MMAR 2015 : 20th international conference on Methods and Models in Automation and Robotics : 24-27 August 2015, Międzyzdroje, Poland* : program, abstracts, proceedings (CD). Szczecin : ZAPOL Sobczyk Sp. z o.o., [2015] + CD - full text CD: pp. 1169-1174.
13. Oprzędkiewicz K., Mitkowski W., Gawin E. (2016), Parameter identification for non integer order, state space models of heat plant In: *MMAR 2016 : 21th international conference on Methods and Models in Automation and Robotics : 29 August-1 September 2016, Międzyzdroje, Poland*, pp. 184-188.
14. Oprzędkiewicz K., Gawin E. (2016) Non integer order, state space model for one dimensional heat transfer process, *Archives of Control Sciences*, 2016 vol. 26 no. 2, pp. 261-275. full text available at: <https://www-1degruyter-com-1atoz.wbg2.bg.agh.edu.pl/downloadpdf/j/acsc.2016.26.issue-2/acsc-2016-0015/acsc-2016-0015.xml>.
15. Ostalczyk P. (2012) Equivalent descriptions of a discrete-time fractional-order linear system and its stability domains, *International Journal of Applied Mathematics and Computer Science* 22(3), pp. 533-538.
16. Petras I: <http://people.tuke.sk/igor.podlubny/USU/matlab/petras/dfod1.m>

Human arm fractional dynamics

Artur Babiarz and Adrian Łęgowski

Institute of Automatic Control
Silesian University of Technology
16 Akademicka St., 44-100 Gliwice, Poland
{artur.babiarz, adrian.legowski}@polsl.pl

Abstract. In this paper the fractional model of the human arm is presented. Proposed approach is an attempt to consider the muscle damping properties in the simplest form possible. As the base for our simulations the equation of motion based on Lagrange formalism was used. In order to obtain fractional properties we propose using fractional-order derivatives instead of integer-ones. This simplification creates opportunity for an easy implementation and comparison with commonly used models. The core of the presented method is solving this nonlinear equation. The preliminary results was shown. The simplest model possible was analyzed. We considered only two DOF (degrees of freedom) planar model without joint limitations and properly distributed masses. This experiment is to show the damping properties of the fractional model.

Keywords: Fractional Calculus, Human arm dynamics, Fractional equation of motion

1 Introduction

Modern science with use of many mathematics tools is able to describe many processes and objects with very good accuracy. However, it appears that some of them are much more difficult to describe with standard approach. That means it requires using very complicated and advanced tools. This causes the lost of the simplicity of the idea. According to many researches, proper approach (i.e. with use of FC - fractional calculus) allows to describe very complicated dynamics with use of very simple idea (which does not mean simple equations or implementation). One of these objects are human limbs.

Human limbs, from kinematic point of view, are very often considered as more or less complex manipulators [6, 14], therefore one can use direct and inverse kinematics techniques to describe the geometry, reachable space and their position.

Dynamical Models of the human arm have been widely described in the literature [12, 17, 16, 5]. It has been noticed, that the dynamics of such object is not easy to describe. First, it is very important to consider the fact, that the shape of the object is changeable. It depends on the rotation in an elbow, on the contraction of muscles etc. These effects may be addressed by various means.

It has been proposed to utilize the switched linear models like those in [2, 4, 3]. Presented work assumes that human arm can be modeled as a two link planar system. It is clear that it is simplification, however the concept stands and proves to be relatively simple to describe and implement.

It has been noted that the human arm has viscoelastic properties [10, 13, 11]. These properties makes accurate modeling a hard task for well known simple approaches. It has been addressed in many ways. One of them is non-integer order derivative.

2 Fractional calculus

The idea of non-integer order derivative and integral is nearly as old as well known integer-order calculus. It goes back to the 1695 and Leibniz's letter to L'Hospital [18]. There are many definitions of fractional derivatives and integrals. In this paper we focus on one definition that can be easily implemented.

The Riemann-Liouville (RL) derivative is defined as follows [22]:

$${}_0D_t^\alpha y(t) = \frac{1}{\Gamma(1-\alpha)} \frac{d}{dt} \int_0^t (t-\tau)^{-\alpha} y(\tau) d\tau. \quad (1)$$

The function $\Gamma()$ is the Gamma function. We assumed that derivative order $\alpha \leq 1$ and $\alpha > 0$.

Currently researchers are looking for new applications of fractional calculus (FC) in various branches of science. Many researchers proved that the FC can be applied in control theory in order to design new type of controllers [7, 15]. In paper [1] some fractional continuous models have been studied.

What is most important for here presented studies is the fact, that FC proved to be an useful tool in modeling human limbs. In papers [20, 21, 19] this concept has been evaluated with experimental data. It appears that complex human arm dynamics can be described in ideologically simple way with use of non-integer order calculus. For now, let us omit the issues with implementation of FC, however it is sufficient to say, that it does not require vast processing power.

These and many others applications prove the usability of fractional calculus and therefore, need for finding new applications in order to improve solutions for well known problems.

3 Model assumptions

In this paper we assume that our model consists of two rotational joints. It has been in details described in paper [2]. Moreover, the state-space equations also have been presented in that paper. In this study we assume only vertical motion in order to evaluate some properties of proposed approach. This simplification makes the model of human arm equal to the model of rigid two link planar manipulator. We assume that whole mass is located in the geometrical center of the joint. There is no damping in the system, at least not in the obvious form. Also, we do

not assume limitations for joint motions. All these properties will be considered in the future study of the problem. The implication of such simplifications is -

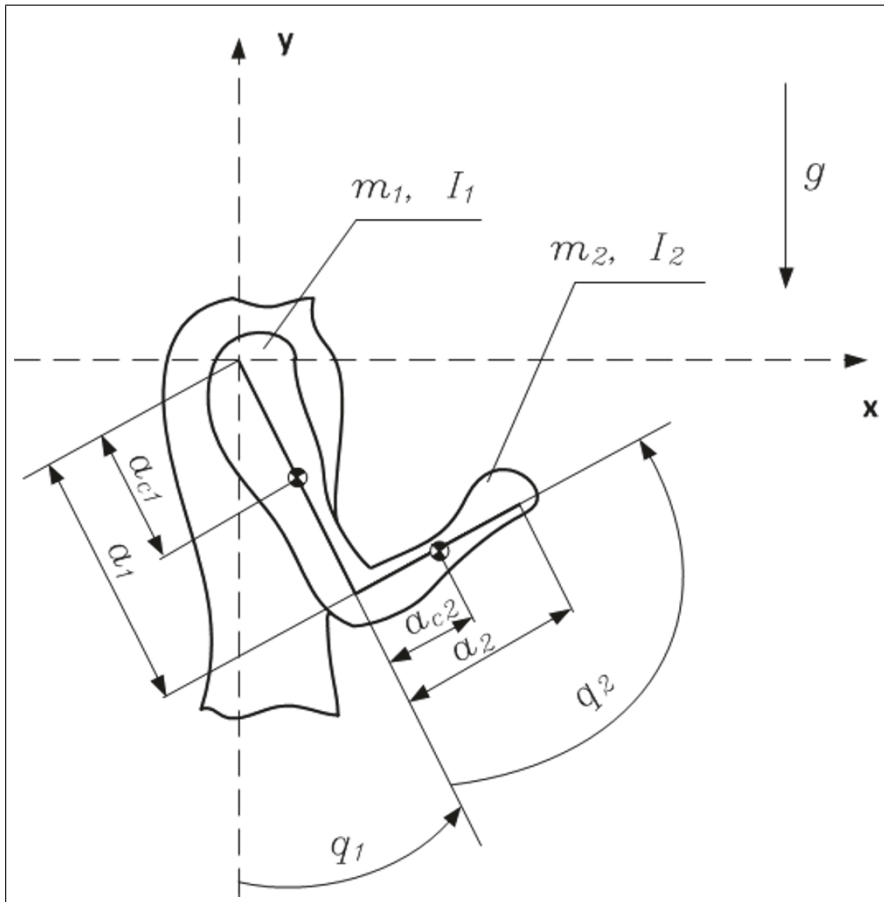


Fig. 1. Simplified model of the human arm (source:[2])

for integer-order model - total lack of damping. Because of that, the response will be in the form of undamped oscillations. This is expected and well known fact for this kind of structures.

The origin of our study is the equation that is the result of Euler-Lagrange description. It takes the form as follows:

$$M(q) \frac{d^2 q}{dt^2} + C(q, \frac{dq}{dt}) \frac{dq}{dt} + G(q) = \tau, \quad (2)$$

where: $M(q)$ - is the inertia matrix, $C(q, \frac{dq}{dt})$ is the Coriolis and centrifugal forces matrix, $G(q)$ is the matrix representing gravity forces. The vector τ represents

the forces and moments of the drives. In our case these will be only two torques. These matrices have the following forms:

$$M(q) = \begin{bmatrix} d_1 & d_3 \cos(q_1 - q_2) \\ d_3 \cos(q_1 - q_2) & d_2 \end{bmatrix},$$

$$C(q, \dot{q}) = \begin{bmatrix} 0 & d_3 \sin(q_1 - q_2) \dot{q}_2 \\ -d_3 \sin(q_1 - q_2) \dot{q}_1 & 0 \end{bmatrix},$$

$$G(q) = \begin{bmatrix} -d_4 g \sin q_1 \\ -d_5 g \sin q_2 \end{bmatrix},$$

where

$$d_1 = m_1 a_{c1}^2 + m_2 a_1^2 + I_1,$$

$$d_2 = m_2 a_{c2}^2 + I_2,$$

$$d_3 = m_2 a_1 a_{c2},$$

$$d_4 = m_1 a_{c1} + m_2 a_2,$$

$$d_5 = m_2 a_{c2}$$

and m_i - is the mass of i -th joint, a_i - is the i -th link length, a_{ci} - is the distance from the i -th joint(its coordinate system) to the center of (its)mass, I_i - is the i -th joint moment of inertia.

Solving the nonlinear differential equation (2) allows to compute the response for given torques and obtaining the functions describing the inner coordinates of the system.

4 Fractional model

In literature it has been stated that the fractional oscillatory models offers very strong damping [9]. The same effect may be noticed in fractional manipulator system (and general pendulum systems) presented in paper [8].

Because of that reason we decided to utilize simplified approach. Because of our simplification it is very easy to implement in any environment.

Our proposition is to change the derivatives order in equation (2) to non-integer orders. As the result of such action we obtain the following equation:

$$M(q)_0 D_t^{1+\alpha} + C(q,0)_0 D_t^\alpha + G(q) = \tau, \quad (3)$$

where: α is the order of fractional derivative. It's main influence should be on damping properties of the system.

With use of this method we intend to include complex human arm dumping properties. We assume that as other fractional oscillatory systems, fractional model of human arm will be strongly damped. For $\alpha = 1$ we obtain standard, integer-order approach.

5 Simulation

The equation (3) is solved numerically. We try to find the following function:

$$q = f(t, \tau(t)) \quad (4)$$

that satisfies the equation (3). In our studies we used Oustaloup's approximation. As for now, we have not used the experimental data in our study. These are only simulation results.

In order to evaluate proposed method we simulated 40 seconds of motion of the human arm. The simulation starts with no torque applied. In 10th second we apply torque to the first joint and in 30th second to the second joint. The responses are compared for various derivation order α in order to expose the influence of this parameter on the trajectory of joint values. We compared responses for $\alpha = 0.1, 0.5, 0.8, 0.99, 1$.

All results are presented in figures, that allows for comparison of these two models.

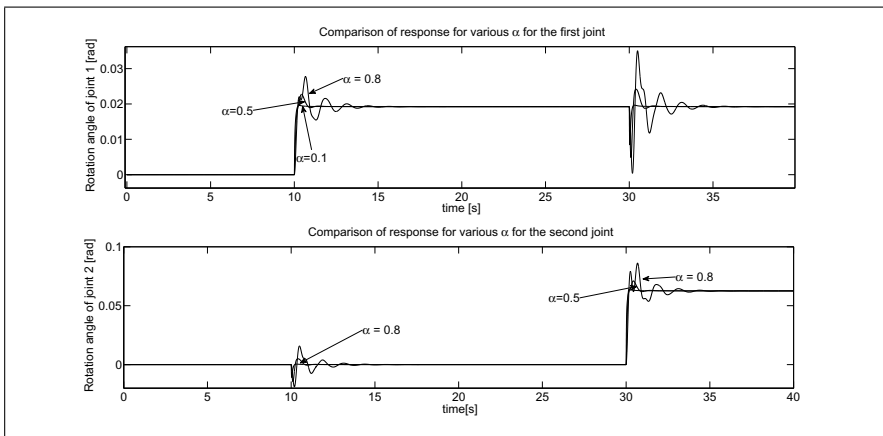


Fig. 2. Comparison for $\alpha = 0.1, 0.5, 0.8$

As can be seen in the Figures 2 and 3, the fractional system seems to have very strong dumping for small values of α . The effect becomes weaker when α rises and finally disappears for $\alpha = 1$. This conclusion is similar to ones from literature.

In order to compare fractional system which is naturally damped we added friction to the integer-order system. It can be seen that differences are present, yet some similarities can be observed in the Figure 4. The responses are similar if it comes to values, yet different considering the shape. We conclude that the fractional system's damping may be different from simple friction.

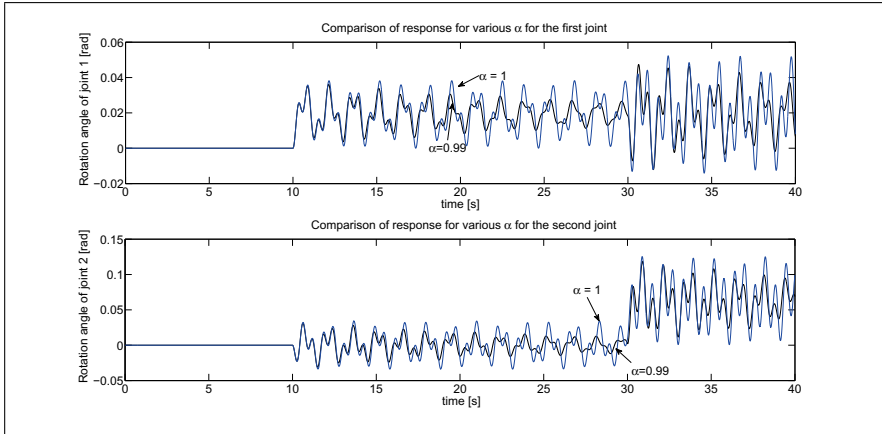


Fig. 3. Comparison for $\alpha = 0.99, 1$

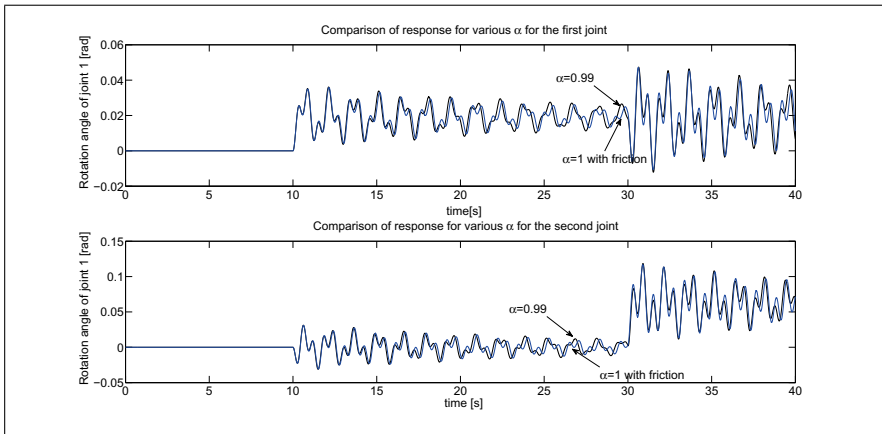


Fig. 4. Comparison of integer-order system with friction and fractional system

6 Conclusion and future work

In presented study the nonlinear fractional differential equation has been considered. Proposed model of the human arm is simplification of general approach yet allows to evaluate some properties. It has been shown that for $\alpha = 1$ proposed model becomes well known integer-order system that has been studied for many years.

We conclude that proposed human arm model has "natural" damping which nature is yet to be discovered.

As the future work, we intend to evaluate obtained results with real viscoelastic manipulator that can imitate the human arm. Moreover, it is clear that presented here models and assumptions are simplification of the real system. We intend to create model in 3 dimensional space with respect of joint variables

limitations. We conclude that utilizing the concept of switched dynamics may be of use for accurate shape modeling.

Acknowledgments. The research presented here was done as a part of the project funded by the National Science Centre in Poland granted according to decision DEC-2014/13/B/ST7/00755 (A.L.) and the Silesian University of Technology research grants BK-204/RAu1/2017 (A.B.). The calculations were performed with the use of IT infrastructure of GeCONiI Upper Silesian Centre for Computational Science and Engineering (NCBiR grant no POIG.02.03.01-24-099/13).

References

1. Aoun, M., Malti, R., Levron, F., Oustaloup, A.: Numerical simulations of fractional systems: An overview of existing methods and improvements. *Nonlinear Dynamics* **38**(1), 117–131 (2004). DOI 10.1007/s11071-004-3750-z
2. Babiarz, A.: On mathematical modelling of the human arm using switched linear system. In: *AIP Conference Proceedings*, vol. 1637, pp. 47–54 (2014)
3. Babiarz, A., Czornik, A., Niezabitowski, M., Zawiski, R.: Mathematical model of a human leg: The switched linear system approach. In: *Pervasive and Embedded Computing and Communication Systems (PECCS), 2015 International Conference on*, pp. 1–8. IEEE (2015)
4. Babiarz, A., Klamka, J., Zawiski, R., Niezabitowski, M.: An approach to observability analysis and estimation of human arm model. In: *11th IEEE International Conference on Control & Automation (ICCA)*, pp. 947–952. IEEE (2014)
5. Biess, A., Flash, T., Liebermann, D.G.: Riemannian geometric approach to human arm dynamics, movement optimization, and invariance. *Physical Review E* **83**(3), 031,927 (2011)
6. Biryukova, E., Roby-Brami, A., Frolov, A., Mokhtari, M.: Kinematics of human arm reconstructed from spatial tracking system recordings. *Journal of biomechanics* **33**(8), 985–995 (2000)
7. Cao, J.Y., Cao, B.G.: Design of fractional order controllers based on particle swarm optimization. In: *2006 1ST IEEE Conference on Industrial Electronics and Applications*, pp. 1–6 (2006). DOI 10.1109/ICIEA.2006.257091
8. David, S., Balthazar, J.M., Julio, B., Oliveira, C.: The fractional-nonlinear robotic manipulator: Modeling and dynamic simulations. In: *AIP Conference Proceedings*, pp. 298–305 (2012)
9. David, S.A., Valentim, C.A.: Fractional euler-lagrange equations applied to oscillatory systems. *Mathematics* **3**(2), 258–272 (2015)
10. Frolov, A.A., Prokopenko, R., Dufosse, M., Ouezdou, F.B.: Adjustment of the human arm viscoelastic properties to the direction of reaching. *Biological cybernetics* **94**(2), 97–109 (2006)
11. Gomi, H., Osu, R.: Task-dependent viscoelasticity of human multijoint arm and its spatial characteristics for interaction with environments. *The Journal of Neuroscience* **18**(21), 8965–8978 (1998)
12. Van der Helm, F.C., Schouten, A.C., de Vlugt, E., Brouwn, G.G.: Identification of intrinsic and reflexive components of human arm dynamics during postural control. *Journal of neuroscience methods* **119**(1), 1–14 (2002)

13. Kubo, K., Kanehisa, H., Kawakami, Y., Fukunaga, T.: Influence of static stretching on viscoelastic properties of human tendon structures in vivo. *Journal of applied physiology* **90**(2), 520–527 (2001)
14. Lenarcic, J., Umek, A.: Simple model of human arm reachable workspace. *IEEE transactions on systems, man, and cybernetics* **24**(8), 1239–1246 (1994)
15. Mackowski, M., Grzejszczak, T., Legowski, A.: An approach to control of human leg switched dynamics. In: 2015 20th International Conference on Control Systems and Computer Science (CSCS), pp. 133–140 (2015). DOI 10.1109/CSCS.2015.67
16. Mobasser, F., Hashtrudi-Zaad, K.: A method for online estimation of human arm dynamics. In: Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE, pp. 2412–2416. IEEE (2006)
17. Rosen, J., Perry, J.C., Manning, N., Burns, S., Hannaford, B.: The human arm kinematics and dynamics during daily activities-toward a 7 dof upper limb powered exoskeleton. In: ICAR'05. Proceedings., 12th International Conference on Advanced Robotics, 2005., pp. 532–539. IEEE (2005)
18. Ross, B.: Fractional Calculus and Its Applications: Proceedings of the International Conference Held at the University of New Haven, June 1974, chap. A brief history and exposition of the fundamental theory of fractional calculus, pp. 1–36. Springer Berlin Heidelberg, Berlin, Heidelberg (1975). DOI 10.1007/BFb0067096
19. Tejado, I., Valério, D., Pires, P., Martins, J.: Fractional models for the human arm (2013)
20. Tejado, I., Valério, D., Pires, P., Martins, J.: Fractional order human arm dynamics with variability analyses. *Mechatronics* **23**(7), 805–812 (2013)
21. Ventura, A., Tejado, I., Valério, D., Martins, J.: Fractional direct and inverse models of the dynamics of a human arm. *Journal of Vibration and Control* p. 1077546315580471 (2015)
22. Vinagre, B., Podlubny, I., Hernandez, A., Feliu, V.: Some approximations of fractional order operators used in control theory and applications. *Fractional calculus and applied analysis* **3**(3), 231–248 (2000)

Approximation and stability analysis of some kinds of switched fractional linear systems

Stefan Domek

West Pomeranian University of Technology, Szczecin
ul. Sikorskiego 37, 70-313 Szczecin

stefan.domek@zut.edu.pl

Abstract. Many systems appearing in practice exhibit a hybrid nature, that is, a coupling between continuous dynamics and discrete events. Special cases of such systems are those in which the linear subsystems are switched according to time or according to state and/or input. In the paper a method for approximation of some kinds of switched fractional linear systems is proposed and their stability is analyzed.

Keywords: switched systems, fractional systems, integer order state approximation

1 Introduction

The quality of control processes depends largely on the quality of the model employed for the synthesis and tuning of the controller. This model should be easy to determine in terms of both the structure and parameters. At the same time it should allow an effective control algorithm to be synthesized in an easy way. This is particularly important in the case of nonlinear processes for which the use of models linearized around the operating point for the synthesis of the control algorithm is insufficient and the selection of an adequate non-linear model and its parameters setting is very difficult.

One of the most effective and yet conceptually understandable methods to capture real properties of many nonlinear industrial processes seems to be replacing the nonlinear description by a set of linear submodels. In such a case, the local submodel is used to determine the local control signal, and the whole idea boils down to the switching of active submodels or active subcontrollers as a function of time. Such models, called switched systems, have been subject of intensive research for several decades [7, 9, 14, 16]. It has been demonstrated that they can effectively model different complex dynamic systems, including systems with perturbed parameters, chaos, multiple limit cycles, and others. They also allow us to analyze more effectively systems existing in modern technology, such as adaptive wide area networks, fault-tolerant systems, systems with multiple periods of sampling, etc. It has also been shown that there is a large class of non-linear control plants that can be stabilized

through switched linear local controllers, but there is no way to do it through a static state feedback [14].

The idea of the switching submodel can also be used in the class of models of fractional order. From the research works conducted worldwide it follows that the description by means of the fractional derivative is one of the most effective methods of modeling the real properties of many complex phenomena and industrial processes. As in the case of integer-order models such a description may be used indirectly for tuning or directly for synthesis of linear control algorithms [13, 21].

But still, in spite of great successes achieved in applications, especially in engineering, chemical, automotive and power industry and traffic control of vehicles, quite a lot of questions of a theoretical nature have not been completely solved. In the case of switched, non-integer order systems the number of open issues is even larger. For example, intensive work is underway on the criteria and conditions for stability and stabilizability of switched linear systems [13, 14].

The paper is organized as follows: Section 2 recalls the basics of the differential calculus of fractional order and fractional order models; Section 3 describes the switched models of fractional order; in Section 4 a new way to study the stability of a class of fractional switched models is proposed. The paper is completed by a summary.

2 Fractional dynamic models

2.1 Differential (difference) calculus of non-integer order

Let us consider the known integer-order $n \geq 0$ differential operator of the function $f(t)$, $t \in \mathbb{R}_+$, ${}_t D_t^n f(t) = \frac{d^n f(t)}{dt^n}$, which is equivalent to the m -fold integral operator in the interval $[t_0, t]$, ${}_t I_t^m f(t) = {}_t D_t^{-m} f(t)$, $m = -n$, $n > 0$.

The generalized differential operator of $\alpha \in \mathbb{R}$ order of the function on the interval can be written as [12, 19]:

$${}_t D_t^\alpha f(t) = \begin{cases} \frac{d^\alpha f(t)}{dt^\alpha} & \text{for } \alpha > 0, \\ f(t) & \text{for } \alpha = 0, \\ {}_t I_t^{-\alpha} f(t) & \text{for } \alpha < 0. \end{cases} \quad (1)$$

under assumption that the real function $f(t)$ is defined almost everywhere for $t \geq 0$ and it is multiple differentiable and integrable (depending on the order α) within every interval $[0, T]$, $T > 0$ and exponentially restricted.

There are known several definitions of the operator (1) proposed by various researchers, e.g. those introduced by Weyl, Fourier, Cauchy and Abel, which differ in properties and/or the range of applicability. However, the most popular and most adopted are three of them: Riemann-Liouville's, Caputo's and Grünwald-Letnikov's, ones [17, 19, 21]. The Grünwald-Letnikov definition of the fractional-order derivative

is particularly popular for reasons of application, especially to digital control systems, where it is natural to use discretized function values $f(t)$ taken with a sampling period for the purpose of computations h .

Definition 1 [12]. A derivative of fractional order α of function $f(t)$ is defined according to Grünwald and Letnikov as follows

$${}_{t_0}^L D_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{j=0}^{\lfloor \frac{t-t_0}{h} \rfloor} c_j^\alpha f(t-jh) \quad (2)$$

where the symbol $\lfloor \cdot \rfloor$ denotes the integer part,

$$c_j^\alpha = (-1)^j \binom{\alpha}{j}, \quad j = 0, 1, 2, \dots \quad (3)$$

and the so-called generalized Newton symbol is given by

$$\binom{\alpha}{j} = \begin{cases} 1 & \text{dla } j = 0 \\ \frac{\alpha(\alpha-1)\dots(\alpha-j+1)}{j!} & \text{dla } j = 1, 2, 3, \dots \end{cases} \quad (4)$$

In practice computer control systems are most commonly used, i.e. discrete control algorithms and discrete models of controlled plants are considered, and thereby discrete functions defined at discrete time instants $t \in \mathbb{Z}$. For discrete functions the fractional-order difference calculus represents an equivalent of the fractional-order differential calculus. Hence, based upon (2), the following definition may be introduced:

Definition 2 [19]. A discrete fractional-order difference of a discrete function is defined by

$${}_{t_0} \Delta_t^\alpha f(t) = \sum_{j=0}^{t-t_0} c_j^\alpha f(t-j), \quad \alpha \in \mathbb{R}, \quad t \in \mathbb{Z} \quad (5)$$

with the most commonly adopted simplified notation $t_0 = 0$ as

$$\Delta^\alpha f(t) = \sum_{j=0}^t c_j^\alpha f(t-j) \quad (6)$$

If it is considered that the number of summands in the sum (6) is unavoidably finite in practice, then the following approximation is adopted most commonly

$$\Delta^\alpha f(t) = \sum_{j=0}^L c_j^\alpha f(t-j) \quad (7)$$

where the number of the samples $f(t)$ (equivalent to the length of memory where the samples are stored in practical realizations) should be chosen so that the truncation error does not exceed a given value ε . It is possible to satisfy, taking into account that the coefficients (3) decrease with the increase of j , that is the effect of samples being distant in time is becoming smaller and smaller. Assuming the value of the function does not exceed the value M at any point the memory length may be estimated from the condition [20]:

$$L \geq \left(\frac{M}{\varepsilon |\Gamma(1-\alpha)|} \right)^{1/\alpha} \tag{8}$$

2.2 Discrete-time models of fractional order

Dynamic systems of non-integer order can be modeled in many ways, with transfer function and state space descriptions being most popular.

The nonlinear discrete-time model of fractional order in state variables can be introduced on the basis of the integer-order model [12]:

$$x(t + 1) = f(x(t), u(t)), \quad t \in \mathbb{Z} \tag{9}$$

Denoting

$$f_d(x(t), u(t)) = f(x(t), u(t)) - x(t) \tag{10}$$

we get

$$\Delta^1 x(t + 1) = f_d(x(t), u(t)) \tag{11}$$

Hence, by analogy, we can write:

Definition 3 [13]. A nonlinear discrete-time model of fractional order α in state variables is given by nonlinear state and output equations

$$\Delta^\alpha x(t + 1) = f_d(x(t), u(t)), \quad x(0) = x_0, \quad t \in \mathbb{Z} \tag{12}$$

$$y(t) = g(x(t), u(t)) \tag{13}$$

where the individual vectors $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^p$ denote the model state, input and output, respectively, $t \in \mathbb{Z}$ denotes a discrete-time independent variable (consecutive sample instants).

In the linear case, by analogy with integer-order models, it may be introduced the definition of linear discrete-time models of fractional order. Taking for simplicity the most common case of the zero matrix \mathbf{D} in the output equation, we get:

Definition 4 [13]. A linear discrete-time model of fractional order α in state variables is given by the state and output equations

$$\Delta^\alpha x(t + 1) = \mathbf{A}_d x(t) + \mathbf{B}u(t), \quad x(0) = x_0, \quad t \in \mathbb{Z} \tag{14}$$

$$y(t) = \mathbf{C}x(t) \tag{15}$$

where x_0 denotes the initial state, $\mathbf{A} \in \mathbb{R}^{n \times n}$ is the state matrix, $\mathbf{B} \in \mathbb{R}^{n \times m}$, $\mathbf{C} \in \mathbb{R}^{p \times n}$ are the input and output matrices and

$$\mathbf{A}_d = \mathbf{A} - \mathbf{I}_n \tag{16}$$

where $\mathbf{I}_n \in \mathbb{R}^{n \times n}$ denotes the identity matrix.

3 Switched models of fractional order

As mentioned, dynamical switched systems are systems that consist of a finite number of dynamical time-invariant subsystems and a logical rule that coordinates switching between these subsystems. The switched signal $\sigma(t) \in \mathbb{Z}^+$, which defines the instantaneous degree of activity of each submodel is determined by a special supervisor. From (12) – (15) it follows:

Definition 5. A nonlinear discrete-time model of a switched system of non-integer order is given by the state equation

$$\Delta^{\alpha_{\sigma(t)}}x(t+1) = f_{d,\sigma(t)}(x(t), u(t)), \quad x(0) = x_0, \quad t \in \mathbb{Z} \quad (17)$$

$$y(t) = g_{\sigma(t)}(x(t)) \quad (18)$$

Definition 6. A linear discrete-time model of a switched system of non-integer order is given by the state equation

$$\Delta^{\alpha_{\sigma(t)}}x(t+1) = \mathbf{A}_{d,\sigma(t)}x(t) + \mathbf{B}_{\sigma(t)}u(t) \quad (19)$$

$$y(t) = \mathbf{C}_{\sigma(t)}x(t) \quad (20)$$

where:

$\sigma(t) : \mathbb{N} \rightarrow J = \{1, 2, \dots, N\}$ denotes the switching signal,

N is the number of possible submodels.

The law that governs switching of individual time-invariant submodels can be written as follows:

$$\sigma(t) = \{(t_0, j_0), \dots, (t_k, j_k) \mid \sigma(t_k) = j_k \in J, \quad k \in \mathbb{N}\} \quad (21)$$

here $t_k \in \mathbb{N}$, $t_0 < t_1 < t_2 \dots < t_k < \dots$, with t_0 denoting the initial time, and t_k the time of the k -th switching. At the instant:

$$t \in \mathcal{N}[t_k, t_{k+1}), \quad \mathcal{N}[t_k, t_{k+1}) = \{t : t \in \mathbb{N}, t_k \leq t < t_{k+1}\}, \quad \sigma(t) = \sigma(t_k) = j_k \quad (22)$$

the j_k -th subsystem is activated. In other words, the solution $x(t)$ of the system (17) or (19) switched according to $\sigma(t)$ is a trajectory with the initial point (t_0, x_0) , and for each $t \in \mathcal{N}[t_k, t_{k+1})$ the modeled switched system is defined by the j_k -th submodel.

In general, the switched signal can be a function of time, its future values, model state variables or model outputs, model inputs, and also can be a function of an external auxiliary signal.

$$\sigma(t+1) = h(t, \sigma(t-i), x(t), y(t), u(t), z(t)), \quad i = 0, 1, 2, \dots \quad (23)$$

Depending on the specific form of the logic equation (23) different switching laws can be distinguished [13], e.g., switching path, time-driven switching law, event-driven switching law. In practice, of greatest importance, especially in the context of

modeling nonlinear plants and designing nonlinear controllers, are special cases of the event-driven switching law, with state and/or input feedback

$$\sigma(t + 1) = h(\sigma(t), x(t), u(t)) \tag{24}$$

and switching systems with an auxiliary switching signal, encountered, for example, in gain scheduling adaptive control systems

$$\sigma(t + 1) = h(\sigma(t), z(t)) \tag{25}$$

Analysis of switched systems of integer order has been the subject of many publications for several years [5, 9, 10, 14, 15, 16, 22]. Depending on the type of the switching signal (23) – (25) it is studied, for example, the problem of the stability of the switched systems, when there is no restrictions imposed on the switching signal (so-called arbitrary switching) and the problem what restrictions should be put on the switching signals in order to guarantee the stability of switched systems (so-called constrained switching).

For example, there are known cases when, in general, the stability requirement for all subsystems is not sufficient to assure stability for the switched system and conversely, when even unstable subsystems, but appropriately switched, may assure stability of the switched system.

In the case of non-integer order systems (17), (19), the variability of the fractional difference of the state vector should be considered in addition to the parameter variability of the function $f_{\sigma(t)}$ or the matrix $\mathbf{A}_{\sigma(t)}$, $\mathbf{B}_{\sigma(t)}$, $\mathbf{C}_{\sigma(t)}$. In literature there are known several types of variability of the fractional derivative (2) differing in their properties [6, 15]. By analogy, the following types of variability can be defined for discrete systems and discrete fractional difference (7):

Definition 7. Discrete fractional differences of the time-variant order of a discrete function $f(t)$, called \mathcal{A} to \mathcal{E} types, are defined as:

$${}^{\mathcal{A}}\Delta_t^{\alpha(t)} f(t) = \sum_{i=0}^L (-1)^i \binom{\alpha(t)}{i} f(t - i) \tag{26}$$

$${}^{\mathcal{B}}\Delta_t^{\alpha(t)} f(t) = \sum_{i=0}^L (-1)^i \binom{\alpha(t - i)}{i} f(t - i) \tag{27}$$

$${}^{\mathcal{C}}\Delta_t^{\alpha(t)} f(t) = \sum_{i=0}^L (-1)^i \binom{\alpha(t - L + i)}{i} f(t - i) \tag{28}$$

$${}^{\mathcal{D}}\Delta_t^{\alpha(t)} f(t) = \left(f(t) - \sum_{i=1}^L (-1)^i \binom{-\alpha(t)}{i} {}^{\mathcal{A}}\Delta_{t-i}^{\alpha(t)} f(t) \right) \tag{29}$$

$${}^{\mathcal{E}}\Delta_t^{\alpha(t)} f(t) = \lim_{h \rightarrow 0} \left(f(t) - \sum_{i=1}^L (-1)^i \binom{-\alpha(t - i)}{i} {}^{\mathcal{B}}\Delta_{t-i}^{\alpha(t)} f(t) \right) \tag{30}$$

A practical interpretation of individual types of the discrete fractional difference of time-variant order is given in Table 1.

Table 1. Interpretation of discrete fractional differences of time-variant order

Type	Interpretation
\mathcal{A} (26)	current coefficients are applicable to all data, also to those being sampled during the validity of past values of the order (whole system memory is changed to the same extent)
β (27)	to past data are applicable coefficients appropriate for discrete time instants from which these data come (the so-called short-term memory is changed)
\mathcal{C} (28)	to past data, more distant in time, are applicable newer coefficients and conversely, to historic data less distant in time are applicable older coefficients. This is equivalent to switching with delay – the older the samples, the earlier are switched coefficients corresponding to them, while the coefficients corresponding to newer data are switched after certain time from the model change (the so-called long-term memory is changed)
\mathcal{D} (29)	discrete fractional difference is expressed through past discrete differences of the function, with consideration for the currently changed order, as in type \mathcal{A} (26)
\mathcal{E} (30)	discrete fractional difference is expressed through past discrete differences of the function; to past differences are applicable coefficients appropriate for discrete time instants from which these differences come, as in type β (27)

4 Stability of discrete-time linear fractional-order switched systems

4.1 Common quadratic Lyapunov function method

Stability of linear systems in state space can be studied by means of, amongst others, the commonly known direct Lyapunov method. Its use boils down to verifying a matrix inequality, the form of which depends on the type of model. For example, an integer-order $\alpha = 1$ discrete-time model is asymptotically stable if and only if there exists a positive definite symmetric matrix $\mathbf{P} \in \mathbb{R}^{n \times n}$, $\mathbf{P} = \mathbf{P}^T$, for which the below given matrix inequality is satisfied

$$\mathbf{A}^T \mathbf{P} \mathbf{A} - \mathbf{P} < 0 \tag{31}$$

For switched integer-order models with arbitrary switching the main tool to assess stability is the so-called common quadratic Lyapunov function (CQLF) derived from the direct Lyapunov method.

Theorem 1 [14]. A switched discrete model (19) of integer order $\alpha = 1$ composed of linear submodels of the order $\alpha = 1$ is asymptotically stable if and only if there exists a positive definite symmetric matrix $\mathbf{P} \in \mathbb{R}^{n \times n}$, $\mathbf{P} = \mathbf{P}^T$, such that

$$\mathbf{A}_j^T \mathbf{P} \mathbf{A}_j - \mathbf{P} < 0 \quad \forall j \in \{1, 2, \dots, N\} \tag{32}$$

However, it should be noted that numerical methods to solve LMIs (32) for a greater number of stable subsystems to proof the existence of a CQLF remains a challenging task and the standard numerical methods are ineffective. Therefore, methods being more numerically effective for assessing the stability of discrete-time integer-order switched systems with arbitrary switching yielding less conservative results are sought for [13]. In the case of discrete-time fractional-order switching systems the problem is much more difficult.

4.2 Switched quadratic Lyapunov function method

One of the ways to simplify stability testing of switched systems is the use of the so-called switched quadratic Lyapunov function (SQLF) as in [14]

$$V(t, x(t)) = x^T(t)P_{\sigma(t)}x(t) \tag{33}$$

where the switched matrix encompasses all positive definite symmetric matrices P_j that solves the Lyapunov equation for each j -th subsystem, $j \in \{1, 2, \dots, N\}$. For SQLF (33) there can be created LMIs that enable the stability of the switched model to be tested in a simpler way. Among others, the following holds true:

Theorem 2 [13]. If there exist positive definite symmetric matrices $P_j, Q_j \in \mathbb{R}^{n \times n}$, $P_j = P_j^T, Q_j = Q_j^T$, auxiliary matrices $F_{ij}, G_{ij} \in \mathbb{R}^{n \times n}$ and scalars μ_{ij} satisfying

$$\begin{bmatrix} A_i^T F_{ij}^T + F_{ij} A_i - P_i + \mu_{ij} Q_i & A_i^T G_{ij}^T - F_{ij} \\ G_{ij} A_i - F_{ij}^T & P_j - G_{ij} - G_{ij}^T + \mu_{ij} Q_j \end{bmatrix} < 0 \tag{34}$$

for all $i, j \in \{1, 2, \dots, N\}$, then the switched linear system (19) is asymptotically stable under arbitrary switching.

Although LMIs of (34) appear complicated, they are numerically easier to solve than LMIs of (32) [14].

4.3 Main result for discrete-time non-integer order switching systems

Consider the discrete-time, linear, non-integer order switched state models (19), (20) with discrete differences of finite memory length, switched according to the types \mathcal{A} (30) or \mathcal{B} (31). For such systems the following results can be formulated

– **discrete differences switched according to the type \mathcal{A} .**

The switching phase in such a case is instant. Fig. 1 shows an example with the switching from the submodel 1 to submodel 2, for $L = 5$.

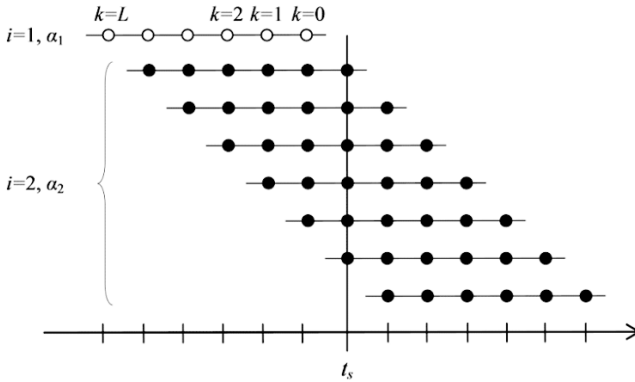


Fig. 1. Switching from the submodel 1 to submodel 2 according to type \mathcal{A} , $L = 5$

Theorem 3. If there exist positive definite symmetric matrices $\mathbf{P}_i, \mathbf{Q}_i \in \mathbb{R}^{n \times n}$, $\mathbf{P}_i = \mathbf{P}_i^T, \mathbf{Q}_i = \mathbf{Q}_i^T$, auxiliary matrices $\mathbf{F}_{ij}, \mathbf{G}_{ij} \in \mathbb{R}^{n \times n}$ and scalars μ_{ij} satisfying

$$\begin{bmatrix} \bar{\mathbf{A}}_i^T \mathbf{F}_{ij}^T + \mathbf{F}_{ij} \bar{\mathbf{A}}_i - \mathbf{P}_i + \mu_{ij} \mathbf{Q}_i & \bar{\mathbf{A}}_i^T \mathbf{G}_{ij}^T - \mathbf{F}_{ij} \\ \mathbf{G}_{ij} \bar{\mathbf{A}}_i - \mathbf{F}_{ij}^T & \mathbf{P}_j - \mathbf{G}_{ij} - \mathbf{G}_{ij}^T + \mu_{ij} \mathbf{Q}_j \end{bmatrix} < 0 \quad (35)$$

for all $i, j \in \{1, 2, \dots, N\}$, where

$$\bar{\mathbf{A}}_i = \begin{bmatrix} \bar{\mathbf{A}}_{d,i} & -\bar{\mathbf{c}}_{2,i} & -\bar{\mathbf{c}}_{3,i} & \dots & -\bar{\mathbf{c}}_{L-1,i} & -\bar{\mathbf{c}}_{L,i} \\ \mathbf{I}_n & \mathbf{0}_n & \mathbf{0}_n & \dots & \mathbf{0}_n & \mathbf{0}_n \\ \mathbf{0}_n & \mathbf{I}_n & \mathbf{0}_n & \dots & \mathbf{0}_n & \mathbf{0}_n \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0}_n & \mathbf{0}_n & \mathbf{0}_n & \dots & \mathbf{I}_n & \mathbf{0}_n \end{bmatrix} \quad (36)$$

$$\bar{\mathbf{A}}_{d,i} = \mathbf{A}_{d,i} + \alpha_i \mathbf{I}_n \quad (37)$$

$$\bar{\mathbf{c}}_{k,i} = c_k^{\alpha_i} \mathbf{I}_n, \quad k = 2, \dots, L \quad (38)$$

then the discrete-time, linear, non-integer order switched state model (19), (20) with the matrices $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i$ and the discrete differences Δ^{α_i} of finite memory length of type \mathcal{A} (26) is asymptotically stable under arbitrary switching.

Proof. Consider the discrete-time, linear, non-integer order state model (14), (15) with the discrete difference of finite memory length (7). For such a model it can be created an expanded state approximation [23] with the use of an "expanded" state formed by the current state and the past states [24]

$$\bar{\mathbf{x}}(t) = [\mathbf{x}^T(t) \quad \mathbf{x}^T(t-1) \quad \dots \quad \mathbf{x}^T(t-L+1) \quad \mathbf{x}^T(t-L)]^T \in \mathbb{R}^{n \cdot L} \quad (39)$$

$$\bar{\mathbf{x}}(t+1) = \bar{\mathbf{A}} \bar{\mathbf{x}}(t) + \bar{\mathbf{B}} u(t), \quad t \in \mathbb{Z} \quad (40)$$

$$\mathbf{y}(t) = \bar{\mathbf{C}} \bar{\mathbf{x}}(t) \quad (41)$$

with

$$\bar{A} = \begin{bmatrix} \bar{A}_d & -\bar{c}_2 & -\bar{c}_3 & \cdots & -\bar{c}_{L-1} & -\bar{c}_L \\ I_n & \mathbf{0}_n & \mathbf{0}_n & \cdots & \mathbf{0}_n & \mathbf{0}_n \\ \mathbf{0}_n & I_n & \mathbf{0}_n & \cdots & \mathbf{0}_n & \mathbf{0}_n \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0}_n & \mathbf{0}_n & \mathbf{0}_n & \cdots & I_n & \mathbf{0}_n \end{bmatrix} \in \mathbb{R}^{n \cdot L \times n \cdot L} \quad (42)$$

$$\bar{B} = [\mathbf{B}^T \quad \mathbf{0}_{m \times n} \quad \cdots \quad \mathbf{0}_{m \times n}]^T \in \mathbb{R}^{n \cdot L \times m} \quad (43)$$

$$\bar{C} = [\mathbf{C} \quad \mathbf{0}_{p \times n} \quad \cdots \quad \mathbf{0}_{p \times n}] \in \mathbb{R}^{p \times n \cdot L} \quad (44)$$

where $I, \mathbf{0}$ is the identity and null matrices of appropriate dimension, and

$$\bar{A}_d = A_d + \alpha I_n \quad (45)$$

$$\bar{c}_k = c_k^\alpha I_n, \quad k = 2, 3, \dots, L \quad (46)$$

Writing all non-integer order submodels with matrices A_i, B_i, C_i and the discrete differences Δ^{α_i} as the integer-order state models (44) – (50), and using Theorem 2, eq. (35) follows directly from eq. (34), which completes the proof.

– **discrete differences switched according to the type β**

Stability analysis for discrete-time, linear, non-integer order switched state models (19), (20) under arbitrary switching with discrete differences of finite memory length (7), switched according to the type β (27) is much more difficult, because the switching phase in such a case lasts L discrete-time instants, where L is the finite memory length. Fig. 2 shows an example with the switching from the submodel 1 to submodel 2, for $L = 5$.

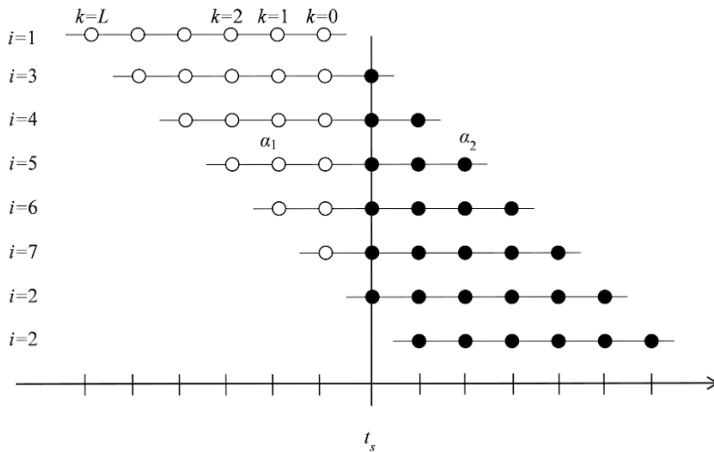


Fig. 2. Switching from the submodel 1 to submodel 2 according to type β , $L = 5$

Thus, it will be considered a simpler case of a switched system that contains only two non-integer order discrete-time subsystems, i.e. with the matrices A_i, B_i, C_i and the discrete differences Δ^{α_i} of finite memory length ($i = 1, 2$), where switching from

subsystem 1 to subsystem 2 and/or vice-versa occurs no more often than at every L discrete-time instant. For such a system the following result can be formulated:

Theorem 4. If there exist positive definite symmetric matrices $\mathbf{P}_i, \mathbf{Q}_i \in \mathbb{R}^{n \times n}$, $\mathbf{P}_i = \mathbf{P}_i^T, \mathbf{Q}_i = \mathbf{Q}_i^T$, auxiliary matrices $\mathbf{F}_{ij}, \mathbf{G}_{ij} \in \mathbb{R}^{n \times n}$ and scalars μ_{ij} satisfying (35) for all $i, j \in \{1, 2, \dots, L, \dots, 2L + 2\}$ and matrices $\bar{\mathbf{A}}_i$ defined by (36),

for basic submodels:

$$\bar{\mathbf{A}}_{d,i} = \mathbf{A}_{d,i} + \alpha_i \mathbf{I}_n \quad \text{for } i = 1, 2 \tag{47}$$

$$\bar{\mathbf{c}}_{k,i} = c_k^{\alpha_i} \mathbf{I}_n, \quad k = 2, \dots, L \quad \text{for } i = 1, 2 \tag{48}$$

and auxiliary submodels for switching phases

$$\bar{\mathbf{A}}_{d,i} = \mathbf{A}_{d,j} + \alpha_j \mathbf{I}_n \quad \text{for } i = 3, \dots, 2L + 2 \tag{49}$$

where

$$j = \begin{cases} 2 & \text{for } i = 3, \dots, L + 2 & (\text{phase } \alpha_1 \rightarrow \alpha_2) \\ 1 & \text{for } i = L + 3, \dots, 2L + 2 & (\text{phase } \alpha_2 \rightarrow \alpha_1) \end{cases} \tag{50}$$

and

$$\bar{\mathbf{c}}_{k,i} = c_k^{\alpha_i} \mathbf{I}_n, \quad k = 2, \dots, L \tag{51}$$

where for $i = 3, \dots, L + 2$ (phase $\alpha_1 \rightarrow \alpha_2$)

$$l = \begin{cases} 1 & \text{if } k > i - 2 \\ 2 & \text{else} \end{cases} \tag{52}$$

and for $i = L + 3, \dots, 2L + 2$ (phase $\alpha_2 \rightarrow \alpha_1$)

$$l = \begin{cases} 2 & \text{if } k > i - L - 2 \\ 1 & \text{else} \end{cases} \tag{53}$$

then the discrete-time, linear, non-integer order switched state model (19), (20) with two non-integer order submodels and the discrete differences of finite memory length of type β (27) is asymptotically stable under arbitrary switching.

Proof. The proof is analogous to that of Theorem 3. After creating the expanded state approximation (39) – (46) for two fractional-order basic submodels (47), (48) and for $2L$ auxiliary fractional-order submodels according to (49) – (53), eq. (35) follows directly from eq. (34), which completes the proof.

5 Summary

In the paper the method for approximation of two kinds of switched fractional linear systems is proposed. It has been employed the expanded state approximation, which makes it possible to utilize results of the stability theory for integer-order switched

system. The stability sufficient conditions for two types of the variability of fractional-orders have been derived. The presented approximation of switched non-integer order systems can also be used to obtain stability necessary conditions.

6 References.

1. Busłowicz M.: Robust stability of positive discrete-time linear systems of fractional order. *Bulletin of Polish Academy of Sciences, Technical Sciences*, Vol. 58, No. 4, pp. 567–572, 2010.
2. Busłowicz M.: Stability of state-space models of linear continuous-time fractional order systems. *Acta Mechanica et Automatica*, Vol. 5, No. 2, pp. 15–22, 2011.
3. Busłowicz M., Ruzewski A.: Robust stability check of fractional discrete-time linear systems with interval uncertainties. In: Latawiec K., Łukaniszyn M., Stanisławski R. (eds.): *Advances in Modelling and Control of Non-integer Order Systems*. LN in Electrical Engineering 320, Springer, pp. 199–208, 2014.
4. Chen Y. Q., Ahnand H.S., Podlubny I.: Robust stability check of fractional order linear time invariant systems with interval uncertainties. *Signal Processing*, Vol. 86, No. 1, pp. 2611–2618, 2006.
5. Domek S.: Piecewise Affine Representation of Discrete in Time, Non-integer Order Systems. In: Mitkowski W., Kacprzyk J., Baranowski J. (eds.): *Advances in the Theory and Applications of Non-integer Order Systems*. LN in Electrical Engineering 257, Springer, pp. 149–160, 2013.
6. Domek S.: *Fractional-order differential calculus in model predictive control*. West Pomeranian University of Technology Academic Press, Szczecin, 2013.
7. Domek S.: Switched state model predictive control of fractional-order nonlinear discrete-time systems. In: Pisano A., Caponetto R. (eds.): *Advances in Fractional Order Control and Estimation, Asian J. Control*, Special Issue, Vol. 15, No. 3, pp. 658–668, 2013.
8. Dzieliński A., Sierociuk D.: Stability of discrete fractional order state-space systems. *Journal of Vibration and Control*, Vol. 14, No. 9–10, pp. 1543–1556, 2008.
9. Fang L., Lin H., Antsaklis P. J.: Stabilization and performance analysis for a class of switched systems. In: *Proc. 43rd IEEE Conf. Decision Control*, Atlantis, pp. 1179–1180, 2004.
10. Geyer T., Torrisi F., Morari M.: Optimal complexity reduction of polyhedral piecewise affine systems. *Automatica*, Vol. 44, No. 7, pp. 1728–1740, 2008.
11. Kaczorek T.: New stability tests of positive standard and fractional linear systems. *Circuits and Systems*, Vol. 2, No. 4, pp. 261–268, 2011.
12. Kaczorek T.: *Selected Problems of Fractional Systems Theory*. Springer, Berlin, 2011.
13. Lin H., Antsaklis P. J.: Switching Stabilization and L2 Gain Performance Controller Synthesis for Discrete-Time Switched Linear Systems. *Proc. 45th IEEE Conference on Decision & Control*, San Diego, CA, USA, pp. 2673–2678, 2006.
14. Lin H., Antsaklis P. J.: Stability and stabilizability of switched linear systems: A survey of recent results. *IEEE Trans. on Automatic Control*, Vol. 54, No. 2, pp. 308–322, 2009.
15. Macias M., Sierociuk D.: An alternative recursive fractional variable-order derivative definition and its analog validation., *Proc. Inter. Conf. on Fractional Differentiation and its Applications (ICFDA)*, Catania, pp. 1–6, 2014.
16. Mäkilä P. M., Partington J. R.: On linear models for nonlinear systems. *Automatica*, Vol. 39, pp. 1–13, 2003.

17. Monje C. A., Chen Y. Q., Vinagre B. M., Xue D., Feliu V.: *Fractional order systems and controls*. Springer-Verlag, London, 2010.
18. Moze M., Sabatier J., and Oustaloup A.: LMI characterization of fractional systems stability. In: Sabatier J., Agrawal O. P., Tenreiro Machado J. A. (eds.): *Advances in Fractional Calculus: Theoretical developments and applications in physics and engineering*, Springer, pp. 419–434, 2007.
19. Ostalczyk P.: The non-integer difference of the discrete-time function and its application to the control system synthesis. *Int. J. Syst. Sci.*, vol. 31, no. 12, pp. 1551–1561, 2000.
20. Petrás I.: Fractional Derivatives, Fractional Integrals, and Fractional Differential Equations in Matlab. In: Assi A. (ed.): *Engineering Education and Research Using MATLAB*, InTech, Rijeka, Shanghai 2011.
21. Podlubny I.: *Fractional Differential Equations*, San Diego, Academic Press, 1999.
22. Shevitz D., Paden B.: Lapunov stability theory of nonsmooth systems. *IEEE Trans. on Automatic Control*, Vol. 39, No. 9, pp. 1910–1914, 1994.
23. Stanisławski R.: *Advances in modeling of fractional difference systems – new accuracy, stability and computational results*. Oficyna Wydawnicza Politechniki Opolskiej, Opole, 2013.
24. Stanisławski R., Latawiec K. L., Łukaniszyn M., Gałek M.: Time-domain approximations to the Grünwald-Letnikov difference with application to modeling of fractional-order state space systems. *Proc. 20th Int. Conference on Methods and Models in Automation and Robotics*, Międzyzdroje, Poland, pp. 579–584, 2015.

Relationship between controllability of standard and fractional linear systems

Jerzy Klamka

Silesian University of Technology
Gliwice, Poland
Email: Jerzy.Klamka@polsl.pl

Abstract. The relationship between the controllability of standard and fractional linear stationary discrete-time addressed. It is shown that the fractional linear discrete-time control system is controllable if and only if the corresponding linear standard discrete-time system is controllable.

Keywords: controllability, fractional, standard, linear, discrete-time, continuous-time, system.

1 Introduction

The notion of controllability of linear systems have been introduced by Kalman. Those notions are the basic concepts of the modern control theory [1, 2, 3, 4]. They have been extended to positive and fractional linear and nonlinear systems [1, 2]. The mathematical fundamentals of fractional calculus are given in the monographs [2]. The positive fractional linear systems have been introduced in [2].

In this paper it will be shown that the fractional discrete-time and continuous-time linear systems are controllable if and only if the standard discrete-time and continuous-time linear systems are controllable.

The paper is organized as follows. In section 2 the basic definitions and theorems concerning standard and fractional discrete-time and continuous-time linear systems are recalled. The relationship between the controllability of the standard and fractional discrete-time linear systems is considered in section 3. Moreover, concluding remarks are given in section 4.

The following notation will be used: $\mathfrak{R}^{n \times m}$ is the set of $n \times m$ real matrices and $\mathfrak{R}^n = \mathfrak{R}^{n \times 1}$, Z_+ is the set of nonnegative integers, I_n is the $n \times n$ identity matrix.

2 Preliminaries

Consider the standard discrete-time linear system described by difference state equation

$$x_{i+1} = Ax_i + Bu_i, \quad i \in Z_+ = \{0, 1, \dots\}, \quad (2.1)$$

where $x_i \in \mathfrak{R}^n$, $u_i \in \mathfrak{R}^m$, $y_i \in \mathfrak{R}^p$ are state, input and output vectors and $A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times m}$, $C \in \mathfrak{R}^{p \times n}$.

The solution to the equation (2.1) is given by

$$x_i = A^i x_0 + \sum_{j=0}^{i-1} A^{i-j-1} B u_j. \quad (2.2)$$

Now let us consider the fractional discrete-time linear system

$$\Delta^\alpha x_{i+1} = Ax_i + Bu_i, \quad 0 < \alpha < 2, \quad (2.3)$$

Where fractional difference operator is defined as follows

$$\Delta^\alpha x_i = \sum_{j=0}^i (-1)^j \binom{\alpha}{j} x_{i-j}, \quad (2.4)$$

$$\binom{\alpha}{j} = \begin{cases} 1 & \text{for } j = 0 \\ \frac{\alpha(\alpha-1)\dots(\alpha-j+1)}{j!} & \text{for } j = 1, 2, \dots \end{cases} \quad (2.5)$$

is the fractional α -order difference of x_i and $x_i \in \mathfrak{R}^n$, $u_i \in \mathfrak{R}^m$, $y_i \in \mathfrak{R}^p$ are state, input and output vectors and $A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times m}$, $C \in \mathfrak{R}^{p \times n}$.

Substitution of (2.4c) into (2.4a) yields

$$x_{i+1} = (A + I_n \alpha) x_i + \sum_{j=2}^{i+1} d_j x_{i-j+1} + Bu_i, \quad i \in Z_+, \quad (2.6)$$

where

$$d_j = d_j(\alpha) = (-1)^{j+1} \binom{\alpha}{j}, \quad j = 2, 3, \dots \quad (2.7)$$

The solution to the equation (2.3) has the form [1]

$$x_i = \Phi_i x_0 + \sum_{j=0}^{i-1} \Phi_{i-j-1} B u_j, \quad (2.8)$$

Where the transition state matrix can be computed using difference equation

$$\Phi_{j+1} = \Phi_j(A + I_n \alpha) + \sum_{k=2}^{j+1} d_k \Phi_{j-k+1}, \quad \Phi_0 = I_n \quad (2.9)$$

and

$$\Phi_0(t) = \sum_{k=0}^{\infty} \frac{A^k t^{k\alpha}}{\Gamma(k\alpha + 1)},$$

$$\Phi(t) = \sum_{k=0}^{\infty} \frac{A^k t^{(k+1)\alpha-1}}{\Gamma[(k+1)\alpha]}$$

Theorem 2.1. (Cayley-Hamilton) Let $A \in \mathfrak{R}^{n \times n}$ and

$$\det[I_n \lambda - A] = \lambda^n + a_{n-1} \lambda^{n-1} + \dots + a_1 \lambda + a_0. \quad (2.10)$$

Then

$$A^n + a_{n-1} A^{n-1} + \dots + a_1 A + a_0 I_n = 0. \quad (2.11)$$

Proof. Proof is given in [1].

Using the Cayley-Hamilton theorem (the equality (2.10)) it is possible to eliminate the powers $k = n, n+1, \dots$ of the matrix A^k in (2.8) and (2.9) we obtain

$$\Phi_0(t) = \sum_{k=0}^{n-1} c_k(t) A^k. \quad (2.12)$$

$$\Phi(t) = \sum_{k=0}^{n-1} d_k(t) A^k. \quad (2.13)$$

The coefficients $c_k(t)$ and $d_k(t)$ can be computed using method given in [2].

3 Controllability of standard and fractional discrete-time linear systems

First of all well known controllability definition is recalled (see e.g. [3]) for more details.

Definition 3.1. The standard linear discrete-time linear system (2.1) is called controllable in the given interval $[0, q]$ if for any initial condition x_0 and any final

state x' at time q , it is possible to find the sequence of admissible controls u_i for $i = 0, 1, \dots, q - 1$, such that the solution of state equation $x_q = x'$

Now the main result of the paper is presented in the following theorem.

Theorem 3.1. The standard linear discrete-time linear system (2.1) is controllable in the interval $[0, q]$ if and only if the fractional discrete-time linear system (2.3) is controllable in the interval $[0, q]$.

Proof. First of all let us observe that the matrix

$$\begin{bmatrix} I_n & \alpha I_n & d_2 + \alpha^2 & \dots & \dots \\ 0 & I_n & 2\alpha & \dots & \dots \\ 0 & 0 & I_n & \dots & \dots \\ \vdots & \vdots & \vdots & \ddots & \dots \\ 0 & 0 & 0 & \dots & I_n \end{bmatrix}$$

is nonsingular.

Taking into account the equalities (2.12), (2.13) and the nonsingularity of the above matrix we conclude that the controllability matrix for fractional linear discrete-time system (2.3) has the following form

$$\begin{aligned} & \text{rank}[\Phi_0 B \mid \Phi_1 B \mid \Phi_2 B \mid \dots \mid \Phi_k B \mid \dots \mid \Phi_{q-1} B] = \\ & = \text{rank}[B \mid AB \mid A^2 B \mid \dots \mid A^k B \mid \dots \mid A^{q-1} B] \times \\ & \times \begin{bmatrix} I_n & \alpha I_n & d_2 + \alpha^2 & \dots & \dots \\ 0 & I_n & 2\alpha & \dots & \dots \\ 0 & 0 & I_n & \dots & \dots \\ \vdots & \vdots & \vdots & \ddots & \dots \\ 0 & 0 & 0 & \dots & I_n \end{bmatrix} = \\ & = \text{rank}[B \mid AB \mid A^2 B \mid \dots \mid A^k B \mid \dots \mid A^{q-1} B] \end{aligned}$$

Thus our theorem follows.

4 Concluding remarks

The relationship between the controllability of the standard and fractional discrete-time. It has been shown that: the fractional discrete-time linear system is controllable if and only if the standard discrete-time linear systems is controllable. The considerations can be extended to the standard and fractional continuous time-varying linear systems.

References

- [1] Kaczorek T.: *Selected Problems of Fractional Systems Theory*, Springer-Verlag, Berlin, 2011.
- [2] Kaczorek T.: *Vectors and Matrices in Automation and Electrotechnics*, WNT, Warsaw, 1998 (in Polish).
- [3] Klamka J.: *Controllability of Dynamical Systems*, Kluwer, Academic Press, Dordrecht, 1991.
- [4] Klamka J.: Controllability of Semilinear Fractional Discrete Systems. Lecture 8th Asian Conference on Intelligent Information and Database Systems, ACIIDS 2016, Da Nang, Vietnam, March 14-16, 2016.

The research presented here was done by authors as part of the projects funded by the National Science Centre in Poland granted according to decisions DEC-2014/13/B/ST7/00755.

Remarks about stability of fractional systems

Wojciech Mitkowski

AGH University of Science and Technology

Faculty of Electrical Engineering, Automatics, Computer Science and Electronics

Department of Automatics

al. A. Mickiewicza 30, 30-059 Krakow, Poland

wojciech.mitkowski@agh.edu.pl

Abstract. In this work, a selected type of non-integer order system is examined. The conditions for asymptotic stability are formulated with use of modified Michailov theorem. A numerical example for $n=3$ was also presented.

Keywords: fractional order systems, stability.

1. Introduction

In recent years, the fractional system are of great interest for scientists and engineers alike [4, 6, 12, 13, 14, 15, 16, 17, 20, 23]. Non-integer order systems are used for physical modelling of various processes. In-depth analysis and control are also considered [14, 16, 20, 23].

In this work, a selected type of non-integer order system is examined. The conditions for asymptotic stability are formulated with use of modified Michailov theorem. A numerical example for $n=3$ was also presented.

The paper is organized as follows. In the second section, there is a description of the analyzed system of non-integer order. The main results concerning asymptotic stability, are presented in sections 3 and 4. Concluding remarks are given in section 5.

2. Fractional system-preliminary information

Consider the fractional system (fractional-order system) described by the equation

$$\frac{d^\alpha x(t)}{dt^\alpha} = Ax(t) + Bu(t), \quad \alpha \in (0, 1] \quad x(0) = x^0, \quad t \geq 0, \quad (1)$$

where $x(t) \in R^n$, $A \in R^{n \times n}$, $B \in R^{n \times r}$, $u(t) \in R^r$, $\frac{d^\alpha x(t)}{dt^\alpha}$ is the fractional derivative (Kaczorek [5, 6]).

The solution of equation (1) is given by

$$\begin{aligned}
 x(t) &= \Phi_0(t)x(0) + \int_0^t \Phi(t-\tau)Bu(\tau)d\tau \Leftrightarrow x(t) = E_\alpha(At^\alpha)x(0) + \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} E_\alpha(A(t-\tau)^\alpha)Bu(\tau)d\tau \\
 \Phi_0(t) &= E_\alpha(At^\alpha), \quad \Phi(t) = \sum_{k=0}^{+\infty} \frac{A^k t^{(k+1)\alpha-1}}{\Gamma[(k+1)\alpha]}
 \end{aligned}
 \tag{2}$$

where $E_\alpha(z)$ for $z = At^\alpha$ is the Mittag-Leffler matrix function (for example Pillai [19]; Kaczorek [6]),

$$E_\alpha(At^\alpha) = I + \frac{At^\alpha}{\Gamma(1+\alpha)} + \frac{A^2 t^{2\alpha}}{\Gamma(1+2\alpha)} \dots = \sum_{k=0}^{\infty} \frac{A^k t^{k\alpha}}{\Gamma(1+k\alpha)}
 \tag{3}$$

and $\Gamma(\alpha)$ denotes Euler's continuous gamma function

$$\Gamma(\alpha) = \int_0^{\infty} t^{\alpha-1} e^{-t} dt = \int_0^1 (\ln(1/t))^{\alpha-1} dt
 \tag{4}$$

Note that $\Gamma(\alpha+1) = \alpha\Gamma(\alpha)$.

Remark 1. From (3) for $\alpha = 1$ we have $E_1(At) = e^{At}$ and differential equation (1) generates the dynamical system. ■

Remark 2. Let $t \in [0, b]$ and let $x(0) = x^0, x(b) = x^1$. There exists control $u(t)$ which steers the state of the system (1) from $x(0) = x^0$ to the final state $x(b) = x^1$ and $u(t)$ is given by the formula [8]

$$\begin{aligned}
 u(t) &= E_\alpha(A^T(b-t)^\alpha)B^T W^{-1}[x^1 - E_\alpha(Ab^\alpha)x^0], \\
 W &= \frac{1}{\Gamma(\alpha)} \int_0^b (b-\tau)^{\alpha-1} [E_\alpha(A(b-\tau)^\alpha)B][E_\alpha(A(b-\tau)^\alpha)B]^T d\tau
 \end{aligned}
 \tag{5}$$

where T denotes the transpose. There exists control $u(t)$ given in (5) if and only if $\det W \neq 0$. ■

Now we consider control $u(t)$ in the following form (feedback loop):

$$u(t) = Kx(t) + v(t),
 \tag{6}$$

where $v(t)$ is an added control, K is matrix. The closed system (1), (6) is given by following equation:

$$\frac{d^\alpha x(t)}{dt^\alpha} = [A+BK]x(t) + Bv(t), \quad \alpha \in (0, 1] \quad x(0) = x^0, \quad t \geq 0, \quad (7)$$

3. Asymptotic stability condition of fractional system

The linear fractional system given by (1) with $u(t) = 0$ is asymptotically stable if and only if

$$|\arg \lambda| > \alpha \frac{\pi}{2} \quad (8)$$

is satisfied for all eigenvalues λ of matrix A (see for example Matignon [9]; Busłowicz [1]; Kaczorek [7]). Stability region of linear fractional-order system (1) is shown in Fig. 1 (gray area).

Remark 3. Let all eigenvalues of the matrix A lie in the asymptotic stability region (see Fig. 1 – gray area). For $\alpha \in (0, 1)$ the solution of equation (1) with $u(t) = 0$ tends to zero asymptotically but not exponentially. ■

Consider the closed-loop system given in (7). The closed-loop system (7) is asymptotically stable (more precisely is BIBO stable) if and only if the eigenvalues of the matrix $A+BK$ are selected so as to be located in the stability region shown in Fig. 1. There exists a gain matrix K such that the closed-loop system matrix $A+BK$ has selected eigenvalues if and only if $(A; B)$ is controllable. Algorithms for the determination of K are generally known (eg. the algorithms using the Ackermann's formulas).

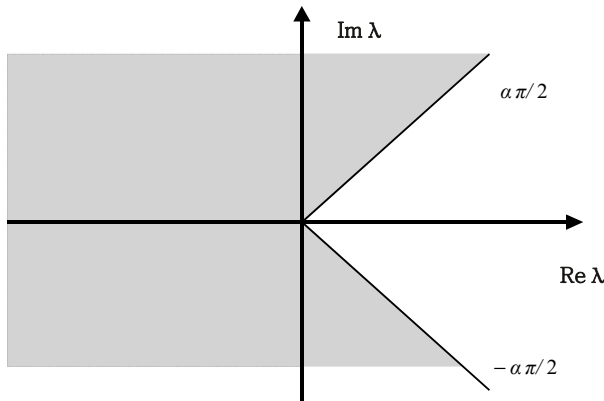


Fig. 1. Asymptotic stability region of linear fractional-order system (5) - gray area

Let $A = [a_{ij}] \in R^{n \times n}$. The characteristic polynomial of matrix A is given by following equality:

$$w(\lambda, A) = f(\lambda) = \det[\lambda I - A] = \lambda^n + a_{n-1}\lambda^{n-1} + \dots + a_1\lambda + a_0 \tag{9}$$

where $a_i \in R$ and for $i = 0, 1, 2, \dots, n-1$, $a_i = (-1)^{n-i} S_{n-i}$ and S_k denote sum of all principal minors of degree k of matrix A (Turowicz [22, p. 117, 131]; Gantmacher [2, p. 78]). For example $S_1 = a_{11} + a_{22} + \dots + a_{nn}$ and $S_n = \det A$. The roots of the polynomial (9) are the eigenvalues of matrix A .

Let $\lambda(A)$ be the spectrum of the matrix A . Let $\lambda_i(A) \in \lambda(A)$ be an eigenvalue of A . The matrix A is asymptotically stable if and only if $\text{Re } \lambda_i(A) < 0$, $i = 1, 2, 3, \dots, n$. The matrix A is said to be Hurwitz if all its eigenvalues lie in the open left half of the complex plane, i.e. $\text{Re } \lambda_i(A) < 0$, $i = 1, 2, 3, \dots, n$.

Remark 4. It is evident that if A is asymptotically stable, then the system (1) is asymptotically stable also (in this case, we use the criterion of Hurwitz).

Stability region of linear fractional-order system (1) is shown in Fig. 1 (gray area) and it is practically defined by the two straight of the following parametric forms:

$$\lambda(t) = [1 \pm j \operatorname{tg}(\alpha \frac{\pi}{2})]t, \quad j^2 = -1, \quad t \in (-\infty, +\infty). \tag{10}$$

In Fig. 1 straight lines are shown in a coordinates $\text{Re } \lambda$ (horizontal axis) and $\text{Im } \lambda$ (vertical axis). Let $\operatorname{tg} \varphi(t) = \operatorname{tg} [\arg \lambda] = \text{Im } \lambda / \text{Re } \lambda$. Now we can formulate the modified Michailov theorem.

Theorem 1. Consider the polynomial (9). Let L be a straight line which does not pass through any of the zero polynomial (9). Let $\Delta_L \arg f(\lambda(t))$ means growth argument, when the parameter t varies from $-\infty$ to $+\infty$. Let l is the number of zeros of polynomial (9) lying on the left side of the line L . Let p is the number of zeros of polynomial (9) lying on the right side of the line L . Zeros count of multiplicity, thus $l + p = n$. Then

$$l - p = \frac{1}{\pi} \Delta_L \arg f(\lambda) \Rightarrow l = \frac{1}{2} [n + \frac{1}{\pi} \Delta_L \arg f(\lambda)], \quad p = \frac{1}{2} [n - \frac{1}{\pi} \Delta_L \arg f(\lambda)]. \tag{11}$$

Proof is given in Turowicz 1967 [21, s. 68]. The further procedure when testing the stability of the system (1) or (7) show an example for $n = 3$.

4. Asymptotic stability condition of fractional system for $n=3$

Consider the continuous-time fractional system described by the equation (1) with $A \in R^{3 \times 3}$. The characteristic polynomial of matrix A is given by following equality:

$$w(\lambda, A) = f(\lambda) = \det[\lambda I - A] = \lambda^3 + b\lambda^2 + c\lambda + d \quad (12)$$

where $b, c, d \in R$. From (12) for $\lambda = y - b/3$ we have (this is the standard procedure for equation (9) with $n = 3$)

$$\tilde{f}(y) = y^3 + 3py + 2q, \quad 3p = \frac{3c - b^2}{3}, \quad 2q = \frac{2b^3}{27} - \frac{bc}{3} + d. \quad (13)$$

Let

$$\Delta = q^2 + p^3. \quad (14)$$

If $\Delta = q^2 + p^3 > 0$, thus $\lambda_1 = s \in R$, $\lambda_{2,3} = \gamma \pm j\omega$, $\gamma, \omega \in R$, where $f(\lambda_i) = 0$ for $i = 1, 2, 3$.

If $\Delta = q^2 + p^3 < 0$, then there are 3 different $\lambda_i \in R$, $i = 1, 2, 3$.

If $\Delta = q^2 + p^3 = 0$ and $p = q = 0$, then $\lambda_1 = \lambda_2 = \lambda_3 \in R$. If $\Delta = q^2 + p^3 = 0$ and $p^3 = -q^2 \neq 0$, then $\lambda_1 = s \in R$ and $\lambda_2 = \lambda_3 \neq s \in R$.

For $\Delta \leq 0$, then $\lambda_i \in R$. If $\lambda_i \in R$ and $\lambda_i < 0$, then system (1) with $n = 3$ is asymptotically stable. Appropriate conditions for b, c, d can be obtained from the Hurwitz theorem.

There exists a gain matrix K such that the closed-loop system (7) with matrix $A + BK$ has selected eigenvalues (in particular lying in the asymptotic stability region-see Fig. 1) if and only if $(A; B)$ is controllable.

Let $\Delta = q^2 + p^3 > 0$. thus $\lambda_1 = s$, $\lambda_{2,3} = \gamma \pm j\omega$ and from (12) we have

$$\begin{aligned} f(\lambda) &= \lambda^3 + b\lambda^2 + c\lambda + d = (\lambda - s)[(\lambda - \gamma)^2 + \omega^2] \\ b &= -(2\gamma + s), \quad c = \gamma^2 + \omega^2 + 2s\gamma, \quad d = -s(\gamma^2 + \omega^2). \end{aligned} \quad (15)$$

For us interesting is the case when $\Delta = q^2 + p^3 > 0$. Then we $\lambda_1 = s \in R$ and $\lambda_{2,3} = \gamma \pm j\omega$, $\gamma, \omega \in R$.

In this case the system (1) with $n = 3$ is asymptotically stable, then $\lambda_1 = s < 0$ and $\lambda_{2,3} = \gamma \pm j\omega$ is located in stability region (see Fig. 1), and then even to the right of the imaginary axis. Thus the system will be asymptotically stable if and only if $\lambda_1 = s \in R$ and $\lambda_2 = \gamma + j\omega$ they will lie on the left of the line

$$\lambda(t) = [1 + j(\operatorname{tg}(\alpha \pi / 2))]t, \quad t \in (-\infty, +\infty). \tag{16}$$

Example 1. Let $\alpha = 1/2$. Thus $\operatorname{tg}(\alpha \pi / 2) = 1$ and from (16) we have $\lambda(t) = (1 + j)t$. In this case from (15) we have

$$f((1 + j)t) = [-2t^3 + ct + d] + j[2t^3 + 2bt^2 + ct], \quad t \in (-\infty, +\infty). \tag{17}$$

and

$$\varphi(t) = \arg f((1 + j)t) = \operatorname{arc\,tg} \left(\frac{[2t^3 + 2bt^2 + ct]}{[-2t^3 + ct + d]} \right). \tag{18}$$

For $s = -1, \gamma = 1, \omega = 2$ we have $b = -1, c = 3, d = 5$ and from (18) we obtain

$$\varphi(t) = \arg f((1 + j)t) = \operatorname{arc\,tg} \left(\frac{2t^3 - 2t^2 + 3t}{-2t^3 + 3t + 5} \right). \tag{19}$$

From (19) notice that

$$\varphi(t) \rightarrow -\frac{\pi}{4} \quad \text{if} \quad t \rightarrow \pm\infty. \tag{20}$$

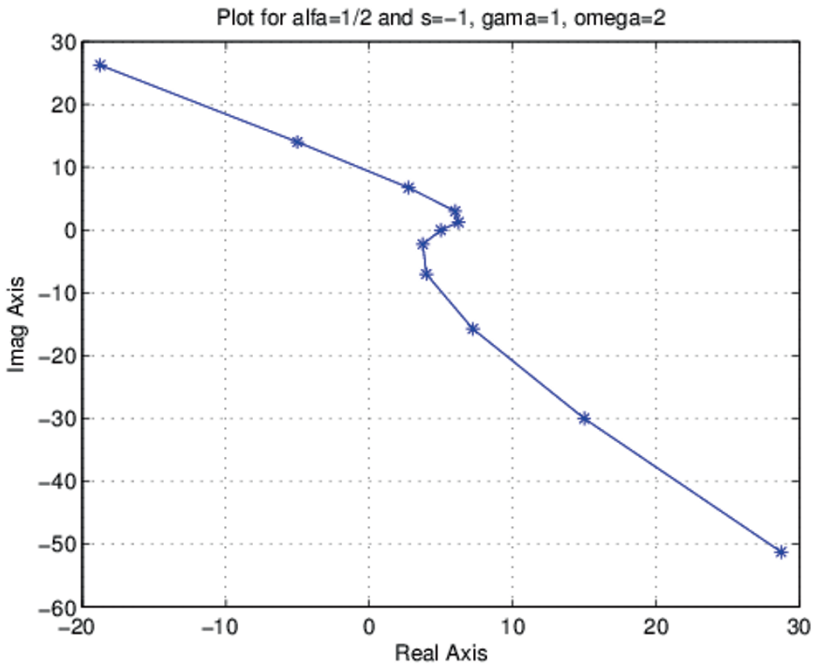


Fig.2. Plot for $\alpha = 1/2$ and $s = -1, \gamma = 1, \omega = 2$

Consider the polynomial (15) with roots $\lambda_1 = s$ and $\lambda_{2,3} = \gamma \pm j\omega$. Consider the sector shown in Fig.1 with $\alpha = 1/2$. If $s = -1, \gamma = 1, \omega = 2$, then in (11) we have $l = 2$ and $p = 1$. Thus from (11) we get $\Delta \arg_{-\infty < t < +\infty} f((1+j)t) = (l-p)\pi = \pi$. This is confirmed by situation shown in Fig. 2 and (20). In Fig. 2, with an increase in t moves from right to left.

In the Fig. 2-5, the points depicted with '*' denote accurate values, which were then interpolated using linear polynomials.

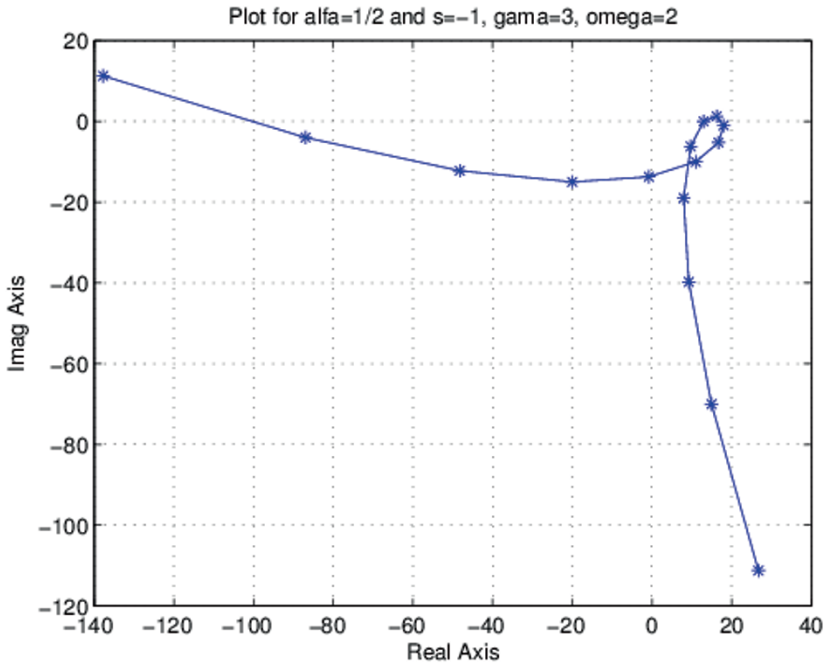


Fig. 3. Plot for $\alpha = 1/2$ and $s = -1, \gamma = 3, \omega = 2$

In Fig. 3 situation for $\alpha = 1/2$ and $s = -1, \gamma = 3, \omega = 2$ is shown. In this case we have $l = 1$ and $p = 2$. Thus $\Delta \arg_{-\infty < t < +\infty} f((1+j)t) = (l-p)\pi = -\pi$. This is confirmed by situation shown in Fig. 2 and (20).

In Fig. 4 situation for $\alpha = 1/2$ and $s = -1, \gamma = 3, \omega = 2$ is shown. In this case we have $l = 0$ and $p = 3$. Thus $\Delta \arg_{-\infty < t < +\infty} f((1+j)t) = (l-p)\pi = -3\pi$. This is confirmed by situation shown in Fig. 2 and (20).

In Fig. 5 situation for $\alpha = 1/2$ and $s = -1, \gamma = -3, \omega = 2$ is shown. In this case we have $l = 3$ and $p = 0$. Thus $\Delta \arg_{-\infty < t < +\infty} f((1+j)t) = (l-p)\pi = 3\pi$. This is confirmed by situation shown in Fig. 2 and (20).

Systems the asymptotically stable in Fig. 2 and Fig. 5 are shown (see asymptotic stability region of linear fractional-order system (5) - gray area).

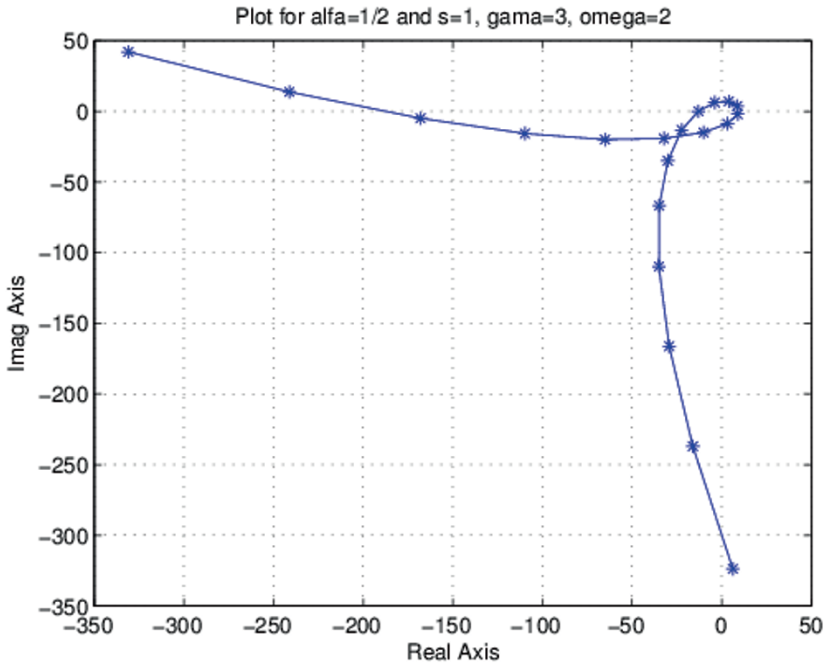


Fig. 4. Plot for $\alpha = 1/2$ and $s = 1, \gamma = 3, \omega = 2$

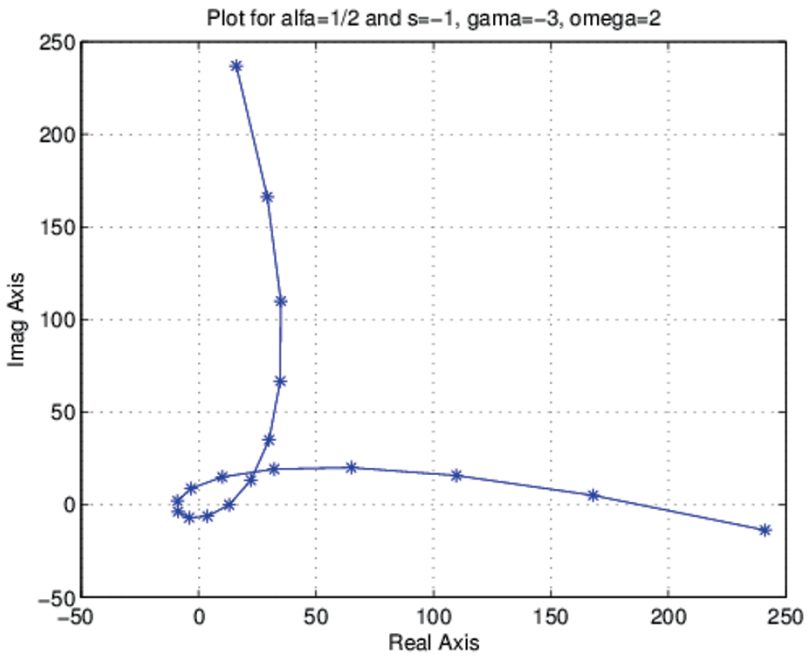


Fig. 5. Plot for $\alpha = 1/2$ and $s = -1, \gamma = -3, \omega = 2$

5. Concluding remarks

An in-depth stability analysis for $n=3$ was conducted in this work. It is worth notice that in practical applications, the model for $n=3$ is often sufficient to describe physical processes.

Similar analysis can be done for $n=4$ and $n>4$. In case if the coefficients a_i are real, the polynomial (9) has 0, 2, or four real roots (for $n=4$). The system (1) will be asymptotically stable if the conjugate pairs of roots with positive real parts will be in sector depicted in the figure 1. Sufficient conditions for asymptotic stability can be formulated with use of (11) from theorem 1. The parametric form of the line L is in (10).

The results from section 4 can be used for analysis of certain types of electrical systems [10, 11, 18]. For control of such systems, the rules from (5) and (6) can be used [3, 8].

Acknowledgment. This work was supported by the AGH (Poland) – the project No 11.11.120.817.

Literature

1. BUSŁOWICZ M., *Stability of State-Space Models of Linear Continuous-Time Fractional Order Systems*. Acta Mechanica et Automatica, 2011, vol. 5, no. 2, p. 15-22.
2. GANTMACHER F. R., *Theory of matrix*, [in Russian], 4 ed., Nauka, Moskwa 1988.
3. KACZOREK T., Fractional positive continuous-time linear systems and their reachability, Int. J. Appl. Math. Comput. Sci. 2008, Vol. 18, No. 2, pp.223-228.
4. KACZOREK T., *Positive linear systems with different fractional orders*, Bull. Pol. Acad.Sci. Tech., 2010, Vol. 58, No. 3, pp. 453-458.
5. KACZOREK T., *Positive fractional linear systems*. Pomiar Automatyka Robotyka, 2011, No. 2, 91-112.
6. KACZOREK T., *Selected Problems in Fractional Systems Theory*, Springer-Verlag 2011.
7. KACZOREK T., *Necessary and sufficient stability conditions of fractional positive continuous-time linear systems*. Acta Mechanica et Automatica, 2011, vol. 5, no. 2, p. 52-54.
8. KLAMKA J., Local Controllability of Fractional Discrete-Time Semilinear Systems. Acta Mechanica et Automatica, 2011, vol. 5, no. 2, p. 55-58.
9. MATIGNON D., *Stability result on fractional differential equations with applications to control processing*. In: IMACS-SMC Proceedings, Lille, France, 1996, pp. 963-968.
10. MITKOWSKI W., *Dynamical properties of Metzler systems*. Bulletin of the Polish Academy of Sciences, Technical Sciences, 2008, Vol. 56, No. 4, p. 309-312.
11. MITKOWSKI W.: Finite-dimensional approximations of distributed RC networks. *Bulletin of The Polish Academy of Sciences Technical Sciences*, Vol. 62, No. 2, 2014. pp. 263-269.

12. MITKOWSKI W., OBRĄCZKA A., *Simple identification of fractional differential equation*. Solid State Phenomena, 2012, Vol. 180, pp 331-338.
13. MITKOWSKI W., SKRUCH P., Fractional-order models of the supercapacitors in the form of RC ladder networks. *Bulletin of The Polish Academy of Sciences Technical Sciences*, Vol. 61, No. 3, 2013. pp. 581-587.
14. OBRĄCZKA A., MITKOWSKI W.: The comparison of parameter identification methods for fractional, partial differential equation. Diffusion and Defect Data – Solid State Data. Part B, *Solid State Phenomena*, 2014 vol. 210, s. 265–270
15. OLDHAM KB, SPANIER J., *The fractional calculus*. New York: Academic Press; 1974.
16. OPRZĘDKIEWICZ K., GAWIN E., MITKOWSKI W., *Modeling Heat Distribution With The Use of A Non-Integer Order, State Space Model*, Int. J. Appl. Math. Comput. Sci., 2016, Vol. 26, No. 4, 749–756
17. OSTALCZYK P., *Variable-, Fractional-Orders Closed-Loop Systems Description*. Acta Mechanica et Automatica, 2011, vol. 5, no. 2, p. 79-85.
18. PETRAS I., *A note on the fractional-order Chua's system*. Chaos, Solitons and Fractals, 2008, 38, p. 140-147.
19. PILLAI R.N., *On Mittag-Leffler functions and related distributions*, Ann. Inst. Statist. Math. 1990, Vol. 42, No. 1, pp. 157-161.
20. PODLUBNY I., *Fractional Differential Equations*, Academic Press, San Diego 1999.
21. TUROWICZ A., *Geometry of zeros of polynomials*. PWN, Warszawa 1967, in Polish.
22. TUROWICZ A., *Theory of Matrix*. 6 ed., 2005, AGH, Kraków, in Polish.
23. WEILBEER M., *Efficient Numerical Methods for Fractional Differential Equations and their Analytical*. Technischen Universität Braunschweig 2005, doktors Dissertation, 1-224.

Part VII
Advanced Robotics

Planning \mathbb{G}^3 -continuous paths for state-constrained mobile robots with bounded curvature of motion

Tomasz Gawron and Maciej Marcin Michałek *

Institute of Automation and Robotics,
Poznań University of Technology (PUT),
Piotrowo 3A, 60-965 Poznań, Poland,
tomasz.gawron@doctorate.put.poznan.pl

Abstract. The bounds on the mobile robot curvature of motion and path curvature continuity constraints usually result either from mechanical construction limitations or practical motion smoothness requirements. Most path planning primitives compatible with those constraints force planning algorithms to utilize costly numerical methods for computation of maximal path curvature or positional path constraints verification. In this paper a novel path primitive is proposed, which can be concatenated with the line and circle segments to form a path with bounded curvature such that its perfect realization by a unicycle robot guarantees continuous time-derivative of its curvature of motion. Satisfaction of prescribed curvature bounds and positional path constraints resulting from obstacles in the environment is formally guaranteed using explicit analytic formulas presented in the paper. It is shown that the proposed approach yields an arbitrarily precise \mathbb{G}^3 -continuous approximation of the Reeds-Shepp paths. Presented analysis is further utilized to formulate the global path planning problem in a continuous domain as a tractable optimization problem. Computational effectiveness of the proposed method has been additionally verified by quantitative comparison of constraint satisfaction checking speed with the η^3 -splines.

Keywords: path planning, state constraints, curvature constraints, mobile robot, unicycle

1 Introduction

1.1 Problem statement

Consider a kinematic unicycle with bounded curvature of motion $\kappa(t)$, i.e.,

$$\dot{\mathbf{q}} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ \cos \theta \\ \sin \theta \end{bmatrix} u_2 = [\mathbf{g}_1 \ \mathbf{g}_2(\theta)] \mathbf{u}, \quad (1)$$

$$\forall t \geq 0 \quad |\kappa(t)| \leq \kappa_b \quad \wedge \quad |\ddot{\kappa}(t)| < \infty, \quad |\kappa(t)| \triangleq \frac{|u_1(t)|}{|u_2(t)|}, \quad 0 < \kappa_b < \infty, \quad (2)$$

* This work was financially supported by the National Science Centre, Poland, as the research grant No. 2016/21/B/ST7/02259.

where $\mathbf{q} = [\theta \ x \ y]^\top = [\theta \ \bar{\mathbf{q}}^\top]^\top \in \mathcal{Q} = \mathbb{R}^3$ is a configuration vector, and $\mathbf{u} = [u_1 \ u_2]^\top \in \mathbb{U} \subset \mathbb{R}^2$ denotes a control input which consists of robot body angular velocity u_1 and longitudinal velocity u_2 of the guidance point $\bar{\mathbf{q}} = [x \ y]^\top$ shown in Fig. 1. Curvature bound κ_b is a prescribed constant resulting from robot motion task specification. In this paper, we consider the problem of planning non-parametrized paths represented as level-curves of particular functions, that is, paths implicitly defined by a set of positions $\mathcal{F} \triangleq \{\bar{\mathbf{q}} : F(\bar{\mathbf{q}}) = 0\}$, where $F(\bar{\mathbf{q}})$ is a design term (see e.g. [11] for control algorithms utilizing such paths). Given an initial configuration $\mathbf{q}_0 = \mathbf{q}(0)$ and reference configuration \mathbf{q}_d , one must plan a path between \mathbf{q}_0 and \mathbf{q}_d which is admissible for system (1) under input constraints (2) and collision-free, that is $\mathcal{F} \subseteq \mathcal{P}_{free}$, where \mathcal{P}_{free} is a known set of admissible robot positions given by a sequence of convex polygons

$$\mathcal{P}_{free} \triangleq \{\mathcal{P}_j\}_{j=0}^M, \quad M > 0. \quad (3)$$

We assume that subsequent polygons share at least one edge and shall be visited sequentially. Furthermore, a perfect realization of the planned path by system (1) must result in $\kappa(t=0) = 0$ and $\kappa(t=t_d) = 0$, where t_d is such that $\mathbf{q}(t_d) = \mathbf{q}_d$.

Remark 1. Note that function F can be planned as a sequence of particular functions f_i , $i = 1, 2, \dots, N$, corresponding to a sequence of path segments assumed to be realized by switching from f_i to f_{i+1} in the vicinity of a specific path segment endpoint. This allows for paths with reversals (switching between forward and backward robot motion).

Remark 2. Note that (2) implies \mathbb{G}^3 -continuity of admissible reference paths, i.e. continuity of $\kappa(t)$ must be guaranteed when the planned path is perfectly followed by system (1). Usually, weaker forms of geometric continuity such as \mathbb{G}^2 -continuity (continuity of path curvature) and \mathbb{G}^1 -continuity (continuity of path tangents) are considered in the literature.

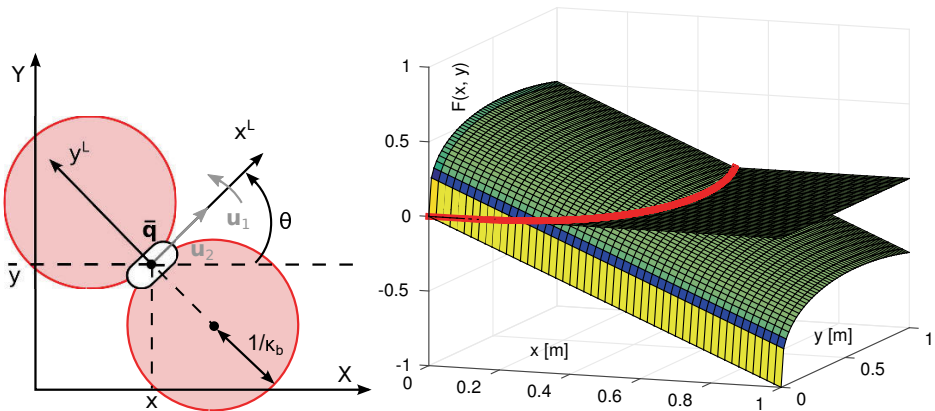


Fig. 1. A unicycle with bounded curvature of motion (left-hand side) and level-curve representation of segment T (cf. (4)) with path shown in red (right-hand side).

The path planning problem stated above is vitally important in robotic applications. The kinematic system (1) serves as a generic robot-body model and captures the most important characteristics of nonholonomic wheeled mobile robots and certain aerial vehicles flying at constant altitudes. Input constraints (2) often result from specific requirements of motion smoothness desired for specific applications (i.e., transport of a heavy payload, vision-based localization). They can be also a consequence of robot mechanical construction (i.e., bounded steering angles present in car-like robots and articulation angles in N-trailer robots, limitations of UAV constructions). \mathbb{G}^3 -continuity of a reference path is also desirable during motion control. For example, it is required to ensure continuous reference velocity of steering angle of car-like robots. Constraints represented by \mathcal{P}_{free} restrain the planned path to an obstacle-free subset of environment.

1.2 Related work

Given its importance, the problem considered in this paper has been widely studied in the literature. For the sake of brevity, we recall only selected representative works. In the seminal work [12], a set of 46 \mathbb{G}^1 -continuous shortest paths of bounded curvature was characterized under the assumption of obstacle-free environment. In [1] this result was extended to an environment bounded by a convex polygon with the restriction of $u_2 \geq 0$. A path planner for cluttered environments utilizing Reeds-Shepp paths from [12] has been developed, e.g., in [3]. To increase robot motion smoothness and improve transients during motion execution, various solutions for planning \mathbb{G}^2 -continuous paths have been developed. Specific Bezier curves of bounded curvature were utilized in [14] and [4]. Clothoid-based \mathbb{G}^2 -continuous approximations of Reeds-Shepp paths were presented in [5] and [2]. \mathbb{G}^3 -continuous paths represented by polynomials have been applied successfully in [9], where satisfaction of curvature bound κ_b and position constraints was ensured in discrete domain using numerical methods. A promising approach for continuous domain guarantees of collision-free polynomial spline-based paths was shown in [8], however it does not provide a method for constraining path curvature and optimization of path length. Worth noting, that in the vast majority of literature only parametric paths are considered, while the possibility of planning non-parametrized paths is rarely considered. It was shown in, e.g. [11], that such a path representation is beneficial for control purposes, because the shortest (orthogonal) distance between the robot and a reference path does not need to be determined.

In this paper we propose an arbitrarily precise \mathbb{G}^3 -continuous approximation of the Reeds-Shepp paths with analytically derived properties utilized for ensuring satisfaction of curvature constraints (2) and computing exact distances between the path and boundaries of the environment defined by set \mathcal{P}_{free} . Contrary to methods available in the literature, the proposed approach gives all the named guarantees simultaneously and in continuous domain, as opposed to various methods relying on discretization of parametric paths. Such guarantees directly affect the planning process. Analytic checking of constraints satisfaction is exact and an order of magnitude faster relative to the speed of numerical methods. Moreover, thanks to the analytical relations utilized in our approach, the global knowledge about path planning problem structure can be utilized by algorithms such as branch-and-cut proposed in [13] to obtain globally optimal solutions in a continuous domain. This seems to be extremely hard to achieve in

case of path primitives for which constraint satisfaction must be checked numerically, such as the ones used in [9] and [5]. A more detailed qualitative comparison of the proposed method with other path primitives is given in Table 1.

Table 1. Qualitative comparison of the proposed method with selected path primitives available in the literature.

path primitive	continuity	bounded curvature	$\bar{q} \in \mathcal{P}_{free}$ checking	length known
Reeds-Shepp	\mathbb{G}^1	yes	analytic	analytic
η^3 -splines	\mathbb{G}^3	no	numerical	numerical
clothoid-based	\mathbb{G}^2	yes	numerical	analytic
Bezier curves	\mathbb{G}^2	yes	numerical	numerical
proposed method	\mathbb{G}^3	yes	analytic	numerical

2 T -smoothed Reeds-Shepp paths

In this paper the results from [6] describing specific properties of the VFO feedback control dedicated to the waypoint-following task are leveraged to develop a method for constructing \mathbb{G}^3 -continuous reference paths for path-following controllers. The integral curve of VFO convergence vector field developed originally in [6] is characterized by several properties essential for obtaining \mathbb{G}^3 -continuous curves. However, the crucial method of path concatenation and path continuity analysis presented in this section were not considered in our previous works.

Following the form of path encoding taken from [12], we introduce a language for encoding paths with 3 words corresponding to path segments:

- $S(\mathbf{w}_1, \mathbf{w}_2)$ denotes a straight line connecting \mathbf{w}_1 with \mathbf{w}_2 ,
- $C(\mathbf{w}_1, \mathbf{w}_2, \kappa_c)$ denotes a circular arc of radius $1/\kappa_c$ connecting \mathbf{w}_1 with \mathbf{w}_2 ,
- $T(\mathbf{w}_1, \mathbf{w}_2)$ is a transition segment connecting \mathbf{w}_1 with \mathbf{w}_2 (defined later),

where $\mathbf{w}_k \triangleq [w_{k\theta} \ w_{kx} \ w_{ky}]^\top = [w_{k\theta} \ \bar{\mathbf{w}}_k^\top]^\top$, $k = 1, 2$, corresponds to coordinates of the beginning and the end of a path segment, respectively. In this framework all the Reeds-Shepp paths can be encoded as sequences of S and C segments. Note that depending on relative positions of \mathbf{w}_1 and \mathbf{w}_2 , the path may contain reversals (i.e., switching between forward and backward motion during perfect realization). Given a word sequence describing a particular Reeds-Shepp path connecting \mathbf{q}_0 and \mathbf{q}_d , it is well known that endpoints \mathbf{w}_k of path segments corresponding to particular words can be computed when curvature κ_c is known. Moreover, curvature of motion along such a path is discontinuous and bounded by $\pm\kappa_c$, its length is known exactly and its distance to boundaries of a convex polygon (cf. (3)) can be straightforwardly computed. Transition segments T shown in Fig. 2 are used to obtain a T -smoothed Reeds-Shepp path, which is \mathbb{G}^3 -continuous and admissible for system (1), as opposed to original Reeds-Shepp paths and their clothoid-based smoothings.

Definition 1. A T -smoothed Reeds-Shepp path is constructed by replacing every segment $C(\mathbf{w}_1, \mathbf{w}_2, \kappa_c)$ with a segment sequence $T(\mathbf{m}_1, \mathbf{m}_2) C(\mathbf{m}_2, \mathbf{m}_3, \kappa_c) T(\mathbf{m}_4, \mathbf{m}_3)$, where $\bar{\mathbf{m}}_k \triangleq [m_{k\theta} \ m_{kx} \ m_{ky}]^\top = [m_{k\theta} \ \bar{\mathbf{m}}_k^\top]^\top$, $k = 1, 2, 3, 4$ denote new endpoints of segments.

Coordinates of new endpoints \mathbf{m}_k , $k = 1, 2, 3, 4$ are determined using relations given in the remainder of this section. We will now show that introduction of specifically defined transition segments T results in preservation of desirable Reeds-Shepp paths properties, while ensuring \mathbb{G}^3 -continuity of a resultant path.

Let us denote variables expressed in a local coordinate frame fixed at point \mathbf{w}_1 by $(\cdot)^1$, that is, $\bar{\mathbf{q}}^1$ corresponds to $\bar{\mathbf{q}}$ expressed in coordinates of \mathbf{w}_1 . A transition segment $T(\mathbf{w}_1, \mathbf{w}_2)$ shown in Fig. 2 is defined by the following curve expressed in coordinates of endpoint \mathbf{w}_1 :

$$x^1 = f_t(y^1) = \begin{cases} \frac{-\text{sign}(x^1)|y^1|}{2} \left[\left(\frac{y^1}{p}\right)^\mu - \left(\frac{y^1}{p}\right)^{-\mu} \right] & \text{for } y^1 \neq 0, \\ 0 & \text{for } y^1 = 0, \end{cases} \quad (4)$$

$$p \triangleq w_{2y}^1 \exp\left(\frac{|\text{arsinh}(w_{2x}^1/w_{2y}^1)|}{\mu}\right), \quad w_{2y}^1 = \frac{K}{\kappa_c} y^*, \quad w_{2x}^1 = \frac{K}{\kappa_c} x^*, \quad (5)$$

with

$$K = \frac{y^* \left(-\mu + \mu^2 \frac{x^*}{r}\right)}{-(r)^2 (1 + \mu^2 - 2\mu \frac{x^*}{r})^{3/2}}, \quad r \triangleq \sqrt{(x^*)^2 + (y^*)^2}, \quad x^* = f_t(y^*),$$

$$y^* = \left(\left(\frac{4}{3} p_5 - p_3 \right)^{1/3} - \frac{-6\mu^3 + \mu^2 + 2\mu + 3}{(6\mu + 3)(\mu + 1)^2} + p_2 \right)^{1/2\mu},$$

while

$$p_1 = -6\mu^4 - 5\mu^3 + 3\mu^2 + 5\mu + 3, \quad p_2 = \frac{4\mu^2 (18\mu^4 + 3\mu^3 - 17\mu^2 - 11\mu + 7)}{9p_1 (\mu + 1)^4 (2\mu + 1)^2},$$

$$p_3 = \frac{(p_1)^3}{27(2\mu + 1)^3(\mu + 1)^9} + \frac{(2\mu - 1)(\mu - 1)^3}{(4\mu + 2)(\mu + 1)^3} - p_4,$$

$$p_4 = \frac{p_1(-6\mu^4 + 5\mu^3 + 3\mu^2 - 5\mu + 3)}{6(2\mu + 1)^2(\mu + 1)^6}, \quad p_5 = \sqrt{\frac{\mu^6(\mu - 1)^3(-36\mu^4 + 33\mu^2 - 29)}{(2\mu + 1)^4(\mu + 1)^9}},$$

where y^* is a rational function of μ , $\kappa_c \neq 0$ denotes the curvature of an adjacent circular segment, while $\mu \in (0.5, 1)$ is a design parameter influencing supremum value of $|\dot{\kappa}|$ along transition segment $T(\mathbf{w}_1, \mathbf{w}_2)$ and its length (see Fig. 2). Given a particular value of μ and transition segment endpoint \mathbf{w}_1 , one can instantly compute coordinates of the other endpoint \mathbf{w}_2 from (5). The curve representing segment $T(\mathbf{w}_1, \mathbf{w}_2)$ is given by (4) with particular parameter values computed from (5). The specific curve (4) was derived in [6]. It is an integral curve of the convergence vector field utilized in the VFO control law. Curve (4) is characterized by beneficial properties, which allow for guaranteeing of bounded curvature of transition segments T and \mathbb{G}^3 -continuity of resultant planned path. Such properties are not present in traditionally used path primitives such as, e.g., clothoids. Let us summarize those properties as follows (see [6] for details):

- P1. Curvature of motion $\kappa(\bar{q})$ along $T(\mathbf{w}_1, \mathbf{w}_2)$ is bounded as follows:
 $\forall \bar{q} \in T(\mathbf{w}_1, \mathbf{w}_2) \quad |\kappa(\bar{q})| \leq |\kappa_c|,$
- P2. Zero curvature of motion at the first endpoint: $\kappa(\bar{q}) \xrightarrow{\bar{q} \rightarrow \bar{w}_1} 0$ for $\mu \in (0.5, 1),$
- P3. Point \bar{w}_2 corresponds to the maximum of curvature $|\kappa(\bar{q})|$ along curve (4),
- P4. T segment degeneration: $\mathbf{w}_2 \xrightarrow{\mu \rightarrow 0.5} \mathbf{w}_1.$

Property P4 states that one can make segments T arbitrarily short by choosing $\mu \in (0.5, 1)$ arbitrarily close to 0.5. As μ tends to 0.5, T -smoothed paths tend to Reeds-Shepp paths. Thus, T -smoothed paths can approximate Reeds-Shepp paths with arbitrary precision selected by choice of parameter $\mu.$

Given the order of path segments determined by Definition 1, one immediately concludes \mathbb{G}^2 -continuity of T -smoothed paths from properties P2 and P3. What is more, T -smoothed paths have curvature bounded by $|\kappa_c|$ by the virtue of property P1 and properties of Reeds-Shepp paths. To prove \mathbb{G}^3 -continuity of T -smoothed paths, one must show that when system (1) perfectly follows a path segment $T(\mathbf{w}_1, \mathbf{w}_2),$ relations R1: $\dot{\kappa}(t) \xrightarrow{t \rightarrow t_{w_1}} 0$ and R2: $\dot{\kappa}(t) \xrightarrow{t \rightarrow t_{w_2}} 0$ are satisfied, where t_{w_1}, t_{w_2} denote time instants at which \bar{q} reaches endpoints $\mathbf{w}_1,$ and $\mathbf{w}_2,$ respectively. Proving such relations is sufficient, since for all segments of type S and C one has $\dot{\kappa}(t) \equiv 0$ and $\dot{\kappa}(t)$ is continuous for transition segment $T(\mathbf{w}_1, \mathbf{w}_2).$ Relation R2 is a consequence of property P3. To prove R1, we recall that orientation θ_a^1 tangent to segment $T(\mathbf{w}_1, \mathbf{w}_2)$ is given by (see [6] for details):

$$\theta_a^1 = \begin{cases} \text{Atan2}(\text{sign}(x^1) y^1, |x^1| - \mu \|\bar{q}^1\|) & \text{for } y^1 \neq 0, \\ 0 & \text{for } y^1 = 0. \end{cases} \quad (6)$$

When system (1) perfectly follows a path segment $T(\mathbf{w}_1, \mathbf{w}_2),$ one has $\dot{y} = u_2 \sin \theta_a$ from (1), thus $\dot{y}^1 = u_2 \sin \theta_a^1$ which, taking into account (4) and (6), means that $\dot{y}^1 \xrightarrow{t \rightarrow t_{w_1}} 0$ and $y^1 \xrightarrow{t \rightarrow t_{w_1}} 0.$ Let us now investigate $\dot{\kappa}(t)$ along $T(\mathbf{w}_1, \mathbf{w}_2)$ computed from the definition of curve's curvature and (4). After some algebra and substitution $\dot{y}^1 = u_2 \sin \theta_a^1,$ one arrives at:

$$\forall \bar{q} \in T(\mathbf{w}_1, \mathbf{w}_2) \quad \dot{\kappa}(t) = \frac{u_2 \chi_1(y^1, x^1) \chi_2((y^1)^{2\mu})}{(y^1)^{1-2\mu} [\chi_3((y^1)^{2\mu}) + c]}, \quad c = \text{const}, \quad (7)$$

with $\chi_k, k = 1, 2, 3$ denoting certain polynomials with non-zero constant terms. Thus, one concludes that for $\mu \in (0.5, 1),$ one has $\dot{\kappa}(t) \rightarrow 0$ as $y^1 \xrightarrow{t \rightarrow t_{w_1}} 0,$ which means that relation R1 is satisfied and, as a consequence, T -smoothed paths are \mathbb{G}^3 -continuous paths admissible for system (1) under constraints (2).

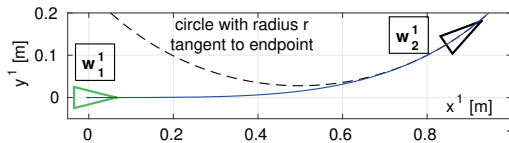


Fig. 2. Transition segment $T(\mathbf{w}_1, \mathbf{w}_2)$ for $\mu = 0.75$ and $r = 1/\kappa_c = 1/3 m.$

It remains to show how new path endpoints $\mathbf{m}_k, k = 1, 2, 3, 4$ introduced in Definition 1 can be computed when only $\mu,$ curvature $\kappa_c,$ initial/final configura-

tion pair and path structure (i.e., word sequence such as, e.g., $STCTSTCTS$) are given. Similarly to Reeds-Shepp paths, the path structure is predetermined or a given set of path structures (e.g., all T -smoothed Reeds-Shepp paths of no more than $W > 0$ words) is checked exhaustively during planning. The method of finding points m_k is also analogous to computation of Reeds-Shepp paths. Let us recall that given a structure of a Reeds-Shepp path, one finds the exact path representation by first computing circles tangent to initial and final configuration and later solving a system of equations to obtain straight-lines or (depending on a path structure) other circles bitangent to the initial two circles. In case of T -smoothed Reeds-Shepp paths one can follow a similar process. For brevity, we describe this process for the $STCTSTCTS$ path structure shown in Fig. 3 resulting from the CSC type paths. First, transition segments T resulting from path structure must be connected to initial and final configurations through S segments. Transition segments endpoints can be computed from (4), (6) and appropriate 2D homogenous coordinate transformations. Second, the placement of transition segments T determines equations of two circles tangent to the transition segments at their respective endpoints w_2 . Third, the radius R of a circle on which endpoint w_1 of a segment $T(w_1, w_2)$ adjacent to the middle segment S will lie (see Fig. 3 for visual interpretation) is given as $R = \|\bar{w}_2 - \bar{c}\|$, where w_2 is computed from (4) for an arbitrarily selected point on adjacent circular segment C , while \bar{c} denotes center of circle corresponding to adjacent segment C . Fourth, using (6) one computes orientation θ_{w_1} of endpoint of segment $T(w_1, w_2)$ adjacent to the middle segment S . Fifth, given the above constraints on positions and orientations of transition segments endpoints adjacent to the middle segment S , one can compute coordinates the unknown path segments endpoints.

3 Planning collision-free T -smoothed paths

Let us denote an infimum of signed distances for all path segments in a T -smoothed path and for all edges of the polygon \mathcal{P}_j (cf. (3)) by d_j . We assume that $d_j > 0$ implies that a path is fully contained in the polygon \mathcal{P}_j . For straight-line path segments S and circular path segments C it is straightforward to compute a signed distance between path segment and an edge of polygon \mathcal{P}_j . The infimum of signed distance between a transition segment T and a polygon edge $y = ax + b$ is an infimum of signed distances to this edge over points $\{\bar{w}_1, \bar{w}_2, [s_x^1 \ l s_x^1]^T\}$, where $s_x^1 = |p/l| \left(-1/|l| + \sqrt{1/(l)^2 + 1} \right)^{1/\mu}$, while $l = [a^1 \mu \sqrt{-\mu^2(a^1)^2 + (a^1)^2 + 1} - a^1] / [(a^1)^2 \mu^2 - 1]$. Thus, d_j can be computed analytically for any T -smoothed path and the path is collision-free iff $d_j > 0$.

We will now formulate global planning of a T -smoothed path as a nonlinear optimization problem by exploiting the fact that any T -smoothed path can be rewritten as a finite sequence of $STCTS$ paths with some segments degenerated to zero length. For example, taking endpoints of arc C as equal to each other, we obtain a valid $STTS$ path. Denoting by $P_i(\mu_i, \kappa_{ci}, \phi_i)$, a T -smoothed $STCTS$ path with $\mu = \mu_i$ for both transition segments, curvature κ_{ci} and circular segment C of arc angle ϕ_i we can formulate the optimization problem:

$$\min_{P_i, i=0,1,\dots,N} \left\{ c_t \sum_{i=0}^N \frac{\mathcal{L}(P_i(\mu_i, 1, \phi_i))}{|\kappa_{ci}|} + c_s \sum_{i=0}^N \mathcal{S}(P_i(\mu_i, 1, \phi_i)) |\kappa_{ci}| - c_d \mathcal{D} \right\},$$

subject to constraints:

$$\begin{aligned} \forall i = 0, 1, \dots, N \quad & 0.5 \leq \mu_i < 1, \quad |\phi_i| \leq \bar{M}(\mu_i - 0.5), \quad |\kappa_{ci}| \leq \kappa_b, \quad \mathcal{D} > 0, \\ \forall i = 0, 1, \dots, N - 1 \quad & P_i \text{ is connected to } P_{i+1}, \quad P_0 \text{ starts at } \mathbf{q}_0, \quad P_N \text{ ends at } \mathbf{q}_d, \end{aligned}$$

where $\bar{M} \gg 0$ is an arbitrary constant, $N > 0$ denotes a predetermined maximal number of *STCTS* paths from which the planned path can be composed, \mathcal{D} is the infimum of distance to boundaries of \mathcal{P}_{free} along the whole planned path (computed using relations for d_j), \mathcal{L} denotes path length of P_i , while \mathcal{S}_i corresponds to supremum of $|\dot{\kappa}|$ along P_i for perfect realization by system (1) with $u_2 = 1 \text{ m/s}$. The weights $c_l \geq 0$, $c_s \geq 0$, $c_d \geq 0$ influence planned path length, supremum of $|\dot{\kappa}|$, and distance to obstacles, respectively. Operator \mathcal{L} and function \mathcal{S} are not known explicitly and were approximated by polynomial functions of μ_i . Note that all the constraints are formulated without any approximations. Constraints on μ_i and ϕ_i ensure that a circular arc C of P_i is degenerated to zero length, when $\mu_i = 0.5$ degenerates transition segment T to zero length (cf. property P4), so that \mathbb{G}^3 -continuity of planned path is preserved, while redundant path segments can be degenerated to zero length. The optimization problem is continuous and has analytically defined constraints, thus we exploit its structure using the method proposed in [13], and implemented in YALMIP (see [10]).

4 Results of computational examples

Simulation results are shown in Fig. 3 and in Fig. 4. Green triangles denote initial configuration \mathbf{q}_0 , while red triangles denote prescribed final configuration \mathbf{q}_d . The curvature time-plots, were presented for perfect path realization by system (1) with $|u_2| = 1 \text{ m/s}$. In Fig. 3 a *STCTSTCTS* T -smoothed path in an obstacle-free environment is shown along with auxiliary circles illustrating the construction process of such a path. One can also investigate the curvature time-plot visible in Fig. 3 illustrating desirable curvature evolution thanks to the \mathbb{G}^3 -continuity. In Fig. 4 an exemplary path planned using global optimization in a moderately complex environment described by \mathcal{P}_{free} is shown. Design parameters were chosen as follows: $\bar{M} = 100$, $N = 6$, $c_l = 0.1$, $c_s = 2$, $c_d = 0.4$. It can be seen from Fig. 4 that, as anticipated, a combination of weights c_l , c_s c_d results in a path of characteristics different, than a length-minimizing one. One can tune the weights to plan safer, smoother paths at the expense of path length. The path was planned in 124 s. Note that planning time is highly dependent on chosen planning algorithm. In this case, the planning algorithm imposes a significant computational cost, because it is global, it operates in continuous domains, and its implementation was not optimized. To assess general computational cost of our method, we have also performed tests designed to compare the performance of analytic constraint satisfaction checking possible for T -smoothed paths with numerical checking necessary for η^3 -splines. Numerical checking was performed using `fmincon` procedure from MATLAB. 10000 random path segments in an environment \mathcal{P}_{free} comprising 2 triangles with combined area of 1 m^2 were tested. On average, checking for satisfaction of constraints (2) and computation of \mathcal{D} for a *STCTS* path was 9.3 times faster than numerical checking for η^3 -splines. Such results suggest that integration of our approach with planning algorithms present in the literature could result in significant computational cost decrease.

5 Final remarks

In this paper we have presented a new approach to smoothing Reeds-Shepp paths to G^3 -continuous paths with prescribed curvature bounds. The proposed method allows for arbitrarily precise approximating of the Reeds-Shepp paths and analytic checking of positional path constraints. Paths planned with our approach can be executed by the car-like and tractor-trailer robots with bounded articulation angles when a level-curve based controller is utilized. The robot-body dynamics and effects of tire friction forces present in larger robots can be considered by planning an appropriate longitudinal velocity profile along the path and utilizing bounds on path curvature as shown in, e.g., [7].

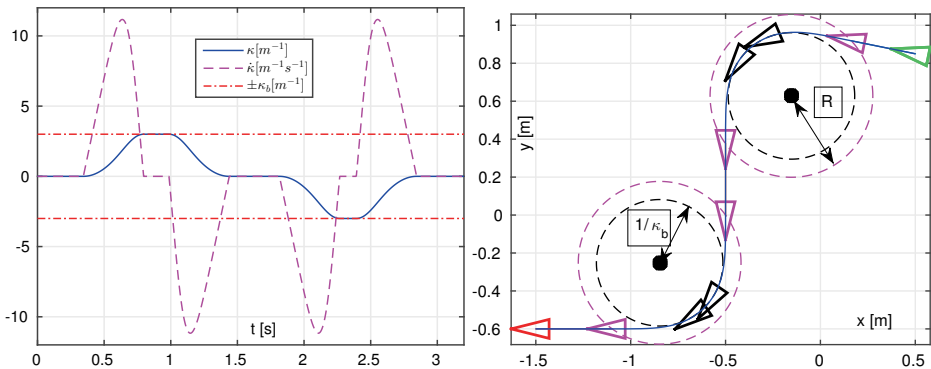


Fig. 3. An exemplary *STCTSTCTS* path planned in an obstacle-free environment with $\mu = 0.7$ and $\kappa_b = 3 \text{ m}^{-1}$ (right-hand side) along with corresponding curvature time-plot for its perfect realization with $u_2 = 1 \text{ m/s}$ (left-hand side). Black triangles denote endpoints of *C* segments, *T* segments span between black and magenta triangles.

References

1. Pankaj K. Agarwal, Therese Biedl, Sylvain Lazard, Steve Robbins, Subhash Suri, and Sue Whitesides. Curvature-constrained shortest paths in a convex polygon. *SIAM Journal on Computing*, 31(6):1814–1851, 2002.
2. E. Bakolas and P. Tsiotras. On the generation of nearly optimal, planar paths of bounded curvature and bounded curvature gradient. In *American Control Conference, 2009. ACC '09*, pages 385–390, St. Louis, Missouri, USA, June 2009.
3. Antonio Bicchi, Giuseppe Casalino, and Corrado Santilli. Planning shortest bounded-curvature paths for a class of nonholonomic vehicles among obstacles. *J.Intell. Robot. Syst.*, 16(4):387–405, 1996.
4. M. Elbanhawi, M. Simic, and R. Jazar. Continuous path smoothing for car-like robots using B-spline curves. *J. Intell. Robot. Syst.*, pages 1–34, 2015.
5. T. Fraichard and Alexis Scheuer. From Reeds and Shepp’s to continuous-curvature paths. *IEEE Trans. on Robotics*, 20(6):1025–1035, Dec 2004.
6. T. Gawron and M.M. Michałek. VFO stabilization of a unicycle robot with bounded curvature of motion. In *2015 Int. Workshop on Robot Motion and Control*, pages 263–268, Poznań, Poland, July 2015.
7. Jeong Hwan Jeon, R.V. Cowlagi, S.C. Peters, S. Karaman, E. Frazzoli, P. Tsiotras, and K. Iagnemma. Optimal motion planning with the half-car dynamical model for

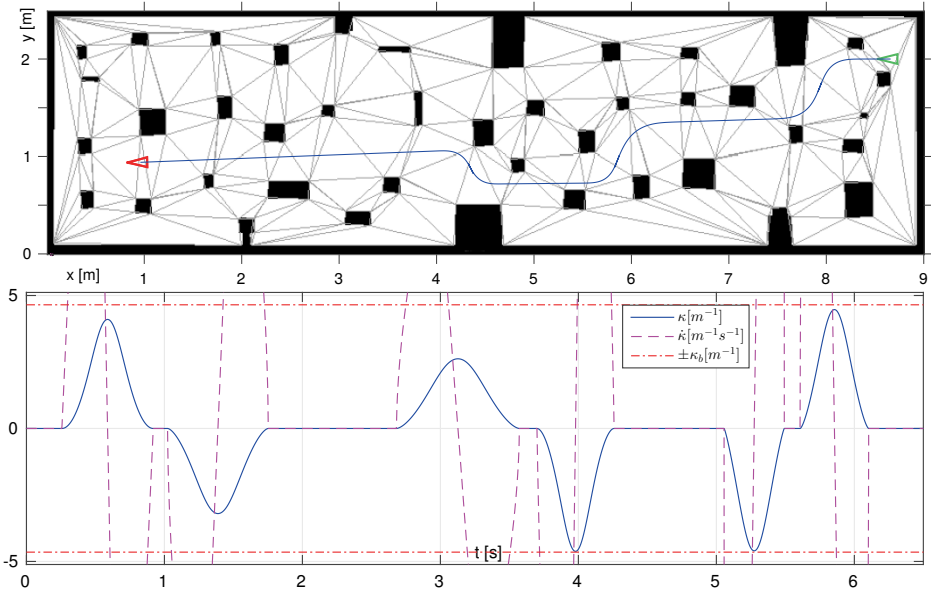


Fig. 4. A \mathbb{G}^3 -continuous path in a cluttered environment planned using global optimization. Curvature bound $\kappa_b = 4.65 \text{ m}^{-1}$ was chosen. Curvature time-plot was obtained for perfect path realization with $u_2 = 1 \text{ m/s}$.

autonomous high-speed driving. In *American Control Conference (ACC), 2013*, pages 188–193, Washington, DC, USA, June 2013.

8. B. Landry, R. Deits, P. R. Florence, and R. Tedrake. Aggressive quadrotor flight through cluttered environments using mixed integer programming. In *2016 IEEE Int. Conf. on Robot. and Autom.*, pages 1469–1475, Stockholm, Sweden, May 2016.
9. G. Lini, A. Piazzini, and L. Consolini. Multi-optimization of η^3 -splines for autonomous parking. In *Decision and Control and European Control Conference (CDC-ECC) 2011*, pages 6367–6372, Orlando, Florida, USA, Dec 2011.
10. J. Löfberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.
11. M. Michałek. A highly scalable path-following controller for N-trailers with off-axle hitching. *Control Engineering Practice*, 29:61–73, 2014.
12. J.A. Reeds and R.A. Shepp. Optimal paths for a car that goes both forward and backwards. *Pac. J. Mathematics*, 145(2):367–393, 1990.
13. Mohit Tawarmalani and Nikolaos V. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, 103(2):225–249, 2005.
14. K. Yang, S. Moon, S. Yoo, J. Kang, N. Doh, H. Kim, and S. Joo. Spline-based RRT path planner for non-holonomic robots. *J. Intell. Robot. Syst.*, 73(1-4):763–782, 2014.

Task allocation for multi-robot teams in dynamic environments

Maciej Hojda

Wroclaw University of Science and Technology,
Faculty of Computer Science and Management,
27 Wyb. Wyspianskiego St., 50-370 Wroclaw, Poland
maciej.hojda@pwr.edu.pl

Abstract. Multi-robot task allocation is a well known and widely researched decision-making problem that is difficult to solve in reasonable time even for small instances. Additional complexity is added by the fact that the parameters of the system may change over time, which happens either by external stimuli or by the task execution itself. One of the common causes behind these changes is the movement of executors or tasks. This paper tackles a problem of multi-task, multi-robot allocation in a such an environment. Formulated and solved is a specific decision-making problem. Performed is an experimental comparison of a dedicated solution algorithm with known methods for the more general Multidimensional Knapsack and Covering problem. Empirical evaluation illustrates that a dedicated approach is competitive and often necessary, as the general approach proves to be too slow.

Keywords: multi-robot task allocation, mobile routing, multidimensional knapsack and covering problem

1 Introduction

Multi-robot systems are expected to quickly and efficiently solve a wide variety of problems. Robots (or executors) are assigned to tasks for execution, and while complex tasks require cooperation of many robots, those executors which are more capable can perform multiple tasks on their own. Autonomous cooperating robots cover a wide spectrum of applications, including: data collection, inspection, monitoring, production and military use [CV,CM,Ho1,TI]. Particularly, of growing interest are problems when tasks are spatially distributed and movement is required in order to perform them [Ho2,LLFNS,MPA,RSSH,TLLB]. Furthermore, the executors are often equipped with multiple tools, such as multiple means of communication [JS,YMHC] or can be selected to perform tasks with varying levels of intensity [MPA,SBCFTW]. Varying modes of execution allow to fine-tune the execution process through more efficient use of available resources. As the tasks and executors grow more diverse, so increases the complexity of single-robot control and management of robotic teams. This paper focuses on the latter aspect by providing a method of joint task allocation and routing in a spatially distributed multi-robot environment.

Formulated and solved is a problem of multi-robot task allocation with mobile executors. The goal is to assign robots to tasks and determine the order of task execution for each executor. In [Ho1] this specific problem was proven to be NP-hard even when formulated in its feasibility version. The approach was to solve a substitutive problem instead. This paper improves upon that result by adapting the solution algorithm to solve the stated problem directly. A comparison with known solution algorithms for the more general Multidimensional Knapsack and Covering Problem [AHL,HALG] emphasises the benefits of a dedicated solution algorithm, what is visible in both the execution time and the quality of the obtained solution.

This paper is divided into five sections: the introduction, the problem formulation, the solution algorithm presentation, the empirical evaluation and the conclusions.

2 Problem formulation

There are I executors, J tasks and L modes, indexes of which are given as follows: executors $\mathbf{I} \triangleq \{1, 2, \dots, I\}$, tasks $\mathbf{J} \triangleq \{1, 2, \dots, J\}$, modes $\mathbf{K} \triangleq \{1, 2, \dots, L\}$. Decision variable defines the assignment of tasks to executor and the order of execution and is denoted by $x \triangleq [x_{i,j,k,l}]_{i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{J}, l \in \mathbf{K}}$ with elements $x_{i,j,k,l} = 1$ if task k is performed by executor i directly after task j and in mode l (0 if otherwise). Binary requirement is formally given as follows

$$\forall i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{J}, l \in \mathbf{K} \quad x_{i,j,k,l} \in \{0, 1\}. \quad (1)$$

Depending on the assignment, different amounts of resources are spent. As $e_{i,k,l}$ denoted is the cost at which the i th executor performs k th task in l th mode. The portion of the task k that is completed throughout this execution is denoted as $\eta_{i,k,l}$. Transitions between tasks cost $\mu_{i,j,k}$ for i th executor performing task k directly after task j . Those values are assumed non-negative, i.e. $e_{i,k,l} \geq 0, \eta_{i,k,l} \geq 0, \mu_{i,j,k} \geq 0$. A shorthand is defined as $e \triangleq [e_{i,k,l}]_{i \in \mathbf{I}, k \in \mathbf{J}, l \in \mathbf{K}}$, $\eta \triangleq [\eta_{i,k,l}]_{i \in \mathbf{I}, k \in \mathbf{J}, l \in \mathbf{K}}$, $\mu \triangleq [\mu_{i,j,k}]_{i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{J}}$. The constraints are divided into two sets of allocation-specific and routing-specific requirements.

First set ensures that at most one mode of execution is allowed for each executor-task pair, that task is completed when its completion rate of E has been achieved, the amount of resource F for each executor is limited:

$$\forall i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{J} \quad \sum_{l \in \mathbf{K}} x_{i,j,k,l} \leq 1, \quad (2)$$

$$\forall k \in \mathbf{J} \setminus \{1\} \quad \sum_{i \in \mathbf{I}, j \in \mathbf{J}, l \in \mathbf{K}} \eta_{i,k,l} x_{i,j,k,l} \geq E, \quad (3)$$

$$\forall i \in \mathbf{I} \quad \sum_{j \in \mathbf{J}, k \in \mathbf{J}, l \in \mathbf{K}} (e_{i,k,l} + \mu_{i,j,k}) x_{i,j,k,l} \leq F. \quad (4)$$

It is assumed that $E > 0$ and $F > 0$. Furthermore, task $j = 1$ is treated as a depot (starting and ending location for all executors) therefore $\eta_{i,1,l} = e_{i,1,l} = 0$.

Second set ensures an uniform starting location for each executor, connectivity of the routes, lack of loops and lack of subcycles:

$$\forall i \in \mathbf{I} \quad \sum_{k \in \mathbf{J}, l \in \mathbf{K}} x_{i,1,k,l} = 1, \quad (5)$$

$$\forall i \in \mathbf{I}, k \in \mathbf{J} \quad \sum_{j \in \mathbf{J}, l \in \mathbf{K}} x_{i,j,k,l} = \sum_{j \in \mathbf{J}, l \in \mathbf{K}} x_{i,k,j,l}, \quad (6)$$

$$\forall i \in \mathbf{I}, k \in \mathbf{J} \quad \sum_{j \in \mathbf{J}, l \in \mathbf{K}} x_{i,j,k,l} \leq 1, \quad (7)$$

$$\begin{aligned} \forall i \in \mathbf{I} \forall S \subset \mathbf{J} \wedge S \neq \emptyset \wedge S \neq \mathbf{J} \quad & \sum_{j \in S, k \in S, l \in \mathbf{K}} x_{i,j,k,l} \leq \\ & |S| - 1 + \frac{J}{J - |S|} - \frac{1}{J - |S|} \sum_{j \in \mathbf{J}, k \in \mathbf{J}, l \in \mathbf{K}} x_{i,j,k,l}. \end{aligned} \quad (8)$$

These requirements do not prohibit some executors from avoiding selected tasks entirely but ensure that every executor visits every task at most once on their route.

Movement and task execution requires resource spendings and this generalized cost is encapsulated in the quality criterion given as follows

$$Q(x) \triangleq \sum_{i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{J}, l \in \mathbf{K}} x_{i,j,k,l} (e_{i,k,l} + \mu_{i,j,k}). \quad (9)$$

The main problem of this paper, which is simultaneous task allocation and routing, can be now defined.

Problem 1 (TAR – task allocation and routing).

Given: $\mathbf{I}, \mathbf{J}, \mathbf{K}, e, \eta, \mu, E, F$

Find: x^* where

$$x^* \triangleq \arg \min_{x \in \mathbf{D}} Q(x), \quad (10)$$

$$\mathbf{D} \triangleq \{x \in \mathbf{X} \triangleq \mathbf{R}_{i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{J}, l \in \mathbf{K}} : (1) \wedge \dots \wedge (8)\}. \quad (11)$$

As was shown in [Ho1] this problem is NP-hard even in its feasibility version (in fact, just the allocation part is).

3 Solution Algorithm

Solution algorithm designed for TAR is based on the idea of a multi-stage functional decomposition. The original problem is divided into mult(-robot) task allocation MTA and mobile routing (MR) problems. The MTA itself is further

divided into several single-task relaxed allocation problems (STA) and relaxed task allocation problems (RTA). Individual problems and steps of the decomposition are given formally as follows. We recall the individual problems and algorithms after [Ho1, Ho2].

To define MTA, let the decision variable, which determines the allocation of tasks to executors, is denoted as $\check{x} \triangleq [\check{x}_{i,k,l}]_{i \in \mathbf{I}, k \in \mathbf{J}, l \in \mathbf{K}}$, where $\check{x}_{i,k,l} = 1$ if executor i is assigned to task k in mode l (0 if otherwise).

Problem 2 (MTA – multi-task allocation).

Given: $\mathbf{I}, \mathbf{J}, \mathbf{K}, \eta, e, E, F$

Find: \check{x}^* where

$$\check{x}^* \triangleq \arg \min_{\check{x} \in \check{\mathbf{D}}} \check{Q}(\check{x}), \quad \check{Q}(\check{x}) \triangleq \sum_{i \in \mathbf{I}, k \in \mathbf{J}, l \in \mathbf{K}} \check{x}_{i,k,l} e_{i,k,l}, \quad (12)$$

$$\check{\mathbf{D}} \triangleq \{\check{x} \in \check{\mathbf{X}} \triangleq \times_{i \in \mathbf{I}, k \in \mathbf{J}, l \in \mathbf{K}} \mathbf{R} : (14) \wedge \dots \wedge (18)\}. \quad (13)$$

$$\forall i \in \mathbf{I}, k \in \mathbf{J}, l \in \mathbf{K} \quad \check{x}_{i,k,l} \in \{0, 1\}, \quad (14)$$

$$\forall i \in \mathbf{I}, k \in \mathbf{J} \quad \sum_{l \in \mathbf{K}} \check{x}_{i,k,l} \leq 1, \quad (15)$$

$$\forall k \in \mathbf{J} \setminus \{1\} \quad \sum_{i \in \mathbf{I}, l \in \mathbf{K}} \check{x}_{i,k,l} \eta_{i,k,l} \geq E, \quad (16)$$

$$\forall i \in \mathbf{I} \quad \sum_{k \in \mathbf{J}, l \in \mathbf{K}} \check{x}_{i,k,l} e_{i,k,l} \leq F, \quad (17)$$

$$\forall i \in \mathbf{I}, l \in \mathbf{K} \quad \check{x}_{i,1,l} = 1 \Leftrightarrow l = 1. \quad (18)$$

The RTA problem is formulated for a fixed task $k \in \mathbf{K}$. Let $\dot{x}_k \triangleq [\dot{x}_{i,k,l}]_{i \in \mathbf{I}, l \in \mathbf{K}}$ be a decision variable (for a fixed $k \in \mathbf{J}$) with binary elements $\dot{x}_{i,k,l} = 1$ if task k is performed in l th mode by i th executor. The problem of a single task allocation is given as follows.

Problem 3 (STA – single task allocation).

Given: $d > 0, k \in \mathbf{J}, E, F, \mathbf{I}, \eta, e, \check{x}$

Find: \dot{x}_k^* where

$$\dot{x}_k^* \triangleq \arg \min_{\dot{x}_k \in \dot{\mathbf{D}}_k} \sum_{i \in \mathbf{I}, l \in \mathbf{K}} \dot{x}_{i,k,l} e_{i,k,l} \quad (19)$$

$$\dot{\mathbf{D}}_k \triangleq \{\dot{x}_k \in \times_{i \in \mathbf{I}, l \in \mathbf{K}} \mathbf{R} : (21) \wedge \dots \wedge (25)\} \quad (20)$$

$$\forall i \in \mathbf{I}, l \in \mathbf{K} \quad \dot{x}_{i,k,l} \in \{0, 1\}, \quad (21)$$

$$\forall i \in \mathbf{I} \quad \sum_{l \in \mathbf{K}} \dot{x}_{i,k,l} \leq 1, \quad (22)$$

$$\sum_{i \in \mathbf{I}, l \in \mathbf{K}} \dot{x}_{i,k,l} \eta_{i,k,l} \geq E, \quad (23)$$

$$\forall i \in \mathbf{I} \quad \sum_{l \in \mathbf{K}} \dot{x}_{i,k,l} e_{i,k,l} \leq F, \quad (24)$$

$$\forall i \in \mathbf{I}, l \in \mathbf{K} \quad \dot{x}_{i,1,l} = 1 \Leftrightarrow l = 1. \quad (25)$$

Finally, we consider a relaxed task allocation problem RTA. The definition is as in STA with constraint (21) relaxed into

$$\forall i \in \mathbf{I}, l \in \mathbf{K} \quad \dot{x}_{i,k,l} \in [0, 1], \quad (26)$$

and an additional constraint

$$\forall i \in \mathbf{I}, l \in \mathbf{K} \quad \dot{x}_{i,k,l} e_{i,k,l} \leq \min\{F, d\}. \quad (27)$$

where d is a positive parameter.

Since RTA is a linear programming problem it can be solved optimally with linear programming methods. Let us denote as $RTA(d)$ the solution to such a problem for parameter d . This solution can be rounded to be a 2-approximate solution of STA using a rounding algorithm RA. Let us also denote as $RA(d)$ the result of rounding with RA of $RTA(d)$.

Algorithm 1 RA

- 1: Let $i := 1, \dot{x} = [0]_{i \in \mathbf{I}, l \in \mathbf{K}}$
 - 2: **if** exists only one $l \in \mathbf{K}$ for which $\dot{x}_{i,k,l} \in (0, 1]$ **then**
 - 3: set $\dot{x}_{i,k,l} = 1$.
 - 4: **if** exist two different $l_1, l_2 \in \mathbf{K}$ for which $\dot{x}_{i,k,l_1}, \dot{x}_{i,k,l_2} \in (0, 1)$ and $\eta_{i,k,l_2} \geq \eta_{i,k,l_1}$ **then**
 - 5: set $\dot{x}_{i,k,l_1} = 1, \dot{x}_{i,k,l_2} = 0$.
 - return** \dot{x} .
-

To solve the STA problem, algorithm A1 is used. Let us denote as $A1(k)$ the result of A1 for the k th task.

Algorithm 2 A1

- 1: Set $\dot{x}_k := [0]_{i \in \mathbf{I}, l \in \mathbf{K}}, j := 0, \bar{d}^j := \sum_{i \in \mathbf{I}} \max_{l \in \mathbf{K}} e_{i,k,l}, \underline{d}^j := \min_{i \in \mathbf{I}, l \in \mathbf{K}} e_{i,k,l}$.
 - 2: Set $j := j + 1$.
 - 3: Set $d^j := \lfloor (\bar{d}^{j-1} + \underline{d}^{j-1})/2 \rfloor$.
 - 4: **if** $RTA(d^j)$ exists and if $\tilde{Q}[RA(d^j)] \leq 2d$ **then**
 - 5: set $\dot{x} := RA(d^j)$,
 - 6: set $\bar{d}^j := d^j - 1, \underline{d}^j := \underline{d}^{j-1}$,
 - 7: **else** set $\underline{d}^j := d^j + 1, \bar{d}^j := \bar{d}^{j-1}$.
 - 8: **if** $\underline{d}^j \leq \bar{d}^j$ **then**
 - 9: go to 2.
 - return** \dot{x} .
-

For solving the MTA problem, algorithm A2f is used.

Next formulated is the mobile routing problem. Given a solution of MTA \check{x} let $\check{\mathbf{J}}_i \triangleq \{k \in \mathbf{J} : \sum_{l \in \mathbf{K}} \check{x}_{i,k,l} = 1\}$ be a set of tasks to which the i th executor was

Algorithm 3 A2f

- 1: Set $k := 1$.
 - 2: $\check{x}_k \triangleq [\check{x}_{i,k,l}]_{i \in \mathbf{I}, l \in \mathbf{K}} := A1(k)$.
 - 3: If \check{x}_k does not exist then **return** no solution.
 - 4: $F_i := F_i - \sum_{i \in \mathbf{I}, l \in \mathbf{K}} \check{x}_{i,k,l} e_{i,k,l}$.
 - 5: $k := k + 1$.
 - 6: If $k \leq J$ go to 2. **return** $\check{x} \triangleq [\check{x}_{i,k,l}]_{i \in \mathbf{I}, k \in \mathbf{J}, l \in \mathbf{K}}$.
-

assigned. Let us denote the route of the i th executor as $y_i \triangleq [y_{i,j,k}]_{j \in \tilde{\mathbf{J}}, k \in \tilde{\mathbf{J}}_i}$ where $y_{i,j,k} = 1$ if task k is executed directly after task j (0 if otherwise). Problem MR is given as follows.

Problem 4 (MR – mobile routing).

Given: $i \in \mathbf{I}, \tilde{\mathbf{J}}_i, \mu$ Find: y_i^* where

$$y_i^* \triangleq \arg \min_{y_i \in \mathbf{D}_y^i} Q_y^i(y_i), \quad Q_y^i(y_i) \triangleq \sum_{j \in \tilde{\mathbf{J}}_i, k \in \tilde{\mathbf{J}}_i} y_{i,j,k} \mu_{i,j,k} \quad (28)$$

$$\mathbf{D}_y^i \triangleq \{y_i \in \mathbf{X}_{j \in \tilde{\mathbf{J}}_i, k \in \tilde{\mathbf{J}}_i} \mathbf{R} : (30) \wedge (31) \wedge (32) \wedge (33)\}. \quad (29)$$

$$\forall j \in \tilde{\mathbf{J}}_i, k \in \tilde{\mathbf{J}}_i \quad y_{i,j,k} \in \{0, 1\}, \quad (30)$$

$$\forall j \in \tilde{\mathbf{J}}_i \quad \sum_{k \in \tilde{\mathbf{J}}_i} y_{i,1,k} = 1, \quad (31)$$

$$\forall j \in \tilde{\mathbf{J}}_i \quad \sum_{j \in \tilde{\mathbf{J}}_i} y_{i,j,k} = \sum_{j \in \tilde{\mathbf{J}}_i} y_{i,k,j}, \quad (32)$$

$$\forall S \subset \tilde{\mathbf{J}}_i \wedge S \neq \emptyset \wedge S \neq \tilde{\mathbf{J}}_i \quad \sum_{j \in S, k \in S'} y_{i,j,k} \geq 2. \quad (33)$$

MR is in fact an instance of the Travelling Salesman Problem and can therefore be solved with dedicated methods. We will use the classic cheapest insertion algorithm. Given solutions \check{x} and y we can obtain a solution to TAR using the following equation

$$x_{i,j,k,l} \triangleq \check{y}_{i,j,k} \check{x}_{i,k,l}. \quad (34)$$

Finally, presented is the solution algorithm to the TAR problem. It is given as follows.

Algorithm 4 TARsol

- 1: Formulate and solve SMTA to obtain \check{x} .
 - 2: Formulate and solve MR under \check{x} . Obtain y .
 - 3: Use (34) to obtain x .
-

This dedicated method of solving TAR is compared empirically with algorithms for the Multidimensional Knapsack and Covering problem in the following section.

4 Empirical evaluation

In this section we compare the TARsol with the Adaptive Memory Search (AMS) [AHL] as well as the Alternating Control Tree (ACT) [HALG]. Both methods are of the iterative improvement type and were tested for varying limits on the maximum number of iterations. Since both methods provide means of solving a maximization problem, the objective function had to be negated. Additionally, for Alternating Control Tree, the constraint (4) in [HALG] had to be modified accordingly. For solving linear programming problems and integer programming problems a solver *GLPK* [glpk] was used.

We use the following abbreviations for tested algorithms:

- *AMS* – x which is the Adaptive Memory Search method run for up to x iterations,
- *ACT* – x which is the Alternating Control Tree method run for up to x iterations.

Algorithm AMS requires determination of several parameters. The evaluation of their influence on the solution is presented in Series 1 of the experiments.

Series 1.

Tested were the following parameters of the AMS algorithm (with default values): number of iterations N , tabu tenure ($base = 10$) and the weight update coefficients $\alpha_* = 1, \alpha_+ = 0, \beta_* = 1, \beta_+ = 0, w_{inc} = 0$. The experiment was done for the problem data: $I = 3, J = 4, L = 2, \mu_{i,1,j} = \mu_{i,j,1} = 1$, for $j, k \geq 2$: $\eta_{i,k,l} = l, e_{i,k,l} = 10 + l, \mu_{i,j,k} = |j - k|, F_i = 50, E = 2$.

Due to the random nature of AMS algorithm, every experiment was repeated 10 times and the results were averaged over all runs. They are presented in the corresponding tables where *Succ* is the number of experiments where a solution was found, Q is the average quality and T is the average execution time (in seconds).

Experiment 1.1 Tested is the number of iterations N . Results are presented in Table 1. For further tuning selected was the value of $N = 1000$.

N	10	20	50	100	200	500	1000	2000	5000
<i>Succ</i>	0	0	4	3	2	4	5	5	4
Q	-	-	108.25	82.33	84	106.75	85.6	90.2	69.5
T	-	-	0.56	1.06	2.06	5.09	10.04	20.22	38.09

Table 1. Results of Experiment 1.1. *Succ*, Q and T for varying N .

Experiment 1.2 Tested is the tabu tenure *base*. Results are presented in Table 2. For further testing, selected was the value of $base = 60$.

Experiment 1.3 Tested are the additive weighting coefficients $\alpha_+ = \beta_+$. Results are presented in Table 3. Selected were the values of $\alpha_+ = \beta_+ = 0$.

Experiment 1.4 Tested are the multiplicative weighting coefficients $\alpha_* = \beta_*$. Results are presented in Table 4. Selected were the values of $\alpha_* = \beta_* = 1$.

<i>base</i>	1	2	5	10	20	50	60	70	80
<i>Succ</i>	5	3	3	6	8	10	10	7	8
<i>Q</i>	87.2	75.67	103.33	80.67	89.75	90.1	84.7	80.43	88.86
<i>T</i>	9.67	9.67	9.88	10.23	10.2	12.76	13.99	13.56	14.59

Table 2. Results of Experiment 1.2. *Succ*, *Q* and *T* for varying *base*.

<i>base</i>	0	0.1	0.2	0.5	1	2	5
<i>Succ</i>	10	9	10	10	10	10	9
<i>Q</i>	80	81,67	81,7	81,6	85,3	84,7	85,11
<i>T</i>	12.8	12.74	12.9	13.17	13.46	14.19	14.71

Table 3. Results of Experiment 1.3. *Succ*, *Q* and *T* for varying α_+, β_+ .

Experiment 1.5 Tested are the weighting coefficient w_{inc} . Results are presented in Table 5. Selected was the value value of $w_{inc} = 0$.

It is clear from the experiments that that basic version of AMS barely manages to solve TAR even for a very simple instance. Increasing the tabu tenure resulted in a significant increase in number of successful experiments without a major change in the average execution time. It is also clear from Tables 3-5 that on-line modifications to weighting coefficients do not improve the results. Those parameters were kept at their default values.

Series 2.

In this series compare are TARsol, ACT and AMS for a selected instance of the TAR problem. Tested is the dependence of quality of the solution and the execution time on the number of tasks. Results are presented in Tables 6 and 7. The execution of *AMS* was repeated 5 times for each set of parameters and the results are the average. The number of successful executions is provided in brackets or not at all if every execution succeeded to provide a feasible solution. Parameters of the problem are as follows: $I = 3, J \in \{4, 5, 6\}, L = 2, \mu_{i,1,l} = \mu_{i,l,1} = 1, \text{ for } j, k \geq 2 : \eta_{i,k,l} = l, e_{i,k,l} = 10 + l, \mu_{i,j,k} = |j - k|, F_i = 100, E = 2.$

<i>base</i>	1	1.1	1.2	1.5	2
<i>Succ</i>	10	6	6	6	3
<i>Q</i>	85.4	74.67	96	91.33	81.67
<i>T</i>	13.77	11.94	12.11	11.79	11.82

Table 4. Results of Experiment 1.4. *Succ*, *Q* and *T* for varying α_*, β_* .

<i>base</i>	0	0.1	0.2	0.5	1
<i>Succ</i>	10	3	0	0	0
<i>Q</i>	90.2	78	-	-	-
<i>T</i>	13.96	12.94	-	-	-

Table 5. Results of Experiment 1.5. *Succ*, *Q* and *T* for varying w_{inc} .

<i>J</i>	<i>TARsol</i>	<i>ACT</i> – 200	<i>AMS</i> – 1000	<i>ACT</i> – 500	<i>AMS</i> – 2000
4	42	62	88.25 (4)	62	84.8
5	59	(0)	141 (1)	77	116.5 (4)
6	77	(0)	(0)	93	173 (1)

Table 6. Results of Series 2. *Q* for varying *J*.

<i>J</i>	<i>TARsol</i>	<i>ACT</i> – 200	<i>AMS</i> – 1000	<i>ACT</i> – 500	<i>AMS</i> – 2000
4	0.06	1.91	12.59 (4)	21.67	28.28
5	0.07	(0)	29.29 (1)	53.07	40.21 (4)
6	0.09	(0)	(0)	105.83	79.6 (1)

Table 7. Results of Series 2. *T* for varying *J*.

Concluding results of Series 2, we can observe a clear advantage of TARsol over AMS and ACT for the tested cases. Both the quality of obtained solutions and the execution time is better for TARsol, the latter by several orders of magnitude. The difference between AMS and ACT alone is similarly straightforward, ACT provides better solutions in a shorter span of time. Furthermore, for every tested case, the TARsol alone provided a feasible solution every time. Obtaining feasibility proved to be most difficult for the tested versions of the AMS algorithm where it often failed, even for small instances.

The main cause behind the observed results seems to lie in the size of the TAR problem. The number of variables and constraints is overwhelming to tackle all at once. Even ACT, which performs a decomposition on its own, did not perform comparably to TARsol.

5 Conclusion

The complexity of TAR prohibits a direct application of the AMS and ACT methods. Both, the execution time and the quality of the solution obtained by the dedicated algorithm TARsol are, on average, better. Furthermore, the algorithm tends to find a solution for cases when AMS and ACT fail. Such results are consistent with those obtained for the simpler MTA which was tackled in [Ho2]. Further study on the properties of TARsol and on other dedicated solution methods is necessary. Use of decomposition in conjunction with AMS and ACT should also be evaluated.

References

- [AHL] Arntzen, H., Hvattum L., Lokketangen A.: Adaptive memory search for multi-demand multidimensional knapsack problems. *Computers & Operations Research*

- 33, pp. 2508–2525. Elsevier (2005)
- [CV] Coltin B., Veloso M., Mobile Robot Task Allocation in Hybrid Wireless Sensor Networks, *Proceedings of International Conference on Intelligent Robots and Systems*, pp. 2932–2937 (2010)
- [CM] Correll N., Martinoli A., Multirobot Inspection of Industrial Machinery, *IEEE Robotics and Automation Magazine*, Vol. 16, pp. 103–112 (2009)
- [Ho1] Hojda M., Task allocation in robot systems with multi-modal capabilities, *IFAC-PapersOnLine, 15th IFAC Symposium on Information Control Problems in Manufacturing – INCOM 2015* Vol. 48, No. 3, pp. 2109–2114, Elsevier (2015)
- [Ho2] Hojda M., Comparison of algorithms for constrained multi-robot task allocation, *Advances in Systems Science : proceedings of the International Conference on Systems Science*, pp. 255–264, Springer (2017)
- [HALG] Hvattum L., Arntzen, H., Lokketangen A., Glover F.: Alternating control tree search for knapsack/covering problems. *Journal of Heuristics* 16, pp. 239–258. Springer (2008)
- [JS] Jung D., Savvides A., An Energy Efficiency Evaluation for Sensor Nodes with Multiple Processors, Radios and Sensor, *Proceedings of the 27th IEEE Conference on Computer Communications*, pp. 1112–1120 (2008)
- [LLFNS] Li X., Lille I., Falcon R., Nayak A., Stojmenovic I., Servicing Wireless Sensor Networks by Mobile Robots, *IEEE Communications Magazine*, Vol. 50, No. 7, pp. 147–154 (2012)
- [MPA] Melodia TI, Pompili D., Akyildiz I., Handling Mobility in Wireless Sensor and Actor Networks, *IEEE Transactions on Mobile Computing*, Vol. 10, No. 2, pp. 160–173 (2010)
- [RSSH] Rahimi M., Shah H., Sukhatme G., Heideman J., Studying the Feasibility of Energy Harvesting in a Mobile Sensor Network *Proceedings of ICRA '03. IEEE International Conference on Robotics and Automation, 2003*, Vol. 1, pp. 14–19 (2003)
- [SBCFTW] Shott B., Bajura M., Czarnaski J., Flidr J., Tho T., Wang L., A modular power-aware microsensor with 1000x dynamic power range, *Proceedings of Information Processing in Sensor Networks*, pp. 469–474 (2005)
- [TI] Tekdas O., Isler V., Using Mobile Robots to Harvest Data from Sensor Fields, *IEEE Wireless Communications*, Vol. 16, No. 1, pp. 22–28 (2009)
- [TLLB] Tirta Y., Li Z., Lu Y-H., Bagchi S., Efficient Collection of Sensor Data in Remote Fields Using Mobile Collectors, *Proceedings of 13th International Conference on Computer Communications and Networks*, Vol. 50, No. 7, pp. 147–154 (2012)
- [YMHC] Yong F., Mo S., Hackmann G., Chenyang L., Practical control of transmission power for Wireless Sensor Networks, *Proceedings of 20th IEEE International Conference on Network Protocols*, pp. 1–10 (2012)
- [glpk] GNU Linear Programming Kit, <https://www.gnu.org/software/glpk/>

Agent-Based Structures of Robot Systems

Cezary Zieliński, Tomasz Winiarski, Tomasz Kornuta

Warsaw University of Technology, Warsaw, Poland,

Email: {C.Zielinski, T.Winiarski, T.Kornuta}@ia.pw.edu.pl,

www home page: <http://robotics.ia.pw.edu.pl>

Abstract. Robot control systems structures based on agents are presented. Agents are classified into 8 categories, with the embodied agent being the most general one. Out of those agents diverse control systems are built. Single/multi-robot systems are considered, where robots can have single or multiple effectors. The presentation of the subject relies on already implemented systems.

Keywords: agent, embodied agent, multi-agent, multi-robot

1 Introduction

Although robot control systems are abundant (e.g. surveys classifying such systems [1–9]), no single method of presenting their structures has been established. The papers [10, 11], describing robot control system architectures, differentiate: 1) architectural structure, i.e. presentation of robot control system subsystems and their interconnections, and 2) architectural style, i.e. description of computational and communication concepts used. However [10] points out that in many implemented systems it is difficult to clearly define their architectural structure and style. Since the time of that publication not much has changed.

This work describes single/multi robot control system architectures in terms of agents. Agents, in turn, consist of effectors (real and virtual), receptors (real and virtual) and their control subsystems. The activities of those subsystems are described in terms of finite state machines switching behaviours parameterised by transition functions as well as terminal and error conditions, e.g. [12–15]. Systems are classified into single/multi-agent and single/multi-robot ones. The fact that robots can be composed of single or multiple effectors is also taken into account. Implemented examples of each case are presented and discussed.

2 Agents and their types

An agent, as defined by [16], is something that acts, but computer agents have other attributes, such as: autonomous control, environment perception, persistence over long time, adaptation to change or taking on another's goals. In robotics interaction with physical environment is emphasised, i.e. gathering information from the environment through receptors (sensors) and influencing the

environment through effectors (e.g. manipulators, grippers or other tools, wheels, tracks or legs). The necessary component of an agent is the control system, which is aware of the goal. To accomplish this goal it acquires information from receptors, makes decisions, and modifies environment state through its effectors. Such an agent is called an embodied agent, and is the subject of this paper.

Thus an agent a_j , where j is its designator, contains: real effectors E_j , influencing the environment, real receptors R_j (exteroceptors) collecting data about the environment, and the control system C_j governing the required behaviours. The control system is composed of three types of entities: virtual effectors e_j , virtual receptors r_j and the control subsystem c_j . The virtual effectors e_j and virtual receptors r_j present to the control subsystem c_j the effector and the environment respectively in a form that is appropriate for the purposes of control, i.e. they implement the ontology required by the control subsystem. The system can contain many real receptors $R_{j,l}$ and many virtual receptors $r_{j,k}$, where l and k are their respective designators. Similarly, it can contain many real effectors $E_{j,m}$ as well as many virtual effectors $e_{j,n}$, m and n being their respective designators. An agent should also be able to establish a two-way communication with other agents $a_{j'}$, $j \neq j'$. The resulting structure is presented in fig. 1.

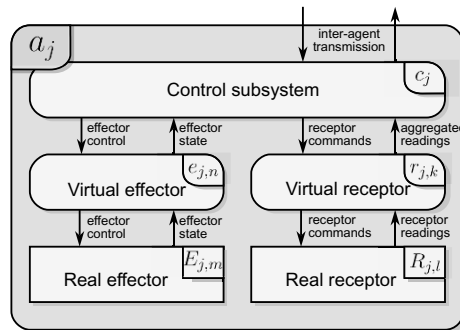


Fig. 1. General structure of an agent a_j

In general an agent a_j is represented by the four-tuple: (C, E, R, T) , where C represents the control subsystem c_j , E represents real effectors $E_{j,m}$ and virtual effectors $e_{j,n}$, R stands for real receptors $R_{j,l}$ and virtual receptors $r_{j,k}$, while T represents inter-agent communication (transmission). Any agent needs the C component to realise its functions, but can be deficient with respect to other components of the four-tuple. Thus 8 types of agents can be created [17] (tab. 1).

A C type agent is a purely computational entity without the possibility of interacting with the environment or other agents. An agent of CE type can influence the environment through its effectors, but cannot perceive it or interact with other agents. Such agents can be used as, e.g., industrial feeders. An agent which monitors the environment through its receptors, but cannot influence its state or modify the environment belongs to the CR type (e.g. a black box in an

aircraft). The C, CE and CR types of agents are of limited utility for robotics. The first truly useful agent is of the CT type. It can perform computations and make decisions on the behalf of other agents, thus hierarchical multi-agent systems will contain such an agent as a supervisor or coordinator. Even if it is not employed as a supervisor it can perform computationally intensive tasks for other agents, e.g. planning. The CER type agents are fully autonomous – they do not interact with other agents indirectly, but can perceive and influence the environment. Lack of direct communication with other agents does not preclude indirect communication using stigmergy [18]. The CRT agents can act as remote sensors for other agents, while the CET type agents act as teleoperated systems, however without sensor feedback, thus haptic systems cannot be constructed using this type of agents. The agents having the highest capabilities are of the CERT type. They can sense the environment, modify it and interact with other agents directly. All other types of agents are subtypes of the CERT type, thus it suffices to just consider the latter. This diversity of agents can be employed both for the design of single- and multi-robot systems. A system can be composed of one or many robots thus single/multi-robot systems emerge. A robot can be represented as a single or multi-agent structure. If a system consists of a single robot represented by a single agent then this is a single-agent system. If a system is composed of several robots or if a single robot is represented by several agents a multi-agent system results. Hence a multi-robot system cannot be a single-agent system. Both a robot and an agent can be single- or multi-effector. In the case of an agent its type acronym will contain an E then. If at least one agent or robot contains many effectors a multi-effector system is produced.

Type	Components	Function
C	$(C, \bullet, \bullet, \bullet)$	zombi
CE	(C, E, \bullet, \bullet)	blind agent
CR	(C, \bullet, R, \bullet)	monitoring agent
CT	(C, \bullet, \bullet, T)	computational agent
CER	(C, E, R, \bullet)	autonomous agent
CRT	(C, \bullet, R, T)	remote sensor agent
CET	(C, E, \bullet, T)	remotely controlled agent
CERT	(C, E, R, T)	full embodied agent

Table 1. Types of agents (\bullet represents the missing component)

3 Structure of robot systems

The majority of robot systems, that have been created up till now, has a fixed structure, but variable structure systems are also possible. In fixed structure

systems neither the constitutive agents and their interconnections nor the internal structure of the agents themselves change. In variable structure systems either the internal structure of the agent changes or the composing agents are exchanged. Further in the paper fixed structure systems will be considered. In the case of variable structure systems their structure evolves over time, but at any instant their structure is fixed, so the description of fixed structure systems structurally encompasses both categories. Nevertheless variable structure systems must exhibit different behaviours than fixed structured ones, e.g. [19].

3.1 Single-agent single-robot systems

A single robot having a single effector and represented by a single agent is a very common type of a system. One such example emerged when research on reactive torque control of redundant arms was conducted. This research led to the conclusion that initial arm kinematic configuration is vital for task executed with the use of operational space impedance control. To provide that the authors of [20] proposed a single agent (CER type) system with a single virtual effector executing two behaviours: joint space impedance control and operational space impedance control. The system used a Kuka LWR4+ manipulator.

The next example also introduces a single-agent (CER type) single-robot system, however this robot consisted of two effectors (manipulator and gripper). Service robots have to operate in human oriented environments, e.g. apartments with doors, both between rooms and mounted in cabinets. Those doors need to be opened by a robot. The robot named Velma [21] consisted of a 7-DOF KUKA LWR4+ impedance controlled manipulator and a position controlled BarrettHand gripper. Visual feedback was used to roughly localize doors and to plan the approach trajectory. Tactile feedback detected contact with the door and the handle. This way an exact location of the contact with the handle was determined. The contact between the hand and the door was maintained by the fingers of the gripper, which pushed the handle from the side.

3.2 Multi-agent single-robot systems

The mobile robot Rex is a single-robot multi-agent system [22]. The robot has a single effector – the mobile platform. It is composed of one CERT type, one CT type and one CRT type agent (fig. 2). The first one is the Locomotion agent a_{loc} – it interacts with the environment. The second one is the Ontology agent a_{ont} – upon request it generates tasks for a_{loc} in the form of a plan. The plan is formulated as a series of points to be traversed, corresponding to the positions specified on the map obtained from the third agent, i.e. the Map agent a_{map} , which gets the information about the position of the robot on the map from a motion capture system (a receptor) placed in the indoor environment. The Locomotion agent a_{loc} receives the current robot position estimates from the Map agent a_{map} . The Locomotion agent a_{loc} has one virtual effector controlling four motors driving the wheels of the mobile platform. This virtual effector treats the IMU, the force sensors located in the bumpers and four encoders as

proprioceptors. It implements model based control using a real-time state estimator. This agent also has a virtual receptor aggregating data from a stereo camera. The control subsystem is responsible for trajectory generation based on the plan received from a_{ont} and the information obtained from the virtual receptor about obstacles that are not present in the map (e.g. people walking by). The trajectory generation is based on the endogenous configuration space approach [23]. The three agent structure of the system is due to the fact that planning is a computationally intensive task, while localisation in relation to the map using the motion capture equipment is feasible only in the indoor environment, thus for outdoor environments significant modification of the agent a_{map} was anticipated. Hence those agents were created separate from the agent a_{loc} .

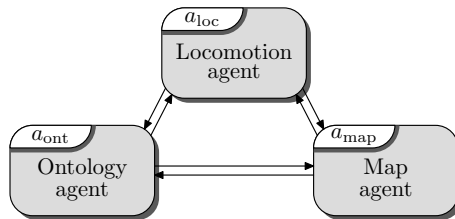


Fig. 2. General structure of the Rex robot system

Another example of multi-agent single-robot is a modified tele-operated robot performing exploratory tasks [24]. This robot had several effectors. Initially the operator used a joystick to guide the robot to the exploration area. This imposed a significant burden on the operator as the distance to be travelled had to be significant while the maximum velocity of the robot was rather low. Hence a control system assuming also autonomous approach to the exploration area had to be created. The task was defined in the following way: the operator has to indicate the goal of motion and once that is done the mobile robot has to drive to its vicinity avoiding obstacles. The following hardware has been chosen: mobile platform, a pan-tilt neck (both containing motors and encoders), a camera, a laser range-finder, ultrasonic sensors, touch sensors, inclinometers, GPS, compass, keyboard (instead of joystick) and a monitor. The operator console consisted of the keyboard and the monitor presenting the image from the camera. The camera and the laser rangefinder were mounted on the pan-tilt neck. The operator used the keyboard to point the coupled rangefinder and camera at the goal that was visible on the screen of the monitor. Once the goal was in the centre of the image its coordinates relative to the camera, and thus the robot were recorded. The robot had to navigate to that goal, avoiding obstacles. For that purpose it used the ultrasonic sensors, touch sensors, camera, GPS and compass. The inclinometer warned it about excessive inclination of the terrain. The task can be decomposed into three distinct parts: selection of the goal, autonomous navigation to the goal, and communication with the operator com-

manding the robot, either manually or supervising it in autonomous mode. Hence three agents have been created (fig. 3). Agent a_{goal} indicating the goal; agent a_{nav} navigating the platform to the selected goal, avoiding obstacles and excessive inclinations; and agent a_{coord} communicating with the operator. It should be noted that agents a_{nav} and a_{goal} are never active simultaneously – either the goal is being set or the robot navigates towards it. Thus the responsibility of the coordinator, agent a_{coord} , is to switch between the two.

The peculiarity of this system is that all three agents are of the CERT type. Each one of them has its own effectors and receptors. Agent a_{nav} uses GPS and compass as well as motor encoders for the purpose of self-localization, ultrasonic and touch sensors to avoid obstacles and inclinometers to avoid overturning. It receives from the agent a_{coord} the coordinates of the goal. All this information is required to safely navigate to the goal. Once the goal is reached agent a_{nav} processes manual control commands received via agent a_{coord} . Agent a_{coord} coordinates the whole system and uses the keyboard (which functions as a receptor) to listen to operator commands and displays the image from the camera (another receptor) on the monitor (playing the role of an effector). Hence all the three agents are of the CERT type.

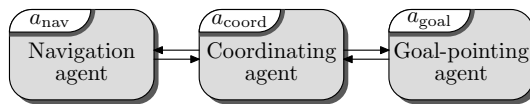


Fig. 3. General structure of the exploratory robot

3.3 Multi-agent multi-robot systems

To investigate how robots can cooperate through stigmergy (i.e. indirect communication through traces left in the environment) [18] a system consisting of two box pushing mobile robots and one robot broadcasting the box goal location has been created [17]. The benefit of creating stigmergic systems is that they are scalable. Robots can be added or extracted from the system without the necessity of modifying their software. In the case of the presented system each of the robots is represented by an agent that had a single effector – a mobile platform, receptors and a control subsystem. As the robots have to know the final location of the box and its current orientation this information was conveyed to them by still another robot represented by an agent a_{broad} broadcasting that information, thus a one-way inter-agent communication with that agent was necessary (fig. 4). This robot did not receive any feedback from the other two and broadcasted messages without knowing how many recipients there are, if any. Hence each of the robots was composed of a single CERT type agent. The two box pushing robots did not communicate directly with each other, so the number of such robots could be increased without the modification of the

software of box pushing robots and the broadcasting robot. Each box pushing robot was represented by an agent a_{box} . Each agent had a single virtual effector controlling the two wheel motors as well as performing direct and inverse kinematics. Moreover it had a virtual receptor aggregating information from a laser scanner, thus the orientation of the box in relation to the robot and the location of the corner of the box could be established. The control subsystem of each box-pushing agent a_{box} computed four feedback functions maintaining: correct box motion direction, right box pushing robot orientation, adequate relationship to the box corner and required pushing velocity. The results of the computation of the four functions were superposed to produce the resultant command for the virtual effector. This compound behaviour of the box-pushing robots caused the box to reach the desired location.



Fig. 4. General structure of the stigmergic box-pushing system

Another multi-agent multi-robot system, however with each robot having several effectors, each driven by a separate agent is presented in fig. 5. Aircraft and car industry require machining (e.g. drilling, milling) of diverse curved surfaces of components. As those components are made of thin sheets they are flexible, thus for the purpose of machining they have to be supported to become rigid. For that purpose fixtures are used. Each type of component needs a different fixture, which is costly both to manufacture and store. An alternative idea emerged, where instead of this multiplicity of fixtures a single system containing several robots translocating themselves under the machined component would be used [25, 26]. The robots rigidify the component only in the vicinity of the machine tool performing its task. As the tool moves the robots translocate themselves making the machined piece rigid along the trajectory of that tool. Such a system needs several robots that can translocate themselves, rise the supporting head to the required level and attach the head to the machined piece. The resulting system [27] consisted of a bench, having an equilateral triangle mesh of docking elements protruding from it, and several robots each consisting of: a mobile base having three legs located in the vertices of an equilateral triangle, a parallel kinematic machine (PKM) [28] acting as a manipulator and a head having the ability to either be soft or rigid [29]. The docking elements delivered electric power and compressed air to the robots through their legs. The mobile base docked the three legs to any three docking elements forming an equilateral triangle on the bench, thus acquiring rigid support, and then the manipulator moved the head to the required location. During this motion the head was soft, but once in place it was attached to the component surface by a vacuum sucker and then solidified, either by withdrawing the air from a soft bag hold-

ing sand particles or by applying a magnetic field to a magnetorheological fluid supporting pins touching the machined surface. The robot to translocate itself raised two legs and rotated around the one attached to the docking element in the bench. The angle of rotation was always a multiple of 60° . Once over the docking elements the two legs were lowered and attached by a docking locks.

The robotised fixture is an example of a multi-effector, multi-agent, multi-robot system [30,31]. The decomposition of the system was guided by the task that had to be accomplished and the fact that the devices constituting the system were developed separately, thus separate testing was required. Hence each robot, consisting of three effectors: mobile base controlled by agent a_{mb} , manipulator controlled by agent a_{pkm} and the head controlled by agent a_{head} fig. 5. Each of those agents was a CERT type agent, having one virtual effector each. The plan of motions [32,33] was delivered by the operator to the system coordinator a_{coord} , which was a CT agent. It controlled both the agent governing the activities of the bench a_{bench} of the CERT type and as many triplets of agents a_{mb} , a_{pkm} and a_{head} as there were robots in the system. In the test system only two robots were used, thus the system consisted of eight agents.

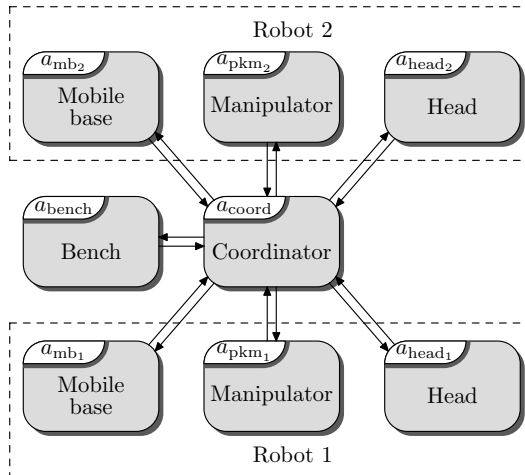


Fig. 5. General structure of the robotic fixturing system

4 Summary

The paper presents the structures of robot systems in terms of agents. Eight types of agents can be distinguished, however the CERT type is the most general one. Robots can have a single effector or several ones. A single robot can be represented as a single agent or as a multi-agent structure. The decomposition of a system into agents stems from the tasks the system has to execute. There is

no single method of decomposition – it depends on the experience of the designer, thus should be treated as an art.

References

1. Matarčić, M.J., Michaud, F.: Behavior-Based Systems. In: The Handbook of Robotics. Springer (June 2008) 891–909
2. Bulter, Z., Rizzi, A.: Distributed and cellular robots. In Khatib, O., Siciliano, B., eds.: Springer Handbook of Robotics. Springer (June 2008) 911–920
3. Parker, L.E.: Multiple mobile robot systems. In Khatib, O., Siciliano, B., eds.: Springer Handbook of Robotics. Springer (June 2008) 921–941
4. Yim, M., Shen, W.M., Salemi, B., Daniela Rus, Moll, M., Hod Lipson, Klavins, E., Chirikjian, G.S.: Modular self-reconfigurable robot systems [grand challenges of robotics]. *IEEE Robotics & Automation Magazine* **14**(1) (March 2007) 43–52
5. Matarčić, M.J.: Issues and approaches in the design of collective autonomous agents. *Robotics and Autonomous Systems* **16**(2) (1995) 321 – 331
6. Farinelli, A., Iocchi, L., Nardi, D.: Multirobot systems: a classification focused on coordination. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **34**(5) (Oct 2004) 2015–2028
7. Dudek, G., Jenkin, M.R.M., Milios, E., Wilkes, D.: A taxonomy for multi-agent robotics. *Autonomous Robots* **3**(4) (1996) 375–397
8. Doriya, R., Mishra, S., Gupta, S.: A brief survey and analysis of multi-robot communication and coordination. In: Computing, Communication Automation (ICCCA), 2015 International Conference on. (May 2015) 1014–1021
9. Chibani, A., Amirat, Y., Mohammed, S., Matson, E., Hagita, N., Barreto, M.: Ubiquitous robotics: Recent challenges and future trends. *Robotics and Autonomous Systems* **61**(11) (2013) 1162 – 1172
10. Kortenkamp, D., Simmons, R.: Robotic systems architectures and programming. In Khatib, O., Siciliano, B., eds.: Springer Handbook of Robotics. Springer (2008) 187–206
11. Coste-Maniere, E., Simmons, R.: Architecture, the backbone of robotic systems. In: Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on. Volume 1. (2000) 67–72 vol.1
12. Zieliński, C., Winiarski, T.: General specification of multi-robot control system structures. *Bulletin of the Polish Academy of Sciences – Technical Sciences* **58**(1) (2010) 15–28
13. Kornuta, T., Zieliński, C.: Robot control system design exemplified by multi-camera visual servoing. *Journal of Intelligent & Robotic Systems* **77**(3–4) (2015) 499–524
14. Zieliński, C., Kornuta, T., Winiarski, T.: A systematic method of designing control systems for service and field robots. In: 19-th IEEE International Conference on Methods and Models in Automation and Robotics, MMAR, IEEE (2014) 1–14
15. Zieliński, C., Kornuta, T.: Diagnostic requirements in multi-robot systems. In: Intelligent Systems in Technical and Medical Diagnostics. Volume 230. Springer (2014) 345–356
16. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall, Upper Saddle River, N.J. (1995)
17. Zieliński, C., Trojanek, P.: Stigmergic cooperation of autonomous robots. *Journal of Mechanism and Machine Theory* **44** (April 2009) 656–670

18. Bonabeau, E., Dorigo, M., Theraulaz, G.: *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, Oxford (1999)
19. Psomopoulos, F., Tsardoulis, E., Giokas, A., Zieliński, C., Prunet, V., Trochidis, I., Daney, D., Serrano, M., Courtes, L., Arampatzis, S., Mitkas, P.: Rapp system architecture. In: *IROS 2014 – Assistance and Service Robotics in a Human Environment*, Workshop in conjunction with IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, Illinois, September 14 (2014) 14–18
20. Winiarski, T., Banachowicz, K., Seredyński, D.: Two mode impedance control of Velma service robot redundant arm. In Szewczyk, R., Zieliński, C., Kaliczyńska, M., eds.: *Progress in Automation, Robotics and Measuring Techniques*. Vol. 2 Robotics. Volume 351 of *Advances in Intelligent Systems and Computing (AISC)*., Springer (2015) 319–328
21. Winiarski, T., Banachowicz, K., Seredyński, D.: Multi-sensory feedback control in door approaching and opening. In Filev, D., Jabłkowski, J., Kacprzyk, J., Krawczak, M., Popchev, I., Rutkowski, L., Sgurev, V., Sotirova, E., Szykarczyk, P., Zadrozny, S., eds.: *Intelligent Systems'2014*. Volume 323 of *Advances in Intelligent Systems and Computing*. Springer International Publishing (2015) 57–70
22. Janiak, M., Zieliński, C.: Control system architecture for the investigation of motion control algorithms on an example of the mobile platform rex. *Bulletin of the Polish Academy of Sciences – Technical Sciences* **63**(3) (2015) 667–678
23. Tchoń, K., Jakubiak, J.: Endogenous configuration space approach to mobile manipulators: A derivation and performance assessment of Jacobian inverse kinematics algorithms. *International Journal of Control* **76**(14) (2003) 1387–1419
24. Zieliński, C., Kornuta, T., Trojanek, P., Winiarski, T.: Method of Designing Autonomous Mobile Robot Control Systems. Part 2: An Example (in Polish). *Pomiary Automatyka Robotyka* (10) (2011) 84–91 (Metoda projektowania układów sterowania autonomicznych robotów mobilnych. Część 2. Przykład zastosowania).
25. Molfino, R., Zoppi, M., Zlatanov, D.: Reconfigurable swarm fixtures. In: *ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots*. (June 22–24 2009) 730–735
26. Leonardo, L., Zoppi, M., Xiong, L., Gagliardi, S., Molfino, R.: Developing a New Concept of Self Reconfigurable Intelligent Swarm Fixtures. (2012) 321–331
27. de Leonardo, L., Zoppi, M., Li, X., Zlatanov, D., Molfino, R.: Swarmifx: A multi-robot-based reconfigurable fixture. *Industrial Robot* (2013) 320—328
28. Neumann, K.: US patent number 4732525 (1988)
29. Gagliardi, S., Li, X., Zoppi, M., de Leonardo, L., Molfino, R.: Adaptable Fixturing Heads for Swarm Fixtures: Discussion of Two Designs. (2012)
30. Zieliński, C., Kornuta, T., Trojanek, P., Winiarski, T., Wałęcki, M.: Specification of a multi-agent robot-based reconfigurable fixture control system. *Robot Motion & Control 2011 (Lecture Notes in Control & Information Sciences)* **422** (2012) 171–182
31. Zieliński, C., Kasprzak, W., Kornuta, T., Szykiewicz, W., Trojanek, P., Wałęcki, M., Winiarski, T., Zielińska, T.: Control and programming of a multi-robot-based reconfigurable fixture. *Industrial Robot: An International Journal* **40**(4) (2013) 329–336
32. Szykiewicz, W., Zielińska, T., Kasprzak, W.: Robotized machining of big work pieces: Localization of supporting heads. *Frontiers of Mechanical Engineering in China* **5**(4) (2010) 357–369
33. Zielińska, T., Kasprzak, W., Szykiewicz, W., Zieliński, C.: Path planning for robotized mobile supports. *Journal of Mechanism and Machine Theory* **78** (2014) 51–64

Global path planning for a specialized autonomous robot for intrusion detection in wireless sensor networks (WSNs) using a new evolutionary algorithm

Piotr Bazydło*, Janusz Kacprzyk**,*** and Krzysztof Lasota*

*Research and Academic Computer Network (NASK)

ul. Kolska 12, 01-045 Warsaw, Poland

**Systems Research Institute, Polish Academy of Sciences

ul. Newelska 6, 01-447 Warsaw, Poland

***Industrial Research Institute for Automation and Measurements PIAP

Al. Jerozolimskie 202, 02-486 Warsaw, Poland

E-mails: piotr.bazydlo@nask.pl; kacprzyk@ibspan.waw.pl;

krzysztof.lasota@nask.pl

Summary. We present a new specialized evolutionary algorithm for the global path planning for mobile robots. We assume that multiple criteria are involved, notably the energy consumption, travel time and movement cost on dangerous areas, as well as constraints like the location obstacles, a limited minimal robot's turning angle, a maximal energy (related to the capacity of batteries) and maximal time of travel. The new algorithm involves some new problem specific evolutionary operations. The simulation results using the V-REP platform involve obstacles, different surfaces and dangerous areas. The results are very encouraging, much better than those obtained using the traditional meta-heuristics, notably the widely used genetic algorithm. Moreover, a novel explicit involvement of energy consumption as a key factor, provides much new insight from both a theoretical and even more practical points of view.

1 Introduction

We consider the following problem. There is a wireless sensor network (to be called WSN, for brevity). Such networks are widely used in many areas, exemplified by the military, industrial, agricultural, infrastructural, health care, to just name a few. Since, usually, nodes of such networks can exchange highly important, often security sensitive data, there may be some malicious attacks on those nodes. Obviously, all efforts to increase security are difficult and costly. Intrusion detection is therefore crucial and it can be performed through specialized mobile agents. The system considered here works basically as follows: there is a central unit that is meant to control mobile robots the purpose of which is intrusion detection in network nodes. From our point of view, the main task of the central unit is to determine the positions to be consecutively followed by the robots to find and report the consecutive endangered nodes.

Obviously, instead of the above mentioned intrusion detection problem in WSN, a much more intuitively appealing problem of, for instance, land mine detection in a battle field is substantially the same.

The problem considered, first of all, is related to the global path planning (GPP) which is, in general, a popular topic in mobile robotics and many effective and efficient algorithms are known. However, in our work we add to the performance function some additional terms related to, first of all, energy consumption and movement difficulties related to, for instance, a difficult terrain. Needless to say that the problem of GPP is a computationally difficult optimization task and for its solution we propose the use of an augmented evolutionary algorithm.

To be more specific, the mobile robot in question is meant to reach positions set by the central control unit, and many factors should be taken into account like:

- energy consumption (related to a limited battery capacity),
- task completion time,
- movement constraints (related to construction of the robot),
- difficult and dangerous areas in the environment, etc.

Different methods for GPP are known, for instance the Roadmap methods [16, 7], methods based on potential functions [13] or heuristic methods like A* and its variants [5]. They are, however, not well suited to problems with multiple criteria as considered here. This is the main motivation for our proposal to use a meta-heuristic based optimization approach, to be more specific, a genetic algorithm with some application specific operators. For some examples of the use of meta-heuristics, cf. [15, 17, 1, 18, 2, 3, 11], to name a few.

Energy consumption is in general a very important functional and economic aspect, specifically in the application considered here because the mobile robots used are powered by batteries of a limited capacity, and have to come back to origins after the completion of their tasks. Therefore, in addition to the path length, and (often its related) time of action, we will explicitly include the energy consumption in our performance function to be minimized. It is worth mentioning that the shorter path need not imply the smallest energy consumption, in general. As to some related works, some approach for an energy efficient, velocity related path planning was proposed in [10], and a two step procedure was given in [8, 9].

In this paper we propose a novel approach to the GPP that employs a modified evolutionary algorithm. Though our discussion is more general, we will use some elements of an implemented MOTHON (Mobile platform for THreat mONitoring) system [6, 4] meant for mobile robots used for intrusion detection in the WSNs (wireless sensor networks). Basically, the mobile robot(s) equipped with specialized hardware and software for intrusion detection in the WSNs.

Our new approach explicitly includes:

- evaluation criteria involving the path length, energy consumption, travel time and movement cost for dangerous zones, constraints on the robot's turning angle, total energy consumption and total travel time, etc.

- specialized evolutionary operators with input parameters which can be changed during algorithm operation,

Notice that we do not deal with the velocity planning which is here meant to be part of the LMP (local motion planning). Technically, we assume our environment to be a 200×200 grid map, with different ground surfaces, obstacles and unwanted zones in the form of areas dangerous for robot(s) operation. The tests are performed using the popular V-REP environment [14].

2 Problem formulation

The problem considered here is to determine a feasible and optimal (of course, this is meant in a somewhat informal way due to the later use of a meta-heuristic) path from the starting point $[x_0, y_0]$ to the end point $[x_N, y_N]$ in the environment given by a grid map presented as two matrices:

- matrix M^T which stores information about the type of the surface at the considered point $[x, y]$ (e.g., flat asphalt for which $M^T[x, y] = 0$, rough macadam for which $M^T[x, y] = 1$, etc.; each with a specific rolling friction factor μ , which has strong influence on the optimization task,
- matrix M^O which stores information about objects located in the environment, like:
 - free cell, $M^O[x, y] = 0$,
 - obstacles, $M^O[x, y] = 1$,
 - dangerous areas, $M^O[x, y] = 2$, etc.

A particular solution (individual, in our population based approach) involves sets of two coordinates $S = [[x_0, y_0], [x_1, y_1], \dots, [x_N, y_N]]$, where N denotes the number of points in the trajectory. The subsequent coordinates from the set S are connected with each other into linear segments $p_k p_{k+1}$ given by the beginning and end point, e.g. $p_0 p_1 = [[x_0, y_0], [x_{01}, y_{01}], [x_{02}, y_{02}], [x_1, y_1]]$, and the whole trajectory is describes as a set of path segments, i.e. $P = [p_0 p_1, p_1 p_2, \dots, p_{N-1} p_N]$. Angles between the consecutive path segments, $A = [\alpha_{0,1}, \alpha_{1,2}, \dots, \alpha_{N-2, N-1}]$, are given as a set A .

The feasible solutions (points along a trajectory) are specified as:

- trajectory points cannot be located on obstacles, i.e.

$$\forall [x, y] \in P : M^O[x, y] \neq 1 \quad (1)$$

- angles between path segments must be higher than the minimal angle α_{min} ,

$$\forall \alpha \in A : \alpha > \alpha_{min} \quad (2)$$

- a maximum energy consumption level cannot be exceeded, i.e.

$$E < E_{max} \quad (3)$$

- a maximal task completion time cannot be exceeded, i.e.

$$t < t_{max} \quad (4)$$

The objective (performance) function, which due to a lack of space will be presented in a brief way, is

$$f(S) = w_e \cdot f_{energy} + w_t \cdot f_{time} + w_c \cdot f_{cost} + w_o \cdot P_{obst} + w_{eL} \cdot P_{en} + w_{tL} \cdot P_{time} + w_a \cdot P_{angle} \quad (5)$$

and is to be minimized.

The form of the objective function follows the very essence of the problem considered, and – for clarity and convenience – its terms (which can have some weights assigned too) can be divided into:

- the performance related terms, i.e. those related to energy, time and distance, are assumed in the following form:
 - the energy related term, which is concerned with the energy consumption, is (cf. [8, 9])

$$f_{energy} = \sum_{k=0}^{N-1} \left(\mu_k \cdot m \cdot g \cdot s_{k,k+1} + \frac{K_s}{V_k} \cdot s_{k,k+1} \right) \quad (6)$$

where: μ_k denotes the rolling friction of the terrain surface, m denotes the robot mass, g is the gravitational acceleration, V_k denotes the robot speed at the considered trajectory fragment $p_k p_{k+1}$, K_s is a static coefficient which stands for energy loss during the transformation of the electrical into the mechanical energy cf. [?] [31]), and

$$s_{k,k+1} = \sqrt{(x_k - x_{k+1})^2 + (y_k - y_{k+1})^2} \quad (7)$$

- the time related term, which is based on the robot's velocity and path length, is.

$$f_{time} = \frac{s_k}{V_k} \cdot h_k, h_k \in \langle 1, 2 \rangle \quad (8)$$

where h_k denotes a difficulty coefficient for the path segment as even with the constant velocity, some path segments may have different travel time (e.g. sand versus asphalt); h_k is 1 if the above aspect is not accounted.

- the movement cost term is the sum of movement costs for all segments on the robot's path, i.e.

$$f_{cost} = M^C[x_N, y_N] + \sum_{k=0}^{N-1} \left(M^C[x_k, y_k] + \sum_{l=1}^{N_{l,k}} M^C[x_{k,l}, y_{k,l}] \right) \quad (9)$$

where N denotes the number of points in individual S , $N_{l,k}$ stands for the number of intermediate points in $p_k p_{k+1}$ and M^C denotes the matrix with movement cost for the particular segments.

- the penalty related terms which concern various types of violations of constraints along the robot's trajectory, namely:
 - the penalty for the path infeasibility, i.e. meant to safeguard against the crossing through obstacles (detect segments at which this occurs), given as

$$p_{obst} = 1000 \cdot \sum_{k=0}^{N-1} O_{k,k+1} \quad (10)$$

$$O_{k,k+1} = \begin{cases} 1 & \text{if } \exists [x,y] \in p_k p_{k+1} : M^O[x,y] = 1 \\ 0 & \text{otherwise} \end{cases}$$

where $O_{k,k+1}$ indicates whether path segment $p_k p_{k+1}$ violates the obstacle feasibility rule (1) or not.

- the penalty for the total amount of energy consumed along a particular trajectory being higher than the battery capacity, i.e.

$$p_{e_n} = 1000 \cdot E_{limit}$$

$$E_{limit} = \begin{cases} 1 & \text{if } f_{energy} > E_{max} \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

- the penalty for the total time of the robot's movement being higher than a specified level, given as

$$p_{time} = 1000 \cdot T_{limit}$$

$$T_{limit} = \begin{cases} 1 & \text{if } f_{time} > T_{max} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

- the penalty for the angle between the consecutive segments of the trajectory being higher than the maximum turning angle of the robot, given as

$$p_{angle} = 1000 \sum_{k=0}^{N-2} TA_{k,k+1} \quad (13)$$

$$TA_{k,k+1} = \begin{cases} 1 & \text{if } \alpha_{k,k+1} < \alpha_{max} \\ 0 & \text{otherwise} \end{cases}$$

where $T_{k,k+1}$ indicates whether the turning angles are feasible or not.

The (performance) objective function (5) reflects both the very purpose of global path planning, and constraints, and will be employed in our evolutionary approach to the global path planning playing the role of the fitness function.

3 A novel evolutionary computation approach to the global path planning

The pseudocode of our new evolutionary approach to the global path planning (GPP) for the problem considered can be depicted as:

Algorithm 1 Evolutionary Global Path Planner

```

1:  $it \leftarrow 0$ 
2:  $population$  = initialize population
3:  $probabilities$  = initialize probabilities for evolutionary operators
4: for  $number\ of\ generations$  do
5:    $it \leftarrow it + 1$ 
6:   create  $new\ population$  list
7:   for  $number\ of\ elitist\ individuals$  do
8:     select currently best and not yet chosen  $individual$  from  $population$  list
9:     append  $new\ population$  list with selected  $individual$ 
10:  for  $tournament\ population\ size$  do
11:    while  $offspring$  not in  $new\ population$  do
12:      draw evolutionary operator
13:      select  $parent(s)$  from  $population$  on the basis of tournament selection
14:      apply evolutionary operator to selected  $parent(s)$ 
15:      if  $offspring$  not in  $new\ population$  then
16:        append  $new\ population$  list with  $offspring$ 
17:  for  $random\ population\ size$  do
18:    while  $offspring$  not in  $new\ population$  do
19:      draw evolutionary operator
20:      select  $parent(s)$  from  $population$  on the basis of random selection
21:      apply evolutionary operator to selected  $parent(s)$ 
22:      if  $offspring$  not in  $new\ population$  then
23:        append  $new\ population$  list with  $offspring$ 
24:  evaluate  $new\ population$ 
25:   $population = new\ population$ 
26:  if  $termination\ condition$  is True then
27:    stop
28:  modify evolutionary operators  $probabilities$  list

```

Due to a lack of space, we will just briefly explain the main elements of our algorithm by emphasizing our new proposals.

For the generation of a new population an elitist approach is performed, with a user selected elitist number, usually $\leq 5\%$, then the population is completed to the proper size which is done in two steps, using a tournament and random selection, based on a current list of probabilities and operators selected.

A single individual consists of at least two points (the start and end points) and is not of a fixed length. Between each set of coordinates x,y a feasibility flag q is set such that

if a path segment p_0p_1 is feasible, the flag q_1 is 0 otherwise is equal to some case dependent integer number, and $q_N = 0$ as point N is the last point of the individual. The use of the evolutionary operators can clearly change the number of points, except for the first and last, and the intermediate points can be determined using various algorithms.

The first population is generated due to the following algorithm which generates a number of populations, and then one (with the bigger variance) is chosen:

Algorithm 2 Population initialization

```

1: for number of populations to generate do
2:   for number of individuals in one population do
3:     insert starting point into individual
4:     draw number of points to insert in specified individual range
5:     for drawn number of points to insert do
6:       draw random point in range  $\langle \min_{xy}, \max_{xy} \rangle$ 
7:       insert point into individual
8:     insert ending point into individual
9:     remove loops in individual
10:    add individual to the population
11:  append list of populations
12: from the populations list, select population with the biggest variance

```

Then, some additional operations on the population exemplified by loop removal are performed but they will not be discussed for a lack of space.

In our approach we use evolutionary operations which are general but their choice is specific for the application.

Namely, the operations performed can be summarized as follows:

- the *one point crossover* operator, a standard operation that in our case exchanges the first and second parts of the two randomly selected trajectories,
- the *mutation operator* in which a random point(s) is/are selected and change/s (mutation/s) of the specific part of the trajectory is/are made,
- the *insert operator* which inserts a random point into two randomly selected points in trajectory,
- the *delete operator* which randomly deletes one or more points from the trajectory,
- the *swap operator* which randomly selects two neighbouring points in the trajectory and changes their order,
- the *improve operator* which selects randomly a point in the trajectory and searches for the best improvement of the values of the specific terms in the fitness function,
- the *repair obstacle operator* which is a more complex operator which, briefly speaking, tries to “repair” the trajectory by avoiding its crossing of obstacles that

can be just plain obstacles but also areas in which the movement costs, energy consumption, difficult surface, etc. occur,

- the *insert-shorten operator* is also a complex operator which tries to make the existing trajectory shorter by basically trying to find a feasible path between two point that result in a higher value of the fitness function,

Of course, all these operators are applied to a specific trajectory or specific trajectories with different probabilities which are selected by an application specific algorithm, starting with an analysis of the diversity (variance) in the populations.

4 Simulation and some results

We will now briefly present some results attained for the analysis of a real problem.

Our approach to the GPP is implemented in Python 2.7 with loops for offspring generation parallelized using multiprocessing. The simulation is performed using the V-REP simulator [14] which can be controlled by Python scripts using specialized libraries.

We used the Pioneer 3-DX mobile robot that is widely used [12]; its size is ca. 0.4×0.3 m. The environment was 100×100 m decomposed into the 200×200 grid (0.5×0.5 m per grid cell). The main terrains friction coefficients and robot parameters were assumed as in [8, 9]. Due to a lack of space, we cannot show numerous parameters, probabilities, etc. of operations, population generations schemes, and other elements of the algorithm which will be given in a next publication.

For illustration, we will show results attained by our new evolutionary approach to the GPP, and compare it to the results obtained by using an ordinary genetic algorithm. The robot starts from $x = 75$, $y = 193$ and its end position is $x = 185$, $y = 193$. We assume two criteria problem, i.e. with the energy consumption and travel time as the (conflicting) criteria, with all other terms of the fitness function representing, via penalties, constraints. The two criteria are combined in the fitness function via the weighted average, with properly chosen weights.

Just for illustration, we will present a comparison of the worst trajectory obtained using our new evolutionary approach to the GPP shown in Fig. 2) and the best trajectory obtained by using the traditional genetic algorithm shown in Fig. 1). It can easily be seen that the former has a substantially lower fitness, energy usage and time of travel than the latter. It can therefore be seen that for quite complex task considered, which involves multiple criteria (objectives), both the energy consumption, travel time, and many constraints, our new evolutionary approach to the global path planning gives much better results than those obtained by the approach which involves the traditional genetic algorithm. For the comparison, best trajectory determined by the new evolutionary algorithm is shown in Fig. 3. In these figures, green colour stands for robot's trajectory, red colour denotes obstacles, gray colour denotes areas with higher friction factor and finally blue colour stands for areas considered as dangerous.

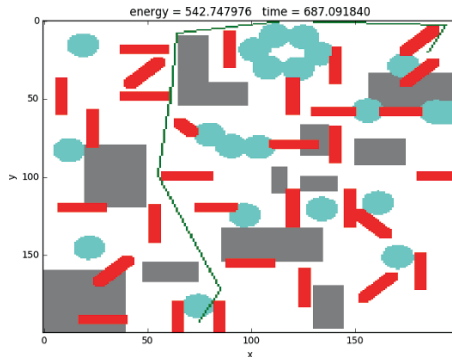


Fig. 1: Best result obtained by using a genetic algorithm

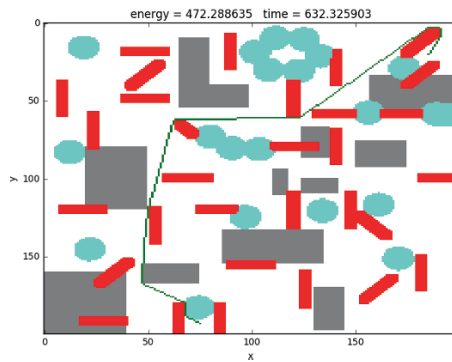


Fig. 2: Worst result obtained by our new evolutionary algorithm

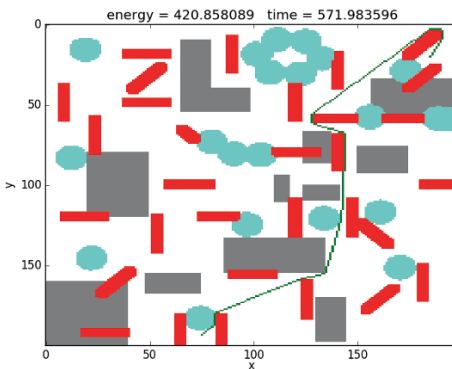


Fig. 3: Worst result obtained by our new evolutionary algorithm

5 Conclusions

Summary. We have proposed a new specialized evolutionary algorithm for the global path planning for mobile robots. We considered the case of robots for intrusion detection in the WSNs (wireless sensor networks). We have assumed the existence of multiple criteria, notably – which is rarely accounted for – the energy consumption, travel time and movement cost on dangerous areas, as well as constraints like the location obstacles, a limited minimal robot’s turning angle, a maximal energy (related to the capacity of batteries) and maximal time of travel. The new algorithm have involved many new problem specific evolutionary operations. The simulation results using the V-REP platform have taken into account obstacles, different surfaces and dangerous areas. Simulation environment was integrated with MOTHON system. The results have been very encouraging, much better than those obtained using the traditional meta-heuristics, notably the widely used genetic algorithm. Moreover, a novel explicit involvement of energy consumption as a key factor, has provided much new insight from both a theoretical and even more so practical points of view.

References

1. I. Ashiru, C. Czarnecki, and T. Routen Ashiru et al. Characteristics of a genetic based approach to path planning for mobile robots. *J. Network and Computer Applications*, 19:149–169, 1996.
2. C. Hocaoglu and C. Sanderson Hocaoglu et al. Planning multiple paths with evolutionary speciation. *IEEE Transactions on Evolutionary Computation*, 5(3):169–191, 2001.
3. Y. Hu and S. Yang Hu and Yang. A knowledge based genetic algorithm for path planning of a mobile robot. *Proc. of IEEE Intl. Conf. On Robotics and Automation, New Orleans, LA, April, 2004*.
4. K.Lasota, P. Bazydo, and A.Kozakiewicz Kozakiewicz et al. Mobile platform for threat monitoring in wireless sensor networks. *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 106–110, 2016.
5. S. Koenig, M. Likhachev, and D. Furcy Koenig et al. Lifelong planning a*. *Artificial Intelligence*, 155:93–146, 2004.
6. A. Kozakiewicz, K. Lasota, M. Marks, and E. Niewiadomska-Szynkiewicz Kozakiewicz et al. Intrusion detection in heterogeneous networks of resource-limited things. *Journal of Telecommunications and Information Technology*, 4:10–14, 2015.
7. S. LaValle. Rapidly-exploring random trees: A new tool for path planning. *TR 98-11, Computer Science Dept., Iowa State University*,, 1998.
8. S. Liu and D. Sun Liu and Sun. Optimal motion planning of a mobile robot with minimum energy consumption. *IEEE/ASME Intl. Conf. On Advanced Intelligent Mechatronics, AIM, Hungary*, pages 43–48, 2011.
9. S. Liu and D. Sun Liu and Sun. Minimizing energy consumption of wheeled mobile robots via optimal motion planning. *IEEE/ASME Transactions on Mechatronics*, 19(2):401–411, 2014.
10. Y. Mei, Y.H. Lu, Y.C. Hu, and C.S.G. Lee Mei et al. Energy-efficient motion planning for mobile robots. *Proc. of IEEE Intl. Conf. On Robotics and Automation*,, 2004(5):4344–4349, 2004.
11. J. Kacprzyk Bigaj et al. P. Bigaj. A memetic algorithm based procedure for a global path planning of a movement constrained mobile robot. *IEEE Congress on Evolutionary Computation, CEC*, pages 135–141, 2013.

12. P. Pioneer. www.mobilerobots.com/researchrobots/pioneerp3dx.aspx. 2016.
13. E. Rimon and D. E. Koditschek Rimon et al. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8:501–518, 1992.
14. E. Rohmer, S. P. N. Singh, and M. Freese Rohmer et al. V-rep: a versatile and scalable robot simulation framework. *EEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2013.
15. T. Shibata and T. Fukuda Shibata et al. Intelligent motion planning by genetic algorithm with fuzzy critic. *Proc. 8th IEEE Int. Symp. Intelligent Control, Chicago, Aug. 25-27*, pages 565–570, 1993.
16. N. H. Sleumer, L. E. Kavraki, P. Svestka, L. E. K. P. Vestka, J. Latombe, and M. H. Overmars Sleumer et al. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12:566–580, 1996.
17. K. Sugihara and Sugihara et al. J. Smith. Genetic algorithms for adaptive motion planning of an autonomous mobile robot. *Proc. of IEEE Intl. Symposium on Computational Intelligence in Robotics and Automation*, pages 138–143, 1997.
18. J. Xiao and Z. Michalewicz K. Trojanowski Xiao et al. Adaptive evolutionary planner/navigator for mobile robots. *IEEE Transactions on Evolutionary Computation*, 1(1):18–28, 1997.

Trajectory planning for Service Ship during emergency STS transfer operation

Anna Witkowska, Roman Śmierzchalski and Przemysław Wilczyński

¹ Gdańsk University of Technology, Elect & Control Engn Dept, Gdańsk, Poland,
anna.witkowska@pg.gda.pl,

² Gdynia Maritime University, Faculty of Navigation, Gdynia, Poland

Abstract. In this paper trajectory for approaching during emergency STS transfer operation with oil spill is considered as a sequence of navigation manoeuvres in specific navigational environment. The designed way points - ship positions and speed are determined as reference values to support navigator in decision making during steering and to mitigate the risk of collision which mostly results from exceeding the speed limit of approaching. To prevent this, the values of position and speed in each way points are determined with respect to specific manoeuvring procedure during STS approaching and by taking into account constraints resulting from ship's manoeuvre of stopping and speed deceleration performance included in manoeuvring booklet. Additional constraints result from STS transfer operation guide and navigation practise. The task of trajectory planning is defined as optimization process to minimize trajectory length, course alteration and maximize safety.

Keywords: Ship to Ship transfer operation, stopping and speed control characteristics, safe trajectory planning, evolutionary algorithm

1 Introduction

To control ship motion at sea trajectory planning in a dynamic environment is used. The issue consists in determining ship trajectory between start and final points, which enables to avoid collision with static and dynamic obstacles and taking into account ship manoeuvring performance. Several solutions have been introduced to solve this problem. One of them contains Game Theory, where a problem of non-collision control strategies in the steering at sea is analyzed [7]. Other methods like Genetic or Evolutionary Algorithm, Particle Swarm Algorithm consider collision avoidance as a multi-criteria, nonlinear optimization problem with navigational time, safety and economy criteria [4]. In the above studies, the focuses were trajectory searching represented by the set of way points consists of desired positions and speed or heading. The transformation of the way points to a feasible trajectory was modelled as straight-lines and inscribed circles [5], polynomials or splines [3]. The above methods assume constant ship's speed on a straight line and decreasing in speed during course alteration, alternatively time of manoeuvre as a function of course, speed and load condition. Dynamic

properties of ship have been implemented from turning circle test. However, in low speed manoeuvring operations at sea like Lightering underway, emergency Ship to Ship transfer (STS), docking the stopping and deceleration characteristics are important and should also be taken into consideration. So the above results cannot be used directly in reality.

STS transfer operation generally involve transshipment between two ships, the large called SBL (Ship to be Lightered) and small one called SS (Service Ship) positioned alongside each other, either while stationary or underway in order to commence cargo transfer [8, 9]. Before mooring and cargo transfer start, the Service Ship has to approach the Ship to be Lightered, which moves on a constant heading with slow speed or drifts about zero. For this purpose basically a collision avoidance manoeuvre has to be carried out in order to obtain the required safety distance between two ships and to take side by side position. The most common incident that occurs during STS operations is a collision between the two ships while manoeuvring alongside each other or sailing [14]. Collision between two ships typically occurs because of reasons which include: incorrect approach angle between the manoeuvring ships; approaching at excessive speed; failure of one or both ships to appreciate meteorological conditions. To mitigate the risk of incidents, guidelines are needed for the navigator of Service Ship, which include information about reference trajectory for approaching in meaning of reference way points p_i : position (x_i, y_i) /or heading ψ_i and speed v_i to take a proper steering decision by ship operator at each stage of ship manoeuvring.

Considered in the paper emergency STS transfer operation means that product tanker (SBL) after collision with general cargo ship lost its ability to manoeuvre and start drifting due to wind. Immediate actions were carried out to reduce oil spill overboard. Small tanker (SS) was designated to emergency STS operation. During this incident can appear additional important aspects like ship and cargo condition (transshipment from undamaged side), wind direction (transshipment from leeward), speed reduction, time limits (optionally to ensure fast transshipment) as well as water area constraints (close to port area), avoidance moving oil spill or other rescue units [16].

Our objective is to determine a desired trajectory for approaching during emergency STS transfer operation taking into account stopping and control manoeuvring characteristics of the vessels involved in manoeuvring booklet. Estimated on available information trajectory allows to take proper manoeuvring decision by ship operator using rudders and propellers and to mitigate oil spill to the environment. The resulting optimal trajectory will have to make an assumption of economic (minimum distance for approaching and course alteration) and safety of manoeuvring (non-collision).

2 Stopping and speed deceleration characteristics

To control the movement of Service Ship during STS Approach Manoeuvre a few general control modes are possible, in order to achieve the final distance from

SBL, parallel course and equal speed. They are recommended based on good navigation practice and STS transfer operation guides [8, 9, 13, 2]. The control modes consist of:

- I. Trajectory Tracking (moderate or high-speed manoeuvring)
- II. Stopping Manoeuvre (stop ship)
- III. Berthing (side manoeuvre)

After Approach Manoeuvre by using I, II control modes ships should manoeuvre alongside at the required safety distance DCPA (Distance at Closest Point of Approach). That means both SS and SBL keep their constant heading $\psi_{SS} \approx \psi_{SBL}$ at minimum controllable constant speed $v_{SS} \approx v_{SBL}$ or drifting about zero (during emergency). In this condition the Berthing operation by using the tunnel thruster and mooring procedure by using lines can start.

Comprehensive details of the ship stopping and speed deceleration characteristics are included in the manoeuvring booklet. This booklet is required to be on board and it has to be available for navigators. Most of the manoeuvring information in the booklet can be estimated but some should be obtained from trials. They contain (among other relevant data) characteristics of main engine, stopping test results (emergency and normal) and speed deceleration performance.

The characteristics of main engine contain possible engine order (Full Sea Ahead, Full Ahead, Half Ahead, Slow Ahead, Dead Slow Ahead, Dead Slow Astern, Slow Astern, Half Astern, Full Astern), propeller revolution, speed, power, pitch ratio. Stopping ability [15] of SS is measured by the track, head, side reach, time required to speed reduction and final course. It covers the following modes of emergency stopping manoeuvres: from Full Sea Ahead to Full Astern; from Full Ahead to Full Astern; from Half Ahead to Full Astern; from Slow Ahead to Full Astern; from Full Sea Ahead to stop engine and following modes of normal stopping manoeuvres: from Full Ahead to stop engine; from Half Ahead to stop engine; from Slow Ahead to stop engine. Deceleration performance concern track reach, head reach and time required. It covers the following modes: from Full Sea Ahead to Full Ahead; from Full Ahead to Half Ahead; from Half Ahead to Slow Ahead; from Slow Ahead to Dead Slow Ahead. When vessel travels along a straight line with the original course (autopilot is on) the track reach and time reach values are taken as the longest travelling distance and the maximum time to decelerate ship speed.

3 Trajectory Planning for approaching

The Service Ship trajectory for approaching is defined as a set of turning points $P = \{p_0, p_1, \dots, p_k\}$ on ship route from current position (initial point) p_0 to the destination (final point) p_k and a set $S = \{s_1, s_2, \dots, s_k\}$ of trajectory segments between them with a segment lengths $D = \{d_1, d_2, \dots, d_k\}$. The way points $p_i(x_i, y_i, v_i, t_i)$, $i \in \{0, 1, \dots, k\}$ of desired trajectory are interpreted as geometrical position x_i, y_i with respect to a maximum ship speed

v_i , $i \in \{0, 1, \dots, k\}$ and time t_i of approaching i -th position. Trajectory segments s_i compose of the path position sequences between way points on straight line as a function in time $s_i(t)$. It can satisfy speed deceleration performance. Deceleration performance means that for a given starting reference speed v_{i-1} at p_{i-1} it is possible to approach by ship the ending one $v_i \leq v_{i-1}$ at p_i with segment length d_i . When the vessel travels on a straight line along the original course this segment length value can't be less than track reach needed for speed deceleration or stop ship.

3.1 Modelling of the way points

Additional modelling require the way points in close proximity of SBL p_k , p_{k-1} and p_{k-2} , $k \geq 2$. They depend directly on STS transfer operation guide and navigation practise during STS Approach Manoeuvre [1, 10]. The example of modelling way points is shown in a Fig.1, where starboard side manoeuvre and NE'yly wind direction is considered. In the open waters the last phase of standard Approach Manoeuvre begins at distance R about 0.5 Nm from the destination point and finish at DCPA approximately 50-100 meters off. The initial way point p_0 consist of a current position (x_0, y_0) , speed v_0 of Service Ship at t_0 , when it start Approach Manoeuvre. The destination point $p_k(x_k, y_k, v_k, t_k)$ has parallel position ($l_{SS} \parallel l_{SBL}$) in a safety distance (DCPA) from SBL and the same speed $v_k \approx v$, to allow starting manoeuvring alongside. When emergency STS trajectory is being planned the SBL maintain its current position (x, y) constant and speed drifting about zero $v \approx 0$. In this case the initial p_0 and destination p_k points have approximately constant positions chosen by the operator or calculated by the simple geometric relationship:

$$\begin{aligned} p_{k-1}|_{(x_{k-1}, y_{k-1})} &\in l_{SS}, \quad l_{SS} \parallel l_{SBL}, \\ DCPA &= \left\| p|_{(x,y)} p_{k-1}|_{(x_{k-1}, y_{k-1})} \right\|_2, \\ v_k &\approx v, \end{aligned} \quad (1)$$

where

$p|_{(x,y)} = (x, y)$, $p_{k-1}|_{(x_{k-1}, y_{k-1})} = (x_{k-1}, y_{k-1})$,
 l_{SS} —straight line covers SS diametrical line,
 l_{SBL} —straight line covers SBL diametrical line.

The previous way point p_{k-1} has position determined on straight line l_{SS} parallel to l_{SBL} :

$$p_{k-1}|_{(x_{k-1}, y_{k-1})} \in l_{SS}, \quad l_{SS} \parallel l_{SBL}. \quad (2)$$

The reference speed v_{k-1} at p_{k-1} is modelled as minimum controllable speed v_{DSA} (Dead Slow Ahead) for safety manoeuvring in close proximity $v_{k-1} = v_{DSA}$, with satisfying deceleration performance on trajectory segment s_k :

$$d_k = \left\| p_{k-1}|_{(x_{k-1}, y_{k-1})} p_k|_{(x_k, y_k)} \right\|_2 \geq track\ reach_k, \quad (3)$$

where $track\ reach_k$ is the travelling distance need to decelerate ship's speed from v_{DSA} to stop.

The way point p_{k-2} is determined on the arc L_{AB} between the end points A and B satisfying $A \in l_{SS}$. The arc is a part of a circle $O(p_k, |AO|, \alpha)$ with a radius $|AO| = R$ of cells and central angle $\alpha \in < 0, 30^0 >$. It is also assumed that reference speed $v_{k-2} = v_{DSA}$ was predetermined as minimum controllable

$$p_{k-2}|_{(x_{k-2}, y_{k-2})} \in L_{AB}, v_{k-2} = v_{DSA}, \quad (4)$$

where

$$L_{AB} \in O(p_k, |AO|, \alpha), \quad \alpha \in < 0, 30^0 >, \quad |AO| = R.$$

4 Formal problem definition

The process of Trajectory Planning for approaching is considered as an example of classical avoiding collisions at sea. It is reduced to an optimization task with static and dynamic constraints in the navigational environment with safety and economic criteria [12, 11].

4.1 Configuration Space/ Search Space

The search space of position variables (the set of all possible solutions) is defined in two-dimensional Euclidean space of navigational environment:

$$X_{env} = \{(x, y) \in E^2: a \leq x \leq b, c \leq y \leq d\}. \quad (5)$$

The space consists of: safe areas $X_{safe}(t)$, static obstacles domains X_{stat} , dynamic obstacles domains $X_{dyn}(t)$.

$$X_{env} = X_{safe}(t) \cup X_{stat} \cup X_{dyn}(t). \quad (6)$$

The choice of maximum speed values v_i , $i \in \{0, 1, \dots, k-1\}$ at each way points of desired trajectory depend on set

$$V = \{v_{FSS}, v_{FA}, v_{HA}, v_{SA}, v_{DSA}\} \quad (7)$$

and $v_i \approx 0$ for $i=k$.

The following engine orders are considered: Full Sea Ahead (v_{FSA}), Full Ahead (v_{FA}), Half Ahead (v_{HA}), Slow Ahead (v_{SA}), Dead Slow Ahead (v_{DSA}). The static obstacles X_{stat} such as land, islands, shallow water are modelled by domains, represented geometrically by convex polygons. The dynamic obstacles X_{dyn} such as other ships are modelled by domains, evaluating in time and represented by hexagon with known current and predicted position, constant course and speed (containing Colregs rules).

Among them can models oil spill domain X_{oil} , SBL domain X_{sbl} and unavailable sector X_{unav} apart which can be treated as static or dynamic, see Fig.1. The shape and size of X_{sbl} depend on ship speed, wind direction, DCPA and side of approach. Oil spill domain X_{oil} can also evaluate in time and depend on emergency and weather conditions. It is also possible to model prediction of the

oil spill area [6]. In the paper an oil spill and SBL domain are represented by static hexagon and triangle domain respectively because of SBL drifting and oil barrier. Unavailable domain X_{unav} contains forbidden sectors which results from ship manoeuvring and operation constraints by using rudders at low speed.

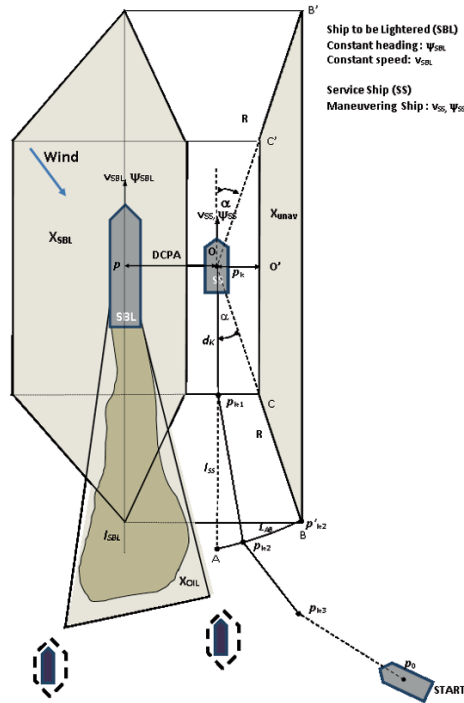


Fig. 1. Modeling of the way points in navigational environment

4.2 Constraints

Desired trajectory satisfies safety and deceleration condition. The reference trajectory during STS assume to be safe if each of way point p_i , $i = \{0, \dots, k\}$ and segment s_i , $i = \{1, \dots, k\}$, between way points does not cross in the area of the environment with the static and dynamic obstacles:

$$S \subset X_{safe}(t) \text{ and } P \subset X_{safe}(t). \tag{8}$$

Deceleration condition means satisfying ship's stopping and speed deceleration performance.

$$v_i \leq v_{i-1}, \quad i = \{1, \dots, k\}, \quad v_k \approx 0, \tag{9}$$

$$d_i \geq \text{track reach}_i, \quad i = \{1, \dots, k\}. \quad (10)$$

Additional constraints can result from weather condition, STS transfer operation guide and navigation practise (see Chap. 3.1).

4.3 Optimization criteria

The problem of determining STS trajectory for approaching is defined as multi-criteria optimization task due to the presence of a function to opposing criteria. Therefore, it is proposed to use an approach based on replacing the multi-objective function by the function single-criterion $f(P, S)$ with weighting factors w_1, w_2, w_3 .

$$f(P, S) = f_{\text{econ}}(P, S) + f_{\text{safe}}(P, S), \quad (11)$$

where

$$\begin{aligned} f_{\text{econ}}(P, S) &= w_1 f_{\text{dist}}(P, S) + w_2 f_{\text{smooth}}(P, S), \\ f_{\text{safe}}(P, S) &= w_3 f_{\text{clear}}(P, S). \end{aligned}$$

The function f consists of costs related to the economics of shipping f_{econ} and safety costs f_{safe} . The economic cost f_{econ} are related to the length of the trajectory f_{dist} as well as the degree of smoothness f_{smooth} associated with course alteration.

Safety costs f_{safe} are associated with avoiding navigation constraints of both static and dynamic f_{clear} . Component f_{clear} defines a safe distance of passing navigation constraints.

5 Simulation tests results

The evolutionary path planning algorithm is proposed based on a natural selection mechanism to determine STS trajectory for approaching as an optimization task. Its most important advantages are build-on adaptation mechanism for a dynamic environment and reaching a multi-criteria task solution in a near-real time. In actual implementation introduced several modification of earliest version of evolutionary path planning method [11] to adapt an algorithm to STS. Each node consists of x, y and v coordinates. Velocity v is generated randomly from the set V (7) and satisfies (9). The feasibility of trajectory means that the trajectory is safe (8) and at the same time satisfies ship's stopping and speed deceleration performance (10). The algorithm takes into account the direction of wind (SS approaches from leeward (Fig.1)). Speed of own ship on straight-line segments can be fixed or variable in a non-linear manner according to speed deceleration performance (Fig.2). So the collision time is also calculated in a non-linear manner on the basis of data from speed deceleration characteristics. The repair mechanism of speed is introduces to satisfies (9) after genetic operators are used. The way points components calculated from Evolutionary Algorithm (EA) are determined as references position (or heading) and speed values for navigator to support decision making at each stage of ship manoeuvring.

In the simulation test, the approach trajectory was determine for the SS type Chemical Tanker 6000 DWT, of length overall 103,6 m powered by one diesel engine rating 3600kW at 200 rpm. The tanker is propeller by 1 fixed pitch propeller. The ship is steered with one rudder which maximum angle 65° . This ship is equipped with one bow tunnel thruster rating 400kW. Stopping ability in deep water can be judged from emergency stop manoeuvre when autopilot is turned on. Figure 2 present deceleration ability of SS in detail when autopilot is turned on. Table 1 includes estimated value of track reach (distance travelled), head reach, side reach, speed and time to stop (time till vessel is dead in water) from Dead Slow Ahead to Stop. The example trajectory of approaching is shown in a Figure 3. It is composed of four way points p_0, p_1, p_2 and p_3 . On the resulting trajectory are determined additional points to support the navigator in decision making. The detailed data as positions, velocities, lengths and times on each trajectory segment are shown in Table 2. The resulting trajectory is safe and satisfies speed deceleration performance in meaning of satisfying velocity, time and track reach constraints. Time to reach intersection points (p_I, p_{II}, p_{III}) by own ship to avoidance collision was calculated depending on the way from time (Fig. 2).

Track reach [Nm]								
0.9					6-44	7.2		
			4-50	9.7			9-31 4.7	
0.8					5-55	7.3	8-39 4.8	
			3-59	9.8	5-06	7.4	7-24 4.9	
0.7	3-09	11.8						
			3-23	10.0	4-18	7.6	6-11 5.0	
0.6								
	2-26	11.9	2-47	10.1	3-31	7.8	5-01 5.2	
0.5								
	1-56	12.0	2-12	10.3	2-45	8.0	3-54 5.5	
0.4								
	1-26	12.2	1-37	10.6	2-01	8.3	2-50 5.8	
0.3								
	0-57	12.4	1-04	10.9	1-19	8.7	1-50 6.1	
0.2								
	0-28	12.7	0-31	11.3	0-39	9.1	0-53 6.6	
0.1								
	0-00	12.9	0-00	11.6	0-00	9.5	0-00 7.0	
0								
	min-s	knots	min-s	knots	min-s	knots	min-s	knots
	FSA to FA		FA to HA		HA to SA		SA to DSA	

Fig. 2. Deceleration performance (from trial test)

Table 1. Emergency stopping ability

To Stop from:	Track R. Nm	Head R. Nm	Side R. Nm	Time R. min-s	Final course $^{\circ}$
Dead Slow Ahead	0.075	0.075	0.00185	1-27	9

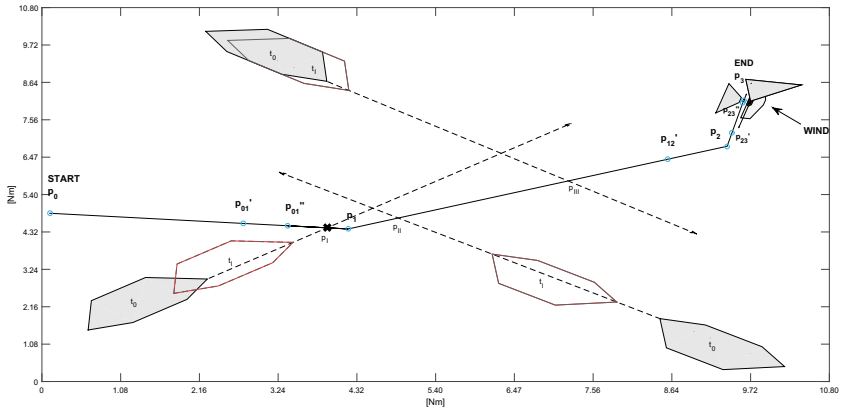


Fig. 3. Example of desired trajectory

6 Conclusion

The paper describe the problem of trajectory planning for approaching during emergency STS transfer operation with oil spill. The problem is considered as a collision avoidance task with respect to additional constraints resulting from transfer operation guide and control possibility. The guide gives us only a possible final Approach Manoeuvre without any step by step instruction about desirable position and speed of reference trajectory which mostly depend on ship speed manoeuvring properties. The information about desired speed at each way points can reduce possible factors that cause collision during STS like incorrect approach angle between the SS and SBL ships, the manoeuvring ship approaching at excessive speed, some form of human error. Taking into account ship manoeuvring characteristics during STS trajectory planning process can support Navigator in decision making to determine desired speed in each phase of manoeuvring and type of steering operation using rudder and propeller and estimated time duration to complete operation.

References

1. Husjord, D., Pedersen, E.: Operational Aspects on Decision-making in STS Lightering. Proc. of the 19th International Offshore and Polar Engineering Conference and Exhibition. Osaka, Japan, (2009).
2. Husjord, D.: Development of a Decision Support System in Ship-To-Ship Lightering. The Journal of Navigation. 69, 1154–1182 (2016)
3. Kolendo, P., Śmierczalski, R.: Experimental comparison of straight lines and polynomial interpolation modelling methods in Ship Evolutionary Trajectory Planning problem. Advances and Intelligent Computations in Diagnosis and Control. 386, 331–340 (2015)

Table 2. Trajectory way points data

	(P, S)	Additional points	x_k [Nm]	y_k [Nm]	v_k [kn]	d_k [Nm]	t_k [min-s]	Track R. [Nm]
1	(p ₀)		0.1080	4.8596	12.8996	0	0	-
2		(p _{01'} , s _{01'})	2.7591	4.5665	12.8996	2.6674	12-24	-
3		(p _{01''} , s _{01''})	3.3695	4.4988	11.6000	0.614	3-9	0.614
4	(p ₁ , s ₁)		4.2004	4.4069	9.4978	0.8360	4-50	0.8360
5		(p _{12'} , s _{12'})	8.5802	6.4189	9.4978	4.8201	30-26	-
6	(p ₂ , s ₂)		9.3971	6.7942	7.0000	0.8990	6-44	0.8990
7		(p _{23'} , s _{23'})	9.4649	7.1900	7.0000	0.4016	3-26	-
8		(p _{23''} , s _{23''})	9.6115	8.0456	4.5000	0.8680	9-31	0.8680
9	(p ₃ , s ₃)		9.6241	8.1195	0	0.0750	1-27	0.075

4. Kuczowski L., Śmierchalski R.: Termination functions for evolutionary path planning algorithm. The 19th International Conference on Methods and Models in Automation and Robotics (MMAR). Miedzyzdroje, POLAND. 636–640 (2014).
5. Lazarowska, A.: Ship's Trajectory Planning for Collision Avoidance at Sea Based on Ant Colony Optimisation. *Journal of Navigation*. 68, 291–307 (2015)
6. Lazuga, K., Gucma, L., Juszkiwicz, W.: Optimal Planning of Pollution Emergency Response with Application of Navigational Risk Management. *Annu. Navig.* 19, 67–77 (2012)
7. Lisowski, J.: Game control methods in avoidance of ships collisions. *Pol. Mar. Res.* 74, 3–10 (2012)
8. Oil Companies International Marine Forum OCIMF/ICS: Ship to Ship transfer guide, petroleum, 4th edition, Witheby Publications. Aylesbury Street. London EC1R0ET. UK. 32–36 (2005)
9. Oil Companies International Marine Forum (OCIMF): Ship to Ship Transfers-Considerations Applicable to Reverse Lightering Operations (2009)
10. Pedersen, E., Shimizu, E., Berg, T., et al.: On the development of guidance system design for ships operating in close proximity. *IEEE Proc of Position, Location and Navigation Symposium*, 1-3, 636–641 (2008)
11. Śmierchalski R. and Michalewicz Z.: Modeling of a Ship Trajectory in Collision Situations at Sea by Evolutionary Algorithm. In: *IEEE Transaction on Evolutionary Computation*. 4, 1-18 (2000)
12. Szlapczyński, R.: Evolutionary sets of safe ship trajectories with speed reduction manoeuvres within traffic separation schemes. *Polish Maritime Research*. 21, 20–27 (2014)
13. Tomera M.: Hybrid real-time way-point controller for ships. *Methods and Models in Automation and Robotics 21st International Conference on*. Miedzyzdroje, Polska (2016)
14. Ventikos, N., Stavrou, D.: Ship to Ship (STS) Transfer of Cargo. Latest Developments and Operational Risk Assessment. *Spoudai Journal of Economics and Business*. 63, 172–180 (2013)
15. Welnicki, P.: *Mechanika Ruchu Okrétu*. Politechnika Gdańska. Gdańsk (1989)
16. Wilczyński P.: STS Transfer Plan. *Technical Ship Management m/t ICARUS III* (2014)

Part VIII
Modeling and Identification

Multistage identification of Wiener-Hammerstein system

Zygmunt Hasiewicz, Paweł Wachel,
Grzegorz Mzyk, and Bartłomiej Kozdraś

Wrocław University of Science and Technology,
Department of Control Systems and Mechatronics,
Wybrzeże Wyspiańskiego 27, 50-370, Wrocław, Poland
pawel.wachel@pwr.edu.pl

Abstract. In the paper a three-stage, semirecursive scheme for the Wiener-Hammerstein system identification is proposed. The algorithm combines both parametric and nonparametric strategies and allows to recover linear and nonlinear subsystems directly from the noisy input-output data. As to the nonlinearity, the main idea is based on the recursive kernel censoring of measurements, while linear dynamics are recovered by a special kind of deconvolution. Efficiency of the obtained estimates is justified by numerical example.

Keywords: System identification, Wiener-Hammerstein system, kernel censoring, deconvolution.

1 Introduction

For the simplest block-oriented objects, such as Wiener and Hammerstein systems, many identification methods have been developed and presented in the literature, see *e.g.* [11], [10], [7], [2]. For more complicated architectures however, relatively less approaches have been proposed and analyzed. In the paper we consider identification of Wiener-Hammerstein system (LNL), being a series connection of two linear dynamic blocks and internal static nonlinearity (see Fig.1 in Section 2). The LNL system is particularly demanding in identification, mainly due to the correlation introduced by the first dynamics. In consequence, most known identification algorithms require Gaussian input signals and are based on the best linear approximation (BLA) of the static nonlinearity [9], [8], [12], [1]. First attempts to recovering of the nonlinear part without this prerequisite have been recently made *e.g.* in [6], [13], [4]. In this paper, based on the off-line approach proposed in [6], we introduce a semirecursive procedure allowing to identify nonlinear subsystem in an on-line fashion under relatively weak, nonparametric, requirements about its static characteristic and almost independently of dynamic parts. Non-Gaussianity of the input signal is also admitted. Regarding linear subsystems, we propose an identification scheme allowing estimation of impulse responses of both dynamical subsystems, based on their convolution and specially selected measurement data.

2 Problem statement

We consider discrete time Wiener-Hammerstein system shown in Fig. 1, *i.e.* the system being cascade connection of two linear dynamic blocks and internal static nonlinearity. The system is described by the following two equations

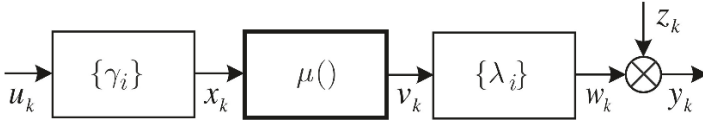


Fig. 1: Wiener-Hammerstein system

$$x_k = \sum_{i=0}^p \gamma_i u_{k-i}, \quad y_k = \sum_{i=0}^q \lambda_i \mu(x_{k-i}) + z_k, \quad (1)$$

where $\mu(x)$ is the nonlinear characteristic of the static block and $\{\gamma_0, \gamma_1, \dots, \gamma_p\}$, $\{\lambda_0, \lambda_1, \dots, \lambda_q\}$ denote impulse responses of the input and output linear subsystems, respectively. It is assumed that the sequences $\{u_k\}$, $\{y_k\}$ are accessible for the measurement, whereas both interconnecting signals $\{x_k\}$ and $\{v_k\}$ are unavailable for experimenter. The identification goal is to recover both impulse responses $\{\gamma_i\}$, $\{\lambda_i\}$ and the nonlinear characteristic $\mu(\cdot)$ from the input-output data sequence $\{(x_k, y_k)\}_{k=1}^N$ assuming that:

Assumption 1. *The orders p and q of the linear dynamic subsystems are finite and known.*

Assumption 2. *The static nonlinear characteristic $\mu(\cdot)$ is at least Lipschitz function, *i.e.*,*

$$|\mu(x) - \mu(v)| \leq l_\mu |x - v|, \quad l_\mu < \infty$$

and at least two times differentiable in arbitrarily small neighborhood B_0 of the point $x_0 = 0$ and that moreover $\mu(x_0) = 0$, $\mu'(x_0) \neq 0$.

Assumption 3. *The input process $\{u_k\}$ is a sequence of i.i.d. bounded random variables with Lipschitz probability density function $f(\cdot)$, *i.e.* $|f(u) - f(v)| \leq l_f |u - v|$, $l_f < \infty$, and $f(x_0) > 0$.*

Assumption 4. *The random noise $\{z_k\}$ is an i.i.d. zero-mean sequence with finite variance σ_z^2 and independent of $\{u_k\}$.*

As **Assumptions 1** and **2** have parametric and nonparametric character, respectively, the following identification algorithm belongs to the class of mixed parametric–nonparametric identification procedures. According to **Assumption 2** the nonlinear characteristic can be any function that fulfills merely mild global

limitations and slightly more restrictive ones in a neighborhood of point $x_0 = 0$. In the paper, to simplify the reasoning, we took $x_0 = 0$, but the method can be considered for arbitrary point x_0 . **Assumption 3** admits relatively wide class of random input signals and in this sense the method introduced further seems to be an essential extension of the up-to-now existing techniques. Finally, in **Assumption 4**, requirements concerning additive noise signal $\{z_k\}$ are formulated. They are typical for the identification algorithms designed for working in a stochastic regime.

Due to inaccessibility of the internal signals, however – without additional prior knowledge – the system characteristics can be identified only up to the unknown multiplicative constants c_0 and c_1 . In particular, the nonlinearity $\mu(x)$ can be recovered only as a scaled function $c_0\mu(c_1x)$ and both impulse responses as the scaled sequences $c_1^{-1}\{\gamma_i\}$, $c_0^{-1}\{\lambda_i\}$, respectively. Hence, without loss of generality, we shall further assume that impulse responses of linear subsystems are normalized, *i.e.* are such that $\Gamma = \sum_{i=0}^p \gamma_i = 1$ and $\Lambda = \sum_{i=0}^q \lambda_i = 1$.

3 Identification of static nonlinearity

We begin with nonparametric identification of the static subsystem (*stage 1*). The idea of the proposed method is based on the approach discussed in [3], [5], [6] and allows to recover nonlinearity $\mu(\cdot)$ independently of the other system components.

Let x be an arbitrary estimation point of $\mu(\cdot)$ and

$$\delta_k(x) \triangleq \sum_{i=0}^{p+q} |u_{k-i} - x| \tag{2}$$

denote l_1 -distance between the input sequence $\{u_{k-p-q}, u_{k-p-q+1}, \dots, u_k\}$ and the given point x . Observe, that for indexes $i = 0, 1, \dots, q$ it holds that

$$\begin{aligned} |x_{k-i} - x| &= \left| \sum_{j=0}^p \gamma_j u_{k-i-j} - \sum_{j=0}^p \gamma_j x \right| \\ &= \left| \sum_{j=0}^p \gamma_j (u_{k-i-j} - x) \right| \leq \sum_{j=0}^p |\gamma_j| \sum_{j=0}^p |u_{k-i-j} - x| \leq c_\gamma \delta_{k-i}(x). \end{aligned} \tag{3}$$

where $c_\gamma = \sum_{j=0}^p |\gamma_j|$. For any positive constant η let us consider a sequence of consecutive system inputs $\{u_{k-p-q}, u_{k-p-q+1}, \dots, u_k\}$ for which $c_\gamma \delta_k(x) \leq \eta$. Based on the inequality (3) we note that for such a set of inputs the unknown interaction items x_{k-i} , $i = 0, 1, \dots, q$ are located close to the estimation point x , provided that η is sufficiently small. Observe moreover, that from the perspective of $\{x_k\}$ and $\{y_k\}$, the Wiener-Hammerstein system is just a Hammerstein system. Hence, borrowing the idea from ([2, Ch. 4]) developed for Hammerstein systems taking account of (3), we propose for recovering $\mu(x)$ the estimate

$$\hat{\mu}_N(x) = \hat{g}_N(x) / \hat{f}_N(x), \tag{4}$$

where

$$\hat{g}_N(x) = \sum_{k=1}^N \frac{1}{\eta_k} y_k K\left(\frac{\delta_k(x)}{\eta_k}\right) \text{ and } \hat{f}_N(x) = \sum_{k=1}^N \frac{1}{\eta_k} K\left(\frac{\delta_k(x)}{\eta_k}\right) \tag{5}$$

and where $K(\cdot)$ is a kernel function assumed further to be a box selector, *i.e.*,

$$K(x) = \mathbf{1}[-1 \leq x \leq 1]. \tag{6}$$

The number sequence $\{\eta_k\}$ in (5) is in turn such that $\{\eta_k\}_{k=1}^N$ tends to zero and $N^{-2} \sum_{k=1}^N \eta_k^{-1} \rightarrow 0$ as $N \rightarrow \infty$.

In recursive form the numerator $\hat{g}_N(x)$ and denominator $\hat{f}_N(x)$ of (4) can be expressed as (*cf.* ([2]),

$$\hat{g}_N(x) = \hat{g}_{N-1}(x) - \frac{1}{N} (\hat{g}_{N-1}(x) - \frac{1}{\eta_N} y_N K(\frac{\delta_N(x)}{\eta_N})), \tag{7}$$

$$\hat{f}_N(x) = \hat{f}_{N-1}(x) - \frac{1}{N} (\hat{f}_{N-1}(x) - \frac{1}{\eta_N} K(\frac{\delta_N(x)}{\eta_N})), \tag{8}$$

which is particularly useful for on-line data processing. At the beginning one can take $\hat{g}_0(x) = 0, \hat{f}_0(x) = 0$ assuming that $0/0 = 0$.

Nonparametric estimation of nonlinearity with the use of equation (4) and formulas (7)–(8) can be performed on arbitrarily dense grid of (*e.g.* equispaced) points x implementing for instance parallel computing.

4 Estimation of convolved impulse responses

In this section we consider identification of series connection of the linear dynamic subsystems (*stage 2*), it is convolution $\{\kappa_i\}_{i=0}^{p+q}$ of the impulse responses $\{\gamma_i\}, \{\lambda_i\}$, *i.e.*

$$\kappa_i = \{\gamma_i\} * \{\lambda_i\} = \sum_{j=0}^i \lambda_j \gamma_{i-j}. \tag{9}$$

As in the previous section, in the proposed approach precise information about other system components (at present nonlinearity $\mu(\cdot)$) is not needed.

Following the definition of $\delta_k(x)$ (see eq. (2)), let now Δ_k be the l_∞ -distance between the components of the sequence $\{u_{k-p-q}, u_{k-p-q+1}, \dots, u_k\}$ and the point $x_0 = 0$ (*cf.* **Assumption 2**), *i.e.* $\Delta_k = \max_{i=0,1,\dots,p+q} |u_{k-i}|$ (we recall that, in general, the point x_0 could be arbitrary). Similarly to (3), observe that for $i = 0, 1, \dots, q$,

$$\begin{aligned} |x_{k-i}| &= \left| \sum_{j=0}^p \gamma_j u_{k-i-j} \right| \leq \sum_{j=0}^p |\gamma_j| |u_{k-i-j}| \\ &= \gamma_{max} \sum_{j=0}^p |u_{k-i-j}| \leq \gamma_{max} (p+1) \Delta_{k-i}, \end{aligned} \tag{10}$$

where $\gamma_{max} = \max_j |\gamma_j|$. Now, due to (10), we note that for arbitrarily chosen constant $h > 0$, if $\Delta_k \leq h$ then also $|x_{k-i}| \leq \gamma_{max}(p+1)h$. In consequence, for sufficiently small h , components of the segment $\{x_{k-q}, x_{k-q+1}, \dots, x_k\}$ are a collection concentrated in the neighborhood B_0 around the point $x_0 = 0$ (see **Assumption 2**) and (since the nonlinearity $\mu(\cdot)$ is at least twice differentiable in B_0) we have

$$\mu(x_k) = cx_k + \rho(x_k), \tag{11}$$

where $c = \mu'(0)$ is unknown (nonzero) constant and $\rho(x_k)$ is a reminder term of order $o(h)$. This, and equations (1) leads, in turn, to the observation that

$$y_k = c \sum_{i=0}^q \sum_{j=0}^p \lambda_i \gamma_j u_{k-j-i} + \sum_{i=0}^q \lambda_i \rho(x_{k-i}) + z_k.$$

Based on the convolution (9), the above equation can be rewritten in a more compact form as

$$y_k = \sum_{i=0}^{p+q} (c\kappa_i) u_{k-i} + r_k + z_k, \tag{12}$$

where $r_k = \sum_{i=0}^q \lambda_i \rho(x_{k-i})$ is a tail of order $o(h)$. Neglecting the tail r_k this result can be now simply exploited for estimation of $\{c\kappa_i \stackrel{\Delta}{=} \bar{\kappa}_i\}$ by correlation method. Note however, that (12) holds true only for properly selected input measurements $\{u_k\}$. Therefore, wishing to use the simple and standard correlation estimate of $\bar{\kappa}_i$, *i.e.*, to apply in fact a sample mean, the appropriate estimate must be equipped with the properly scaled kernel (box) selector $K(\cdot)$ (see eq. (6) and (7)–(8)), yielding

$$\hat{\kappa}_{i,N} = \frac{1}{N} \sum_{k=p+q+1}^{N-(p+q)} \left[\frac{1}{h_k^{p+q+3}} K\left(\frac{\Delta_k}{h_k}\right) \right] u_k y_{k+\tau}, \tag{13}$$

where $i = 0, 1, \dots, p+q$ and the weighting factor $(1/h_k^{p+q+3})K(\Delta_k/h_k)$ selects input items $\{u_k\}$ for which proper interaction inputs $\{x_k\}$ are concentrated in the neighborhood B_0 (see **Assumption 2**). We assume that number sequence $\{h_k\}_{k=1}^N$ tends to 0 as $N \rightarrow \infty$. In recursive version, the estimate (13) takes the form

$$\hat{\kappa}_{i,N} = \hat{\kappa}_{i,N-1} - \frac{1}{N} \left(\hat{\kappa}_{i,N-1} - u_{N-(p+q)} y_{N-(p+q)+\tau} \left[\frac{1}{h_{N-(p+q)}^{p+q+3}} K\left(\frac{\Delta_{N-(p+q)}}{h_{N-(p+q)}}\right) \right] \right) \tag{14}$$

similar to (7)–(8) and to start the routine we can take $\hat{\kappa}_{i,0} = 0$.

5 Identification of linear dynamics

Our goal is to decompose scaled convolved estimate $\{\hat{\kappa}_{i,N}\}$ in order to obtain separate weighting functions describing individual dynamic blocks (up to a scale;

stage 3). To present and motivate the general reasoning let us assume that, in theory, the true $\mu(\cdot)$ and $\{\bar{\kappa}_i\}$ are known. Based on assumption that the orders p and q are known too, we can construct the model of genuine system (1), parametrized by vectors $g = (g_0, g_1, \dots, g_p)^T$, $l = (l_0, l_1, \dots, l_q)^T$. Similarly to the true system description (1), the considered model is described by the equations

$$\tilde{y}_k(l, g) = \sum_{i=0}^q l_i \mu(\tilde{x}_{k-i}), \quad \tilde{x}_k = \sum_{i=0}^p g_i u_{k-i}. \tag{15}$$

We are now about to use $\tilde{y}_k(l, g)$ for determining $\lambda = (\lambda_0, \lambda_1, \dots, \lambda_q)^T$ and $\gamma = (\gamma_0, \gamma_1, \dots, \gamma_p)^T$. To this end, we can take, for instance, the mean squared error function $Q(l, g)$, parametrized by the vectors l and g , *i.e.*

$$Q(l, g) = E[y_k - \tilde{y}_k(l, g)]^2. \tag{16}$$

Noticing that for true impulse responses λ and γ it holds that $Q(\lambda, \gamma) = \sigma_z^2$ and that for any l, g , we have $Q(l, g) \geq Q(\lambda, \gamma)$, we get

$$\{\lambda, \gamma\} = \underset{l, g}{\operatorname{arg\,min}} Q(l, g) \tag{17}$$

provided that both subsystems are identifiable. Formally, equation (17) can be a basis for identification of both linear subsystems. Proper optimization problem can however be difficult and time consuming in the case of large dimensionality of λ and γ .

Alternatively, in order to avoid direct optimization as in (17) – and to enable parallel computation – we exploit that locally (due to (11)) the Wiener-Hammerstein system can be treated as serial connection of two linear subsystems with impulse responses λ and γ , and noise-free output w_k (see Fig. 1), and described as a whole by the equation $w_k = u_k \sum_{i=0}^{p+q} \bar{\kappa}_i d^i = u_k W(d)$, where d is time delay operator *i.e.*, $d^i u_k = u_{k-i}$ and

$$\begin{aligned} W(d) &= \bar{\kappa}_{p+q} d^{p+q} + \bar{\kappa}_{p+q-1} d^{p+q+1} + \dots + \bar{\kappa}_1 d + \bar{\kappa}_0 \\ &= \bar{\kappa}_{p+q} (d - d_1)(d - d_2) \dots (d - d_{p+q}), \end{aligned} \tag{18}$$

with d_1, d_2, \dots, d_{p+q} being the roots of $W(d)$, where the factorization as in (18) can be easily obtained by using standard software. Let $\Omega = \{d_1, d_2, \dots, d_{p+q}\}$ be the set of all roots, and Θ be any subset of Ω consisting of p elements, *i.e.* of cardinality equal to the length of impulse response of the first dynamics. The polynomial $W(d)$ can be factorized as follows

$$W(d) = \bar{\kappa}_{p+q} \Gamma_\Theta(d) \Lambda_\Theta(d), \tag{19}$$

where $\Gamma_\Theta(d) = \prod_{d_i \in \Theta} (d - d_i)$, and $\Lambda_\Theta(d) = \prod_{d_i \in \Omega \setminus \Theta} (d - d_i)$. Since the possible conjugate roots of $W(d)$ must be included to the same factor $\Gamma_\Theta(d)$ or $\Lambda_\Theta(d)$ (consequently, appropriate transfer function), the total number of feasible representations as in (19) does not exceed $\binom{p+q}{p}$ and coefficients of the polynomials

$$\Gamma_\Theta(d) = \gamma_{\Theta,0} + \gamma_{\Theta,1} d + \dots + \gamma_{\Theta,p} d^p, \tag{20}$$

$$\Lambda_\Theta(d) = \lambda_{\Theta,0} + \lambda_{\Theta,1} d + \dots + \lambda_{\Theta,q} d^q \tag{21}$$

yield eventually normalized impulse responses of the first and second linear blocks, *i.e.*

$$g_{\theta} = (\bar{\gamma}_{\theta,0}, \bar{\gamma}_{\theta,1}, \dots, \bar{\gamma}_{\theta,p})^T, \quad l_{\theta} = (\bar{\lambda}_{\theta,0}, \bar{\lambda}_{\theta,1}, \dots, \bar{\lambda}_{\theta,q})^T,$$

where $\bar{\gamma}_{\theta,i} = \frac{\gamma_{\theta,i}}{\sum_{j=0}^p \gamma_{\theta,j}}$, and $\bar{\lambda}_{\theta,i} = \frac{\lambda_{\theta,i}}{\sum_{j=0}^q \lambda_{\theta,j}}$. Including (15) and (16), let us now introduce the following quality index

$$\tilde{Q}_N(l_{\theta}, g_{\theta}) = \frac{1}{N} \sum_{k=1}^N \left[\frac{1}{h_k} \bar{K}\left(\frac{\Delta_k}{h_k}\right) \right] [y_k - \tilde{y}_k(l_{\theta}, g_{\theta})]^2,$$

for each possible l_{θ}, g_{θ} , where $\{h_k\}_{k=1}^N$ is a number sequence tending to zero as $N \rightarrow \infty$. This rule is to some extent an empirical counterpart of (16) with an additional weighting factor $(1/h_k)\bar{K}(\Delta_k/h_k)$, and the modification in comparison with (13) is that now $\bar{K}(\cdot)$ is an anti-kernel of $K(\cdot)$, *i.e.* $\bar{K}(x) = -K(x) + 1$ (compare (13)) guaranteeing that at this estimation stage we exploit the data not taken into account in the estimate (13). Now, as the solution, we can take

$$\{\tilde{\lambda}_N, \tilde{\gamma}_N\} = \arg \min_{\Theta \subset \Omega} \tilde{Q}_N(l_{\theta}, g_{\theta}). \tag{22}$$

Though the proposed procedure is obviously not recursive, it possesses however the fundamental feature of such a scheme. Namely, we repeatedly recompute and compare the value of the same function $\tilde{Q}_N(l_{\theta}, g_{\theta})$ for various intermediate arguments l_{θ}, g_{θ} to find the optimum solution (22), *i.e.* true impulse responses. On the other hand the procedure replaces the possible truly recursive and hard numerical optimization routine of (16), and makes easier solution of our problem for large dimensions of p and q . It realizes in fact exhaustive search strategy w.r.t. a $(p + q)$ -element discrete set of impulse response items, and can be further improved by using more efficient method of discrete programming.

In the empirical counterpart of the procedure we use the estimates $\hat{\mu}_N(\cdot)$ and $\{\hat{\kappa}_{i,N}\}$ instead of $\mu(\cdot)$ and $\{\bar{\kappa}_i\}$, according to the plug-in strategy, getting the estimates $\hat{\lambda}_N, \hat{\gamma}_N$.

6 Simulation example

In the simulation experiment, the Wiener-Hammerstein system with linear subsystems impulse responses $\gamma = (0.6, 0.3, 0.1)^T$, and $\lambda = (0.7, 0.3)^T$ was investigated. The nonlinear static characteristic was chosen as

$$\mu(x, c) = c_1 x + c_2 x^2, \quad \mathbf{c} = (c_1, c_2)^T = (-1, 1)^T.$$

The system was excited and disturbed by random, uniformly distributed sequences $u_k \sim \mathcal{U}[-2, 2]$, and $z_k \sim \mathcal{U}[-0.1, 0.1]$. Unknown parameters were recovered based on $\{(u_k, y_k)\}_{i=1}^{N=30000}$ measurements. The following estimates were obtained

$$\hat{\gamma} = (0.6017, 0.3039, 0.0944)^T, \quad \hat{\lambda} = (0.7048, 0.2952)^T,$$

The simulation results are illustrated in Figures 2a and 2b, respectively. In Figure 2a, the nonparametric estimate (5) evaluated at equispaced points, compared with the true characteristic $\mu(\cdot)$ is shown. In turn, Figure 2b presents a comparison of the outputs of the true system (dots) and estimated system model (circles). The simulation results confirm applicability of the proposed method.

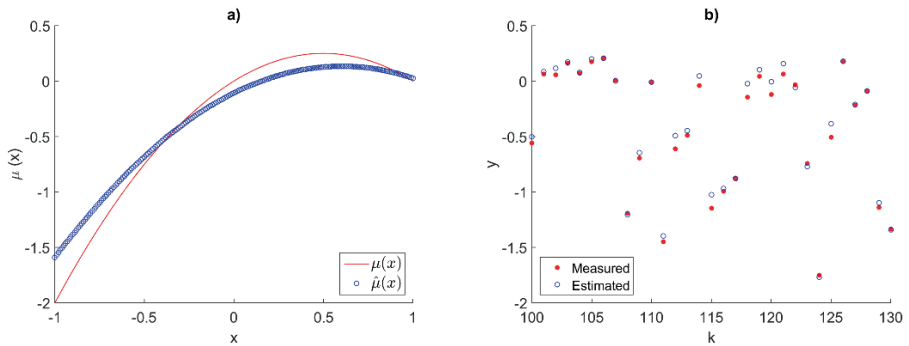


Fig. 2: The results of identification. a) The true characteristic $\mu(x)$ and the estimate $\hat{\mu}(x)$ and b) Outputs of the system and model

7 Conclusions

Based only on the input-output measurements of the whole system, both static and dynamic components of the Wiener-Hammerstein system can be recovered with use of the presented algorithm. The proposed method works for a wide class of excitations and for any probability distribution of random output noise with finite variance. Because of the online (recursive) nature of the estimates, the scheme can be implemented in an adaptive fashion, *e.g.*, for system controllers or devices which require up-to-date knowledge about the process under control. Deconvolution of confounded impulse responses can be performed by using parallel computing techniques and may be efficiently solved even for blocks with long dynamics. The simulation results show efficiency of the method, however rigorous convergent analysis is at present an open problem.

Acknowledgment

The work was partially supported by the National Science Centre, Poland, grant No. 2016/21/B/ST7/02284.

References

1. Giri, F., Bai, E.: Block-Oriented Nonlinear System Identification. Lecture Notes in Control and Information Sciences 404, Springer (2010)
2. Greblicki, W., Pawlak, M.: Nonparametric System Identification. Cambridge University Press; 1 edition (2008)
3. Mzyk, G.: Wiener-Hammerstein system identification with non-Gaussian input. In: IFAC International Workshop on Adaptation and Learning in Control and Signal Processing. Antalya, Turkey (2010)
4. Mzyk, G.: Nonparametric recovering of nonlinearity in Wiener-Hammerstein systems. In: Proceedings of the 9th International Conference on Informatics in Control, Automation and Robotics, ICINCO. pp. 439–445. IEEE, Rome, Italy (2012)
5. Mzyk, G.: Combined Parametric-Nonparametric Identification of Block-Oriented Systems. Lecture Notes in Control and Information Sciences 454, Springer (2014)
6. Mzyk, G., Wachel, P.: Kernel-based identification of Wiener-Hammerstein system. Accepted for Publication in Automatica (2017)
7. Pawlak, M., Hasiewicz, Z., Wachel, P.: On nonparametric identification of Wiener systems. IEEE Transactions on Signal Processing 55(2), 482–492 (2007)
8. Sjöberg, J., Lauwers, L., Schoukens, J.: Identification of Wiener-Hammerstein models: Two algorithms based on the best split of a linear model applied to the SYSID'09 benchmark problem. Control Engineering Practice 20(11), 1119–1125 (2012)
9. Sjöberg, J., Schoukens, J.: Initializing Wiener-Hammerstein models based on partitioning of the best linear approximation. Automatica 48(2), 353–359 (2012)
10. Śliwiński, P., Hasiewicz, Z.: Computational algorithms for multiscale identification of nonlinearities in Hammerstein systems with random inputs. IEEE Transactions on Signal Processing 53(1), 360–364 (2005)
11. Śliwiński, P., Hasiewicz, Z., Wachel, P.: A simple scheme for semi-recursive identification of Hammerstein system nonlinearity by Haar wavelets. International Journal of Applied Mathematics and Computer Science 23(3), 507–520 (2013)
12. Westwick, D., Schoukens, J.: Initial estimates of the linear subsystems of Wiener-Hammerstein models. Automatica 48(11), 2931–2936 (2012)
13. Wills, A., Schön, T., Ljung, L., Ninnes, B.: Identification of Hammerstein-Wiener models. Automatica 49(1), 70–81 (2013)

A new method of multi-inertial systems identification by the Strejc model

Witold Byrski

Department of Automatics and Biomedical Engineering
AGH University of Science and Technology, Krakow, Poland
wby@agh.edu.pl

Abstract. The paper presents a new method of active identification by the use of n th order Strejc model with time delay. The model approximates the dynamics of multi-inertial system based on its step response. The basic and inconvenient version of identification method for this model was published by Strejc in 1959. Different results of research on this model were published in other works. In almost all these methods a key role is played by the graphic procedures for setting coordinates of the inflection point in step response of the system, drawing a tangent line in this point and finally determining the specific intervals given by this line. Based on the Strejc table one can then determine the hypothetical order and the time constant of the model. The reasons for the model approximation errors in this method are the inaccuracy of location of the inflection point, the procedure of tangent line drawing and the main idea that the step response of a real system and the model response are equal only in the one point. The method presented in this work, is based on the condition that the normalized step responses of the system and of the n th order Strejc model must be equal in two chosen points spaced from each other that will guarantee a good matching of the whole step characteristics and hence a good approximation model. As it turned out, the simple analytical formulas can be received which enable fast and simple determination of Strejc model parameters (for any specifying order $n \geq 1$). The most important features of this method are lack of procedure for the location of the inflection point and the tangent line drawing procedure, as well as the non-assumption in advance of the order of the model (in contrast to the Strejc method). The presented method is suitable for easy implementation in PLC controllers with self-tuning.

Keywords: Strejc model identification, systems with time delay, self-tuning PLC

1 Introduction

The paper presents a new method of parameter identification of n th order transfer function with one multiple time constant T_n and time delay τ_n , which approximates the dynamics of an unknown multi-inertial system, based on its step response $h_0(t)$.

The basic version of the identification method for this model was first time published by Strejc in 1959 [1]. Then, additional research concerning this model was presented in many works. In some of them different ideas were used, for instance the

Stirling's formula for factorial approximation or artificially scaled time constant T/n , so as to get one point intersection of the all step responses of Strejc models with different orders (in fact, such a point does not exist). All of these methods originate from the one main tenet that the inflection point of a normalized step characteristic $h_0(t)$ of an unknown real system, as well as the slope of the tangent line in this point, should be the same as in the Strejc model. Thus, the parameters of the single point have to determine the quality of approximation. These standard procedures for determination of the order n , time constant T_n and time delay τ_n use the three stage identification algorithm in which a crucial role is played by the graphic procedure for determination of the inflection point coordinates, determination of the slope of the tangent line and determination of time intervals pointed by this line at the levels $h_0=0$ and $h_0=1$. There are different reasons for the errors during identification of the Strejc model. Besides the main reason, which was the idea of replacing several different time constants by the same and multiple constant T_n , the Strejc method itself introduces some inaccuracies, especially by the use of the idea that the step response and its derivative of the real system are equal to the step response and its derivative of the model, only in the one point. The other sources of the errors are the above-mentioned inconvenient graphic procedures as well as iterative procedures for selection of the proper order of the model (the calculated order of the model most frequently appears to be non-integer number). The graphic errors can be minimized by computer pre-processing of the measurement characteristic in the order of the precise location of the inflection point (the moment of maximum impulse response). Without the use of a computer, graphical inaccuracies are made immediately that give difficult to estimate identification errors.

Therefore some authors developed methods which propose the omitting of these procedures but limited only to the models of the first order. By drawing a secant line through two selected points it is easy to determine a substitute time constant for the Kűpfmüller first order model. To the group of these simple methods belongs a Cohen-Coon method [2], in which, for model identification, two points are chosen on the normalized step characteristic: $h_0(T_{50})=0.5$ and $h_0(T_{63.2})=0.632$. The same methodology for another two points $h_0(T_{28})=0.28$ and $h_0(T_{40})=0.4$ was presented by Broida (1969) [3]. There are also other known results for an additional points $h_0(T_{35.3})=0.353$ and $h_0(T_8)=0.853$. However, the formulas for T and τ obtained by these methods are very simple, they cannot give good results for the systems of the high order.

In order to increase the accuracy of identification, the author of this work developed, in 2015, an extension of two point identification method and obtained its generalization for any order of the Strejc model. The derived formulas were not found in the subject literature (despite searching in literature data basis) and hence it seems that they are quite new. The proposed approximation method uses the condition that the normalized experimental step response ($h_0(\infty)=1$) should always fit the Strejc characteristic of n th order at two specially chosen points spaced from each other that will guarantee a good matching of the entire step characteristics and finally a good approximation model. One of these points is constant e.g. $h_0(T_{90}) = 0.9$ and the second one is chosen as the known inflection point of Strejc model $h[t_p(n)]$ (different for the different order of the chosen model). As it turned out it is possible to obtain very simple

analytical formulas for fast and accurate identification of the Strejc model with time delay and for chosen by the designer the order n of the model. All obtained model's responses are very well-fitted into the real system response because they coincide with two spaced apart points.

However, the most important features of the presented method are the lack of the procedures for examination of the coordinates of inflection point, as well as the lack of graphic setting and tangent line processing. An extra feature is also the fact that the designer may choose the order n of the model, which differs this method from the classical Strejc method. For use of the derived formulas manual calculations and the use of simple four function calculator are sufficient (three multiplications and two subtractions of two numbers are needed). Hence, the presented procedures are especially suitable for easy implementation in self-tuning PLC controllers. Many producers of adaptive PLCs use the Strejc model for auto-tuning. For instance, in the Siemens SIPART controllers DR21, DR22, DR24, according to the company's MP 31 catalogue, the SIEPID adaptive identification and controller tuning method has even been patented.

In the subject literature one can find the results of different research for the Strejc model [2], [3], [4], and [14]–[17] (in Polish) as well as the works in which the utilization of this model for self-tuning controllers is presented [5] – [13].

2 The Strejc Model

The transfer function of the Strejc model is given by the formula (1)

$$G(s) = \frac{K}{(Ts + 1)^n} \quad (1)$$

The response of this model for the step control signal $u(t) = I(t)$ with zero initial conditions is given by the known formula:

$$h(t) = K \left[1 - e^{-\frac{t}{T}} \sum_{i=0}^{n-1} \left(\frac{t}{T} \right)^i \frac{1}{i!} \right] = K \left[1 - e^{-\frac{t}{T}} \left(1 + \frac{t}{T} + \frac{t^2}{2T^2} + \frac{t^3}{3!T^3} + \dots + \frac{t^{n-1}}{(n-1)!T^{n-1}} \right) \right] \quad (2)$$

The impulse response for $u(t) = \delta(t)$ is given by the formula (3) (it is the derivative of (2)). The moment of the inflection point in the step response (2), can be calculated based on the necessary condition of the existence of the extremum of function $g(t)$:

$$g(t) = \frac{K}{T(n-1)!} \left(\frac{t}{T} \right)^{n-1} e^{-\frac{t}{T}} \quad (3)$$

$$\ddot{h}(t) = \dot{g}(t) = \frac{K \cdot t^{n-2}}{(n-1)!T^n} \left[1 - \frac{t}{(n-1)T} \right] \cdot e^{-\frac{t}{T}} = 0 \quad (4)$$

The solution of (4) (apart from the trivial solution $t=0$ and $t=\infty$), gives the time of the inflection point T_p .

$$T_p = T(n-1) \quad (5)$$

However, not obvious is the location of the point T_p on characteristic that can be seen in Fig.1, where the curve of the step response $h(t)$ of the transfer function (1) with $T=2.5$ and $n=4$ is shown. It seems that $T_p \in [6, 8]$. The use of a computer and plotting of $g(t)$ can finally exactly point to the location of $T_p = 7.5$, (this agrees with the calculations according to (5), $T_p=2.5*3=7.5$). In a non-symmetrical case for the multi-inertial system with different time constants e.g. $T_1=1, T_2=2, T_3=3, T_4=4, T_5=5, T_6=6$, the exact location of the point $T_p = 16.4$, is quite uncertain (Fig.4).

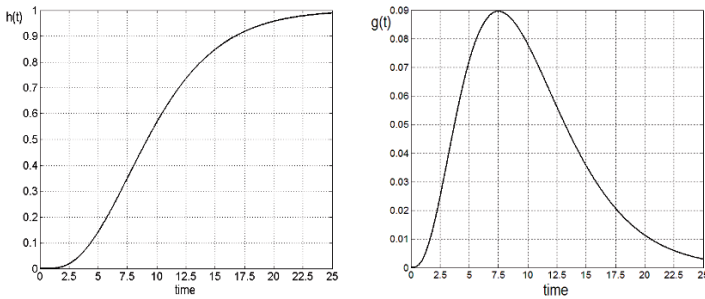


Fig. 1. The step and impulse responses $h(t)$ and $g(t)$ for $n=4$

Substituting (5) for (2) one can obtain the formula for the value of inflection point $h(T_p)$. It turned out that this value does not depend on the time constant T and only depends on the model order n .

$$h(T_p) = K \left[1 - e^{-(n-1)} \sum_{i=0}^{n-1} \frac{(n-1)^i}{i!} \right] = K \left[1 - e^{-(n-1)} \left(1 + \frac{(n-1)^1}{1!} + \frac{(n-1)^2}{2!} + \frac{(n-1)^3}{3!} + \dots + \frac{(n-1)^{n-1}}{(n-1)!} \right) \right] \dots (6)$$

Assuming normalized step characteristic of n -th order model as $h_n(T_p) = h(T_p(n))/K$, one can obtain the known values of h_n and $T_p(n)$ for the Strejc model.

Table 1. The inflation point values for the Strejc model

n	$h(T_p(n))/K = h_n$	$T_p(n)$	$T_p(n)$		
			T=2	T=4	T=5
2	$h_2 = [1 - e^{-1}(1+1)] = 0.2642611$	T	2=T21	4=T22	5
3	$h_3 = [1 - e^{-2}(1+2+2^2/2)] = 0.3233236$	2 T	4	8=T31	10=T32
4	$h_4 = [1 - e^{-3}(1+3+3^2/2+3^3/6)] = 0.3527681$	3 T	6	12	15
5	$h_5 = [1 - e^{-4}(1+4+4^2/2+4^3/6+4^4/24)] = 0.371163$	4 T	8	16	20
6	$h_6 = [1 - e^{-5}(1+5+5^2/2+5^3/6+5^4/24+5^5/120)] = 0.3840$	5 T	10	20	25

The curves $h_n(t)$ denoted **T21** (for $n=2$ and $T=2$) and **T22** (for $n=2$ and $T=4$), as well as **T31** (for $n=3$ and $T=4$) and **T32** (for $n=3$ and $T=5$), are plotted in Fig.2.

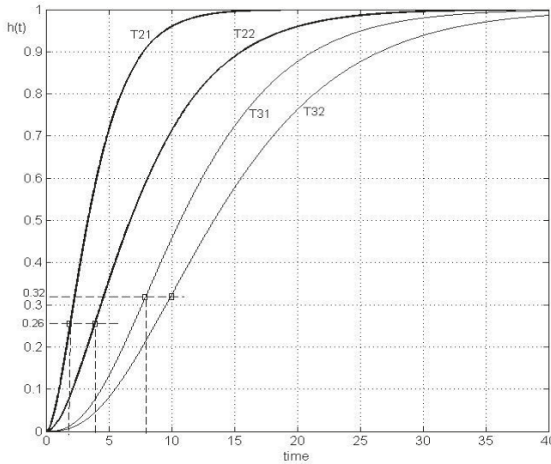


Fig. 2. The step responses for $n=2$ and $n=3$

Important conclusions:

- For the selected n , increasing of the time constant T **does not change** the value of $h_n(T_p(n))$ at the inflection point T_p (curves T21 and T22, and T31 and T32 in Fig.2) but changes the moment of occurrence of the inflection point T_p i.e., **the point T_p moves to the right.**
- For the selected T , increasing of the order n , **moves up** the value $h_n(T_p)$ at the inflection point, and the inflection point T_p , **moves to the right.**

3 The new identification algorithm for an unknown system

In this paper the new identification method for Strejc model with time delay will be presented.

$$G(s) = \frac{K e^{-s\tau_n}}{(T_n s + 1)^n} \tag{7}$$

The step response of an unknown multi-inertial system $h_0(t)$ may be approximated by the step response of the Strejc model of the order n (7) in such a way that both characteristics will be equal at two chosen points – at the inflection point t_{0n} of the model (7), $t_{0n} = \tau_n + T_p = \tau_n + T_n(n-1)$ and $h_0(t_{0n}) = h_n(T_p)$ and at the second point e.g. T_{90} in which normalized characteristic of the unknown system will be equal to characteristic (2), $h_0(T_{90}) = h_n(T_{90} - \tau_n) = 0.9$.

Then the time delay is calculated from: $\tau_n = t_{0n} - T_p = t_{0n} - T_n(n-1)$.

Summarize: from the experimental step response $h_0(t)$ one should only find two moments of time: t_{0n} and T_{90} , where both characteristics should be equal to each other.

Based on this idea, the analytical formulas for parameter identification can be calculated at once. Below will be shown the derivation of these formulas for different model orders n , assuming that the second intersection point will be at time T_{90} , i.e. $\mathbf{h}_0(T_{90}) = \mathbf{h}_n(T_{90} - \tau_n) = \mathbf{0}$. A similar formula for T_{80} will be derived in Section 4.

3.1 The calculation for $n = 2$ and $h_2 = 0.264$ and T_{90} .

For the value h_2 one should find on the experimental characteristic the time t_{02} , for which $h_0(t_{02}) = 0.264$. The formula for time delay $\tau_n = t_{0n} - T_p = t_{0n} - T_n(n-1)$ has a form $\tau_2 = t_{02} - T_p = t_{02} - T_2$.

Denote by the variable x_2 : $x_2 = (T_{90} - t_{02})/T_2$, and assume that $\tau_2 = t_{02} - T_2$, then one can obtain from (2) the formula for the moment $t = T_{90} - \tau_2$:

$$0.9 = 1 - e^{-\frac{(T_{90} - \tau_2)}{T_2}} \left(1 + \frac{T_{90} - \tau_2}{T_2} \right) = 1 - e^{-\frac{(T_{90} - t_{02} + T_2)}{T_2}} \left(1 + \frac{T_{90} - t_{02} + T_2}{T_2} \right) = 1 - e^{-(x_2 + 1)}(1 + x_2 + 1), \quad (8)$$

$$0.1 = e^{-(x_2 + 1)}(x_2 + 2)$$

$$x_2 = -1 + \ln(10 * (x_2 + 2)) = -1 + \ln(10) + \ln(x_2 + 2) = 1.30258 + \ln(2 + x_2) \quad (9)$$

The solution of the last equation can be easily found by the principle of contraction mapping: $x_2 = 2.88972$. Hence,

$$T_2 = \frac{T_{90} - t_{02}}{x_2} = \frac{1}{x_2} * (T_{90} - t_{02}); \quad T_2 = 0.346054 * (T_{90} - t_{02}). \quad (10)$$

$$\tau_2 = t_{02} - T_2 = t_{02} - \frac{T_{90} - t_{02}}{x_2} = \left(1 + \frac{1}{x_2} \right) \cdot t_{02} - \frac{1}{x_2} T_{90}; \quad \tau_2 = 1.346054 * t_{02} - 0.346054 * T_{90}. \quad (11)$$

The Strejc model was identified for $n=2$ with T_2 and time delay τ_2 , based only on the knowledge of the two times T_{90} and t_{02} taken from an experimental characteristic.

3.2 The calculation for $n = 3$ and $h_3 = 0.3233$ and T_{90} .

For the value h_3 one should find on the experimental characteristic the time t_{03} , for which $h_0(t_{03}) = 0.3233$. The formula for time delay $\tau_n = t_{0n} - T_p = t_{0n} - T_n(n-1)$ has a form $\tau_3 = t_{03} - T_p = t_{03} - 2T_3$.

Denote by the variable x_3 , $x_3 = (T_{90} - t_{03})/T_3$, and assume that $\tau_3 = t_{03} - 2T_3$, then one can obtain from (2) the formula for the moment $t = T_{90} - \tau_3$:

$$0.9 = 1 - e^{-\frac{(T_{90} - \tau_3)}{T_3}} \left(1 + \frac{T_{90} - \tau_3}{T_3} + \frac{(T_{90} - \tau_3)^2}{2T_3^2} \right),$$

$$0.9 = 1 - e^{-\frac{(T_{90} - t_{03} + 2T_3)}{T_3}} \left(1 + \frac{T_{90} - t_{03} + 2T_3}{T_3} + \frac{(T_{90} - t_{03} + 2T_3)^2}{2T_3^2} \right),$$

$$0.1 = e^{-(x_3 + 2)} \left(3 + x_3 + \frac{x_3^2}{2} + 2x_3 + 2 \right) = 0.5 e^{-(x_3 + 2)} (10 + 6x_3 + x_3^2),$$

$$x_3 = -2 + \ln \left(\frac{10 + 6x_3 + x_3^2}{0.2} \right) = -0.390562 + \ln(x_3^2 + 6x_3 + 10).$$

The solution of the last equation can be easily found by the principle of contraction mapping $x_3=3.32232$. Hence,

$$T_3 = \frac{T_{90}-t_{03}}{x_3} = \frac{1}{x_3} * (T_{90}-t_{03}); \quad T_3 = 0.30099 * (T_{90}-t_{03}), \quad (12)$$

$$\tau_3 = t_{03} - 2T_3 = t_{03} - 2 \cdot \frac{T_{90}-t_{03}}{x_3} = \left(1 + \frac{2}{x_3}\right) \cdot t_{03} - \frac{2}{x_3} T_{90}; \quad \tau_3 = 1.60199 * t_{03} - 0.60199 * T_{90}. \quad (13)$$

3.3 The calculation for $n=4$ and $h_4 = 0.3527$ and T_{90} .

For the value h_4 one should find on the experimental characteristic the time t_{04} , for which $h_0(t_{04}) = 0.3527$. The formula for time delay $\tau_n = t_{0n} - T_p = t_{0n} - T_n(n-1)$ has a form $\tau_4 = t_{04} - T_p = t_{04} - 3T_4$.

Denote by the variable x_4 : $x_4 = (T_{90} - t_{04})/T_4$, and assume that $\tau_4 = t_{04} - 3T_4$, then one can obtain from (2), the formula for the moment $t = T_{90} - \tau_4$:

$$0.9 = 1 - e^{-\frac{(T_{90}-\tau_4)}{T_4}} \left(1 + \frac{T_{90}-\tau_4}{T_4} + \frac{(T_{90}-\tau_4)^2}{2T_4^2} + \frac{(T_{90}-\tau_4)^3}{6T_4^3} \right)$$

Hence,

$$\begin{aligned} 0.1 &= e^{-(x_4+3)} \left(4 + x_4 + \frac{x_4^2}{2} + 3x_4 + 4.5 + \frac{x_4^3}{6} + \frac{9x_4^2}{6} + \frac{27x_4}{6} + 4.5 \right) = \\ &= (1/6) e^{-(x_4+3)} (78 + 51x_4 + 12x_4^2 + x_4^3), \\ x_4 &= -3 + \ln(10/6) + \ln(x_4^3 + 12x_4^2 + 51x_4 + 78), \\ x_4 &= -2.48917 + \ln(x_4^3 + 12x_4^2 + 51x_4 + 78). \end{aligned}$$

The solution of the last equation can be easily found by the principle of contraction mapping: $x_4=3.68079$. Hence,

$$T_4 = \frac{T_{90}-t_{04}}{x_4} = \frac{1}{x_4} * (T_{90}-t_{04}); \quad T_4 = 0.27168 * (T_{90}-t_{04}); \quad (14)$$

$$\tau_4 = t_{04} - 3T_4 = t_{04} - 3 \cdot \frac{T_{90}-t_{04}}{x_4} = \left(1 + \frac{3}{x_4}\right) \cdot t_{04} - \frac{3}{x_4} T_{90}; \quad \tau_4 = 1.81504 * t_{04} - 0.81504 * T_{90} \quad (15)$$

3.4 The calculation for $n=5$ and $h_5 = 0.3711$ and T_{90} .

For the value h_5 one should find on the experimental characteristic the time t_{05} , for which $h_0(t_{05}) = 0.3711$. The formula for time delay $\tau_n = t_{0n} - T_p = t_{0n} - T_n(n-1)$ has a form $\tau_5 = t_{05} - T_p = t_{05} - 4T_5$.

Denote by the variable x_5 : $x_5 = (T_{90} - t_{05})/T_5$, and assume that $\tau_5 = t_{05} - 4T_5$, then one can obtain from (2), the formula for the moment $t = T_{90} - \tau_5$:

$$\begin{aligned}
 0.9 &= 1 - e^{-\frac{(T_{90}-\tau_5)}{T_5}} \left(1 + \frac{T_{90}-\tau_5}{T_5} + \frac{(T_{90}-\tau_5)^2}{2T_5^2} + \frac{(T_{90}-\tau_5)^3}{6T_5^3} + \frac{(T_{90}-\tau_5)^4}{24T_5^4} \right) = \\
 &= 1 - e^{-\frac{(T_{90}-t_{04}+4T_4)}{T_4}} \left(1 + \frac{T_{90}-t_{05}}{T_5} + 4 + \frac{(T_{90}-t_{05}+4T_4)^2}{2T_5^2} + \frac{(T_{90}-t_{05}+4T_5)^3}{6T_5^3} + \frac{(T_{90}-t_{05}+4T_5)^4}{24T_5^4} \right), \\
 0.1 &= e^{-(x_5+4)} \left(5 + x_5 + \frac{x_5^2}{2} + 4x_5 + 8 + \frac{x_5^3}{6} + \frac{12x_5^2}{6} + \frac{48x_5}{6} + \frac{64}{6} + \frac{x_5^4}{24} + \frac{16x_5^3}{24} + \frac{96x_5^2}{24} + \frac{256x_5}{24} + \frac{256}{24} \right),
 \end{aligned}$$

After some calculation,

$$x_5 = -4.875468737 + \ln \left[824 + 568x_5 + 156x_5^2 + 20x_5^3 + x_5^4 \right].$$

The solution of the last equation can be easily found by the principle of contraction mapping: $x_5 = 3.99359$. Hence,

$$T_5 = \frac{T_{90}-t_{05}}{x_5} = \frac{1}{x_5} * (T_{90}-t_{05}); \quad T_5 = 0.25040 * (T_{90}-t_{05}); \tag{16}$$

$$\tau_5 = t_{05} - 4T_5 = t_{05} - 4 * T_5 = t_{05} - 1.0016051 * (T_{90}-t_{05}); \quad \tau_5 = 2.0016 * t_{05} - 1.0016 * T_{90} \tag{17}$$

All the Strejc models with chosen order n were identified based only on the knowledge of two times: T_{90} and t_{0n} taken from experimental characteristic.

Table 2. The final results for different Strejc models of the order n and the time T_{90} .

n	h_n	T_n	τ_n	τ_n
2	0.264	$T_2 = 0.34605 * (T_{90} - t_{02})$	$\tau_2 = t_{02} - T_2$	$\tau_2 = 1.34605 * t_{02} - 0.34605 * T_{90}$
3	0.323	$T_3 = 0.30099 * (T_{90} - t_{03})$	$\tau_3 = t_{03} - 2T_3$	$\tau_3 = 1.60199 * t_{03} - 0.60199 * T_{90}$
4	0.353	$T_4 = 0.27168 * (T_{90} - t_{04})$	$\tau_4 = t_{04} - 3T_4$	$\tau_4 = 1.81504 * t_{04} - 0.81504 * T_{90}$
5	0.371	$T_5 = 0.25040 * (T_{90} - t_{05})$	$\tau_5 = t_{05} - 4T_5$	$\tau_5 = 2.00160 * t_{05} - 1.00160 * T_{90}$

All these formulas are correct. This can be checked, assuming that the case with $\tau_n = 0$ will happen, then one can calculate T_{90} from the last column of Table 2. Substituting it to the formula in column 3, it occurs that according to (5), $t_{0n} = (n-1)T_n = T_p$.

4 The formulas for the models assuming the second checkpoint time T_{80}

Below, without derivation we present the final formulas for the Strejc models obtained with the assumption of the second time T_{80} , where both normalized characteristics – experimental and given by the model – are equal at the level 0.8, i.e. $h_n(T_{80}-\tau_n) = h_0(T_{80}) = 0.8$.

Table 3. The final results for Strejc models of nth order (new method) for T_{80} .

n	h_n	T_n	τ_n	τ_n
2	0.264	$T_2 = 0.50143 * (T_{80} - t_{02})$	$\tau_2 = t_{02} - T_2$	$\tau_2 = 1.50143 * t_{02} - 0.50143 * T_{80}$
3	0.323	$T_3 = 0.43878 * (T_{80} - t_{03})$	$\tau_3 = t_{03} - 2T_3$	$\tau_3 = 1.87756 * t_{03} - 0.87756 * T_{80}$
4	0.353	$T_4 = 0.39761 * (T_{80} - t_{04})$	$\tau_4 = t_{04} - 3T_4$	$\tau_4 = 2.19282 * t_{04} - 1.19282 * T_{80}$
5	0.371	$T_5 = 0.36751 * (T_{80} - t_{05})$	$\tau_5 = t_{05} - 4T_5$	$\tau_5 = 2.47006 * t_{05} - 1.47006 * T_{80}$

5 The numerical example

For the system of 6th order with transfer function $G(s)$ the step response $h_0(t)$ has been stored:

$$G(s) = \frac{1}{(s+1)(2s+1)(3s+1)(4s+1)(5s+1)(6s+1)} \tag{18}$$

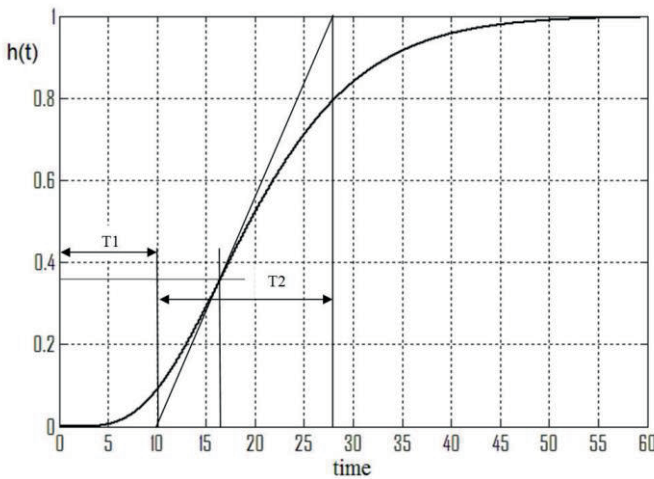


Fig. 3. Step response $h_0(t)$, for the system (18), $n=6$.

The classic Strejc method

Identifying on Fig.3 (with the aid of the computer) the inflection point of experimental characteristic $h_0(t_p)=0.358$ and $t_p=16.4$ and setting graphically the tangent line, one can find that $T1E = 10.0$ and $T2E = 17.8$. The equality $T2=T2E$ holds. Using the basic Strejc table (beneath in Table 4, the one row for $n=4$ from this known table is presented) one can obtain $T=T2/4.463=3.99$, $T_p=3*T=11.97$ and $T1=T*1.425=5.68$. It can be seen that $T1 \neq T1E$. Hence $\tau_4 = t_p - T_p$.

Table 4. The Strejc table [1] (the row for $n=4$) and the identified model

Strejc Table for $n=4$					
n	$h(T_p)$	$T2/T$	$T1/T$	$T1/T2$	T_p/T
4	0.353	4.463	1.425	0.319	3

Model identified by the Strejc method			
$T=T2/4.463$	$T1=T*1.425$	$T_p=3*T$	$\tau_4=t_p - T_p$
3.99	5.68	11.97	4.43

The new method:

From experimental characteristic $h_0(t)$, for the values $h_0(t_{02})=0.2642$, $h_0(t_{03})=0.3233$, $h_0(t_{04})=0.3527$ and $h_0(T_{90})=0.9$, the moments $t_{02}=14.38$, $t_{03}=15.65$, $t_{04}=16.28$ and $T_{90}=33.71$ have been found.

Based on formulas from Table 2, the three models can be immediately found for $n = 2, 3, 4$. The quality of these models was evaluated by the simulation tests and calculation of the integral of the squared error between an experimental step response of the system (18) and response of each model: $\varepsilon(t)=h_0(t)-h_n(t)$ on interval $[0, 80]$ (when all responses practically have the values equal to $h_0(\infty)=1$). The results for the time T_{90} are summarized in Table 5. In this table the Strejc model identified by classical method is also presented.

Table 5. The comparison of the models for $T_{90}=33.71$.

n	h_n	t_{0n}	T_n	τ_n	$\int_0^{80} [\varepsilon(t)]^2 dt$	$\int_0^{80} \varepsilon(t) dt$
2	0.264	14.38	6.689	7.69	0.01415	0.6171
3	0.323	15.65	5.436	4.78	0.00239	0.2267
4	0.353	16.28	4.735	2.075	0.000062	0.0428
4 by Strejc	0.358	16.4	3.99	4.43	0.03762	1.217

Despite the fact that the localization of the inflection point on the experimental characteristic has been done accurately (by the use of computer), one can see from Fig.4 and Table 5, that the Strejc model obtained by classic approach gives the worst quality of approximation, even twice worse than that given by the model of the second order obtained by the new method. The model of 4th order (by the new method) practically covers the experimental characteristic. Thus, the final formula is:

$$G(s) = \frac{1}{(s+1)(2s+1)(3s+1)(4s+1)(5s+1)(6s+1)} \cong \frac{e^{-2.07s}}{(4.73s+1)^4}$$

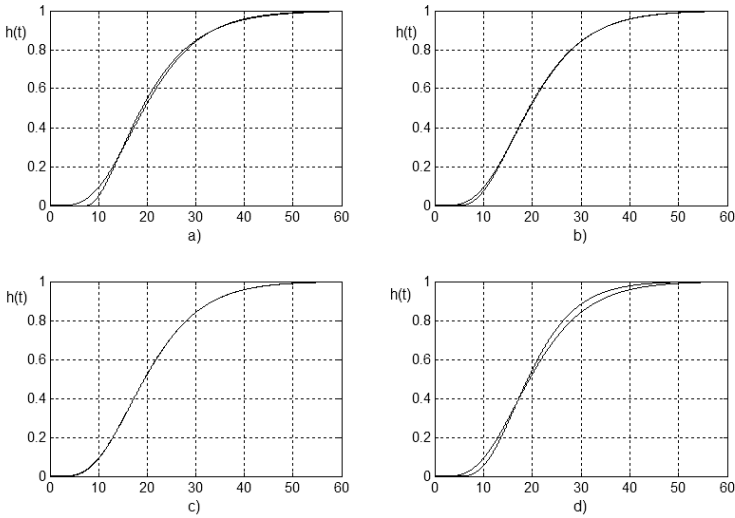


Fig. 4. The comparison of characteristics: a) experimental and the 2nd order (new method), b) experimental and the 3rd order (new method), c) experimental and the 4th order (new method), d) experimental and the 4th order model by the Strejc method

Based on Table 3 the results of identification of system (18) for the time T_{80} are summarized in Table 6. In this table the Strejc model identified by classical method is also presented.

Table 6. The comparison of models for $T_{80} = 28.16$

n	h_n	t_{0n}	T_n	τ_n	$\int_0^{80} [\varepsilon(t)]^2 dt$	$\int_0^{80} \varepsilon(t) dt$
2	0.264	14.38	6.91	7.47	0.01050	0.584
3	0.323	15.65	5.49	4.67	0.00192	0.215
4	0.353	16.28	4.72	2.11	0.000079	0.0474
4 by Strejc	0.358	16.4	3.99	4.43	0.03762	1.217

Comparing Table 5 and Table 6 it can be concluded that for the low order of the models it is better to match the characteristics by the use of time T_{80} and for the higher orders it is better to use the time T_{90} , although the differences look practicably negligible.

The last question is how this method works for the experimental characteristics obtained from noisy measurements. To answer this question an extra experiment was made. For assumed $n=4$, based on noisy characteristic of the model (18) (Fig.5.) the designer can choose two checkpoints and find two corresponding times. The first point is $[h_0(t_{04})=0.35$ and $t_{04}=16.2]$. The second is $[h_0(T_{90})=0.9$ and $T_{90}=33.7]$. Then the formulas from Table 2 give the 4th order model:

$$G(s) = \frac{e^{-1.94s}}{(4.75s+1)^4} \cong \frac{1}{(s+1)(2s+1)(3s+1)(4s+1)(5s+1)(6s+1)}$$

Plotting the two characteristics (Fig.5) – the experimental characteristic with noise and the characteristic given by the above model – one can observe a very good fit of both characteristics. This means that the presented method can also be applied to data with noisy measurements and filtering procedures are not needed. In such a noisy case the Strejc method may cause larger errors due to the procedures of setting of the inflection point and problems with conducting of the tangent line.

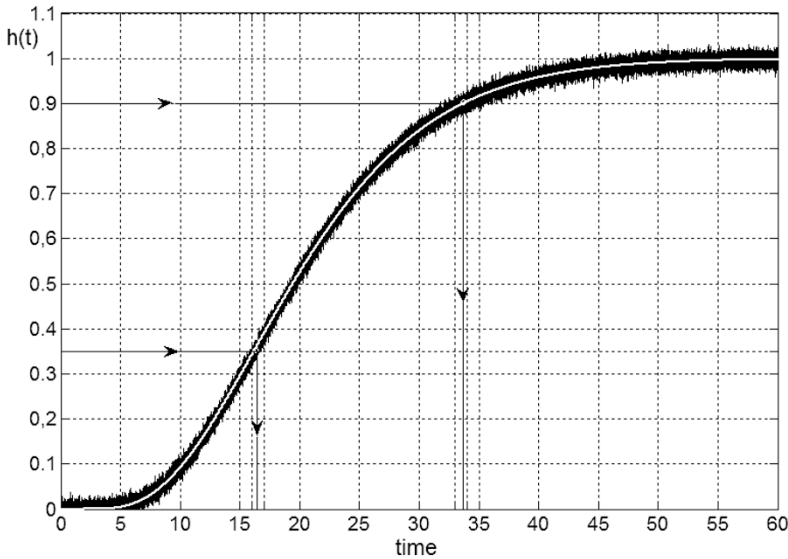


Fig. 5. Fitting of two characteristics – given by the model and by the noisy measurement

6 Conclusions

The obtained new formulas for identification of the Strejc models are very simple so they may be used in PLC self-tuning.

It seems that up to now these formulas were unknown and unpublished. The classical and rather inconvenient Strejc method from 1959 is still being commonly implemented. The obtained new formulas are universal, they do not need the iteration pro-

cedure and, what is more important they do not need the answer where the inflection point is located. The graphical procedure of setting the tangent line is also unnecessary. The identification procedure does not need any comparison simulation procedure (any Runge-Kutta solvers) and, more importantly, no nonlinear optimization procedures in the space R^3 for finding three numbers T , τ , n are needed.

The method requires only two numbers – the times t_{0n} and T_{90} (or T_{80}) which can be found by the direct analysis of the recorded experimental characteristic $h_0(t)$. The designer decides about the order of the model n and the level $h_n(T_p)$ from the Table 1 and chooses from Table 2 or Table 3, suitable row and concurrently value $h_n = h_0(t_{0n})$ and hence the time t_{0n} . The reading of the second time concerns the time T_{90} where $h_0(T_{90})=0.9$ (models from Table 2) or the time T_{80} where $h_0(T_{80})=0.8$ (models from Table 3).

The presented method can also use the noisy measurement characteristics.

In the discussed new method, the inflection points of both characteristics (experimental and model) are not equal to each other because there is no information about where the inflection point of the experimental characteristic is placed. Intuition says that better matching of these two points, should further improve the accuracy of identification. However, the inequality of the location of these points has no particular importance because the second point $h(T_{90})$ or $h(T_{80})$ still ensure good coverage of both characteristics. It is worth to note that the classical Strejc methodology which assumes equalization of the inflection points of experimental and model characteristics, forces the assumption of indicated order n of the model, while the method presented in this paper allows for reasonable selection of the best model of a lower order, e.g. for the reasons of easier synthesis of control system. The extension of Table 2 and Table 3 for higher model orders $n = 6, 7, 8...$ can be quite quickly derived. The formula for any second checkpoint e.g. T_{70} or T_{95} can also be derived. However the results for both T_{90} and T_{80} show high performance of identification, so they can be used for programming self-tuning PLC.

It is obvious that the formulas presented in Table 2 and Table 3 can be used not only for the problems of identification of the existing real system with given experimental characteristic but also for the synthesis of any model (7), if only the designer will select the model order n and the two points of its desirable step response $[h_n(t_{0n}), t_{0n}]$, $[h(T_{90}), T_{90}]$.

References

- [1] Strejc V, Näherungsverfahren für aperiodische Übergangscharakteristiken. Regelungstechnik 7, (1959), p.124–128.
- [2] Cohen, G.H., G.A. Coon, Theoretical Consideration of Retarded Control, Transactions of the ASME, 75, 1953, p.827-834,.
- [3] Broida V., L'extrapolation des reponses indicielles aperiodiques, Automatisme, T.XIV, no.3, 1969, p.105-114.

- [4] Hilscher K., Kennwertbestimmung aus der Übergangsfunktion bei Geräten und Strecken. deren Übertragungsfunktion sich durch n Verzögerungsglieder mit gleicher Zeitkonstante T einschließlich einer Totzeit T_f , approximieren last, Messen, Steuern, Regeln. v.7, H.8. 1964, p.272.
- [5] O'Dwyer A., Handbook of PI and PID Controller, Tuning Rules, 3rd ed. Imperial College Press, Dublin, 1848162421
- [6] Ziegler J.G., N.B., Nichols, Optimum settings for automatic controllers. *Transactions of ASME* 64, 1942, 759–768.
- [7] Åström K.J., T. Hägglund, Automatic tuning of simple regulators with specifications on phase and amplitude margins, *Automatica*, 20, 1984, 645-651.
- [8] Åström K.J., C.C. Hang, P. Persson & W.K. Ho. Towards intelligent PID control. *Automatica*, 28, 1992, 1-9.
- [9] Hang C.C., K.J. Åström, & W.K., Ho, Refinements of the Ziegler-Nichols Tuning Formula. In: *IEE Proc.Design, Control Theory and Appl.* 138, 1991. pp. 111– 118
- [10] Schei T.S., Automatic tuning of PID controllers based on transfer function estimation. *Automatica*, 30, 1994, 1983-1986.
- [11] Scali C., G. Marchetti, & D. Semino, Relay with additional delay for identification and autotuning of completely unknown processes, *Ind. Eng. Chem. Res.*, 38, 1999, p.1987-1997.
- [12] Leva A., C. Cox & A. Ruano. Hands-on PID autotuning: a guide to better utilisation. *IFAC Professional Brief*, 2002
- [13] Dey C., R. Mudi, An improved auto-tuning scheme for PID controllers, *ISA Transactions*, 48, 2009, p.396-409.
- [14] Kolaj W., J.Mozaryn, M.Syfert, PLC-PIDTuner: Application for PID Tuning with SIMATIC S7 PLC Controllers, <http://www.academia.edu/28147156/PLC>
- [15] Żuchowski A., O pewnej metodzie wyznaczenia parametrów modelu Strejca, PAK, vol.10,2/1993,s.33-35.
- [16] Żuchowski A., Wyznaczanie parametrów rozszerzonego modelu Strejca, w oparciu o pomiar charakterystyki skokowej, PAK, 7/2000, s.6-9.
- [17] Żuchowski A., Nietypowe metody eksperymentalnego wyznaczenia parametrów zastępczego modelu Strejca, PAK, vol. 59, 1/2013, s.55-58.

Graph description of the process and its applications

Jan Maciej Kościelny*, Anna Sztyber**, Michał Syfert***

Institute of Automatic Control and Robotics, Warsaw University of Technology

***jmk@mchtr.pw.edu.pl, **sztyber@mchtr.pw.edu.pl,

***m.syfert@mchtr.pw.edu.pl,

Abstract. The paper presents the method of qualitative modeling of industrial process in the form of cause-and-effect graph that directly takes into account fault influence on process variables. Selected applications of that graph are briefly characterized. Its usefulness in alarm analyzes, designing of process diagnostics and HAZOP safety analyses is described.

Keywords: graph of process, alarm analysis, diagnostics design, HAZOP safety analysis

1 Introduction

Knowledge of process models is required for realization of control tasks, optimization, diagnostics, operators training and others. Acquiring models is usually an essential part of the costs related with implementation of these tasks. Quantitative models describing physical phenomena taking place in the installation are the most complete mathematical description of the process. Building quantitative model for complex industrial installation and experimental defining of its parameters is difficult, time-consuming and expensive. The difficulties increase several times when fault influence should be taken into account by the model created for the purpose of on-line diagnostics or implementing process simulators. Because of these, the nonlinear analytical models are usually developed only for the part of the process that are critical in terms of safety. Nuclear power plants or planes are the examples.

Building models based on measurement data acquired by the historians in the control systems is a different approach. Neural, fuzzy, statistical or linear parametric models are created this way. Such models represent process operation (functioning) in normal state (fault free). Acquiring measurement data for process states with faults is usually impossible.

Designing complete analytical models for industrial processes, that are not related with high risk, e.g. food industry, may not be justified because of too big modelling costs. Qualitative models are much more simple to build. In many cases, they can be the basis for realizing a number of tasks, in particular, alarm analysis and fault detection and isolation.

Some of the qualitative modeling methods (de Kleer's and Brown's qualitative physics, Forbus' qualitative process theory and Kuipers' qualitative simulation) were

described in monograph of Gatnar [3]. Most common among the qualitative models applied in diagnostics are: directed graphs, bond graphs [15] and structural model [1]. The survey of other qualitative models used in diagnostics can be found in work [16].

In this work the qualitative process model in the form of cause-and-effect graph is presented. Numerous applications of such model are also briefly described.

Cause-and-effect graphs reflects the relations between process variables (signals), regardless of whether they are measured or not. Most commonly used are directed SDG graphs (Signed Directed Graph) and their modifications [4, 7, 10]. In SDG graphs, process variables correspond to the graph nodes, while directed graph arcs represent cause-and-effect relations between variables. Graph branches are usually signed. Positive influence denotes compliance, while negative one inconformity of the directions of change of the values of process variables.

Presented cause-and-effect process graph GP differs from the SDG graphs by directly taking into account the influence of faults on cause-and-effect relations describing the process. The idea of the graph was presented in work [5], and its extensions were given in works [8, 11, 13].

Cause-and-effect graphs are created based on the expert's knowledge. However, there is also possible an alternative approach involving the discovery of knowledge about the relations between process variables form the historians [14]. Knowledge discovered with the use of statistical techniques of data mining is expressed in the cause-and-effect relations between control and measured signals. The graph build in such a way does not represent the signals with unmeasured values. The main difficulty during the constructing of such graphs in an automatic way are the feedbacks existing in the process itself, independently from control loops.

2 Qualitative model – GP graph

Graph of the process GP is a qualitative model describing cause-and-effect relationships between variables in the process with taking into account the influence of faults [5, 8, 11, 13]. Directed GP graph is an extension of known SDG graphs, which are used for representation of cause-and-effect relationships between variables or alarms in the technological installation.

Vertices of the GP graph reflect variables, while arcs represent the influence of particular variables on each other. The set of all actual quantities (variables) characterizing the process is denoted as X . The following subsets can be distinguished in this set:

$$X = X_U \cup X_D \cup X_X \cup X_Y, \quad (1)$$

where: X_U – set of control variables, X_D – set of input variables of unknown values (disturbances), X_X – set of state (internal) variables (not measured), X_Y – set of output variables (measured).

Control system generates control signals $u \in U$. They are equal to actual control signals $x \in X_U$ (Fig. 1) in the case of lack of control paths faults. All variables $x \in X_Y$ are measured. Thus, the set of measured signals Y is equally large as the set X_Y and

the values of corresponding elements of those sets are consistent (taking into account the accuracy of measuring devices) in the case of lack of measurement paths faults.

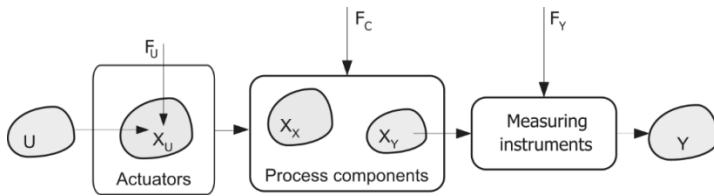


Fig. 1. Influence of measurement and control paths faults on process variables

In the set F of possible faults of the process one can distinguish: fault of control paths F_U , faults of process components F_C and faults of measurement paths F_Y :

$$F = F_U \cup F_C \cup F_Y. \tag{2}$$

Furthermore, the disturbances D , which values are unknown, influence the process. Thus, the set of values V characterizing the diagnosed process can be divided into the following disjoint subsets:

$$V = U \cup Y \cup X \cup F \cup D. \tag{3}$$

From the formal point of view, the GP graph is a Berg graph with no loops (directed graph without loops). It can be noted in the following form [5]:

$$GP = \langle V, A \rangle, \tag{4}$$

$$A \subset V \times V, |V| = n, |A| = m, \tag{5}$$

where: V – set of vertices (synonymous with (3)), A – two-part relation defined on the set of vertices; the set of ordered pairs representing the arcs of the graph.

It is possible to isolate several subgraphs in the GP graph in respect to the type of vertices that create them. Such division and designation of interrelations between subgraphs are shown in Fig. 2.

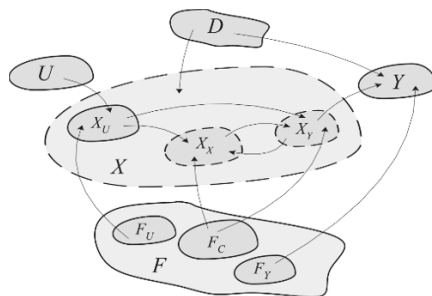


Fig. 2. Structure of GP graph. Areas bounded by a continuous line denote independent subgraphs

The subset X of the vertices of the GP graph forms subgraph GP_X . This graph represents the relations between actual variables characterizing the process. Let it be named as the graph of actual process variables and noted in the following way:

$$GP_X = \langle X, A_X \rangle, A_X \subset X \times X. \tag{6}$$

Process variables Z , which values are known, are used for diagnosing. Thus: $Z = U \cup Y$.

The diagram of serially connected three tanks that store toxic substance is shown in Fig. 3. Graph of the process for that assembly is shown in Fig. 4. The list of symbols of process variables and faults are given in work [13].

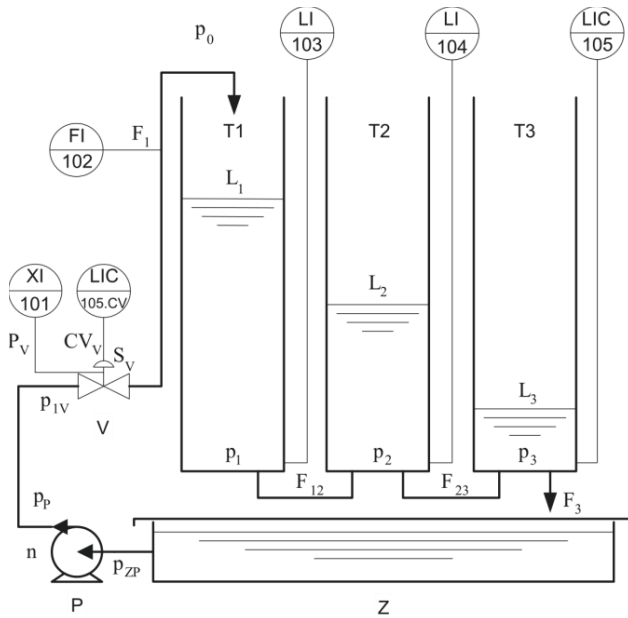


Fig. 3. Assembly of serially connected three tanks storing toxic substance

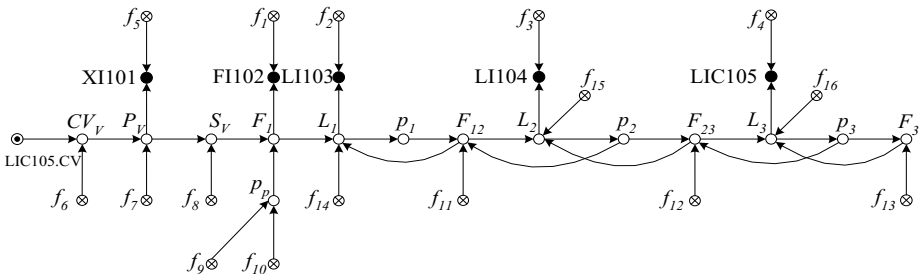


Fig. 4. GP graph for three tank assembly

3 Applications

3.1 Constructing qualitative model

The definition of the relations between process variables in the form of cause-and-effect graph should be the first step in the construction of quantitative model, e.g. the simulator for operators training. Knowing the qualitative model, one can continue finding physical relations between process variables or identifying the relations using archival measurements. This approach allows to build process simulators. Where it is possible, one uses physical analytical equations to model processes taking place in the installation. In the other case, one uses approximation models, which should imitate real process with required accuracy. For this purpose one can use neural or fuzzy TSK models. Different types of partial process models are obtained in this way. Their mutual connectivity is described by a graph of a process. The partial models can be connected and joined together to obtain one simulator.

Such approach for building simulators has many advantages. It enables joining the knowledge about process defined in the form of models describing physical equations with the knowledge discovery using archival measurement databases.

3.2 Reduction of the set of alarms

The alarm overload is a common problem taking place in the process control and supervisory systems. The data of EEMUA (The Engineering Equipment and Materials Users' Association) shows that the daily average of alarms number in petrochemical industry is about 1500, and in power industry about 2000, while, according to the recommendations it should not exceed 144. It is quite common, that the number of alarms appearing in the period of several minutes is greater than 100, and sometimes reaches 700. The interpretation of a huge number of alarms is a serious problem for the process operators. It leads to an information overload and, as a consequence, to high stress. This process can lead to the mistakes of operator handling, which, cumulating with existing faults, can cause major accidents.

Therefore, it is very important to limit the number of alarms, especially to reduce the alarms generated by one, common root cause. Each alarm should be helpful and informative for the operator, and the set of alarms should simplify the isolation of arising hazards (including faults). Graph of a process is very helpful for this purpose. It allows to find the set of alarms being a consequence of a particular fault. Therefore, it allows to eliminate redundant alarms raised by a common cause, and the alarms not important in respect to isolation of particular faults.

3.3 Designing diagnostics based on alarms

The alarm system acts as a simple diagnostic system of the process in commonly used decentralized control systems (DCS) as well as in the systems of supervisory control and monitoring (SCADA). In the alarm systems the methods of limits control are used for fault detection. Fault isolation is usually not conducted in such systems.

It is possible to design, based on GP graph, the system reasoning about faults utilizing alarms. Base on the analysis of GP graph it is possible to determine the set of alarms being the consequence of a given fault. In many cases, it is also possible to determine the sequence of alarms corresponding to that fault. After the reduction of negligible alarms it become possible to design the rules for particular faults where alarms are rules premises and conclusions point out possible faults.

The exemplary rules for faults of the assembly of tanks presented in Fig. 1 that can be derived from its GP graph are given below.

The rule for fault f_{14} (leakage from tank 1) has the following form:

$$\text{if } (LO_{L_1} \rightarrow LO_{L_2} \rightarrow LO_{L_3}) \text{ then } f_{14}. \quad (7)$$

The rule for fault f_{15} (leakage from tank 2) has the following form:

$$\text{if } (LO_{L_2} \rightarrow (LO_{L_1} \text{ and } LO_{L_3})) \text{ then } f_{15}. \quad (8)$$

The rule for fault f_{11} (clogging of the pipe between tanks 1 and 2) has the following form:

$$\text{if } (HI_{L_1} \rightarrow LO_{L_2} \rightarrow LO_{L_3}) \text{ then } f_{11}. \quad (9)$$

The rule premises can be identical for some of the faults. It means, that such faults are unisolable. Instead of two contradictory rules (with identical premises and different consequences) one rule that points out both faults connected with alternate symbol in the conclusion should be created.

3.4 Determining the structures of models used for diagnostic purpose

When diagnostic system uses local models tuned with archival process data (neural, fuzzy or parametric models) it is an important issue to select the model inputs and outputs. Graph GP allows us to analyze causal relationships between variables which lead to the selection of appropriate model structures. The model structure is a pair: output (modelled) variable and the set of input variables. The proper model structure should fulfill the following conditions:

1. The set of inputs should be complete, i.e. all process variables influencing modelled variable and independent from the other inputs should be included in the inputs set.
2. Input variables should be independent.

The dependency is understood as an existence of a causal path between variables in a graph. The formal description of a model structures and the algorithm for finding all model structures were presented in [13].

The exemplary model structure $m(L_1)$ for a three tank assembly from Fig. 1 was shown in Fig. 5.

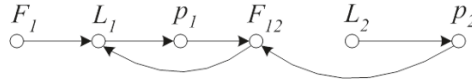


Fig. 5. Model structure $m(L_1) = \{F_1, L_2\}$

3.5 Determining the faults-symptoms relation

Each model structure can be used to build a model and a residual. One can easily find the set of faults influencing this residual by supplementing the sub-graph of a model structure with faults. The model structure is sensitive to all the faults present in the sub-graph.

Therefore for each model structure one obtain a row of a binary diagnostic matrix containing ones for faults included in a sub-graph of a model structure.

The example of finding faults residual based on model $m(L_1)$ in shown in Fig. 6. The exemplary part of a binary diagnostic matrix is shown in Fig. 7.

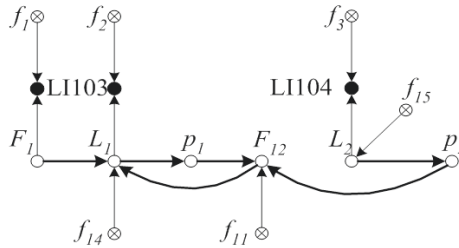


Fig. 6. Faults influencing model structure $m(L_1) = \{F_1, L_2\}$

S / F	f_1	f_2	f_3	...	f_{11}	...	f_{14}	f_{15}	...
			
$s_i(r_{L_1})$	1	1	1	...	1	...	1	1	...
			

Fig. 7. Sensitivity to faults of a model structure $m(L_1) = \{F_1, L_2\}$

In work [6] it was shown, that GP graph is a source of knowledge acquisition about the forming sequence of fault symptoms. Such knowledge can be utilized to increase fault distinguishability. Some of the faults unisolable based on set of tests can be isolable based on different symptoms sequences.

3.6 Selection of optimal set of measurements and set of tests for advanced process diagnostics (based on quantitative models)

Graph GP can be used to find all model structures and their sensitivity to faults. This information allows us to decide which sensors are needed by a diagnostic system.

However, this is a complex optimization problem. The example of an optimization algorithm implementation can be found in [9].

Causal relations also provide qualitative insight into relations between faults and process variables – these dependencies can help us in solving sensor placement problem. Let us consider exemplary graph presented in Fig. 8. The fault f_2 influences variables b and c , fault f_3 affects only variable c . This means that to isolate the faults f_2 and f_3 we need to place a sensor on variable b . The full description of the sensor placement algorithm using GP was presented in [12].

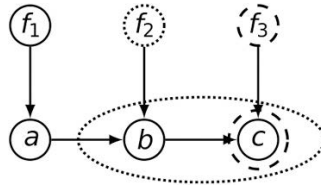


Fig. 8. Exemplary GP graph

3.7 Supporting HAZOP analysis

The most common used method of hazard assessment in the process industry is HAZOP, i.e. hazard and operability study. The state of the art of HAZOP approach was presented in work [2].

The HAZOP methodology does not guarantee the completeness of the risk analysis. In the case of complex systems with very long list of hazards there is a risk of omitting important hazards. The installation under analysis is divided into nodes. During the analysis separated nodes are considered. In the most cases the nodes are not independent. There are mutual relations among them, which may stay unnoticed. The existence of internal feedback loops between nodes is especially dangerous. The quality of the analysis depends only on the competence and experience of the working team.

Application of GP graph to support HAZOP analysis allows to assure higher degree of completeness of the analysis. Achieving higher completeness can be reached by systematic listing of all potential hazards and operational problems, as well as the possibility to include all causes of deviations of process parameters. Process graph models and presents all feedback influences taking place in the process. Therefore, its use can increase the completeness of the analysis. It is illustrated in the example below.

Let one consider the assembly of tanks shown in Fig. 1. This assembly contains internal feedback loops, which make hazard analysis harder, particularly in the case when the process is divided into nodes. The assembly was divided into two nodes during HAZOP analysis. The first node contains pump, control valve and tank 1. The second one contains two tanks with toxic substance. The division of the process into the nodes is shown in Fig. 9. Due to the serial connection of tanks and the task of level control in the tank 3, the main hazard is the danger of tank 1 overflow. Leakages of the toxic substance from the tanks are also dangerous.

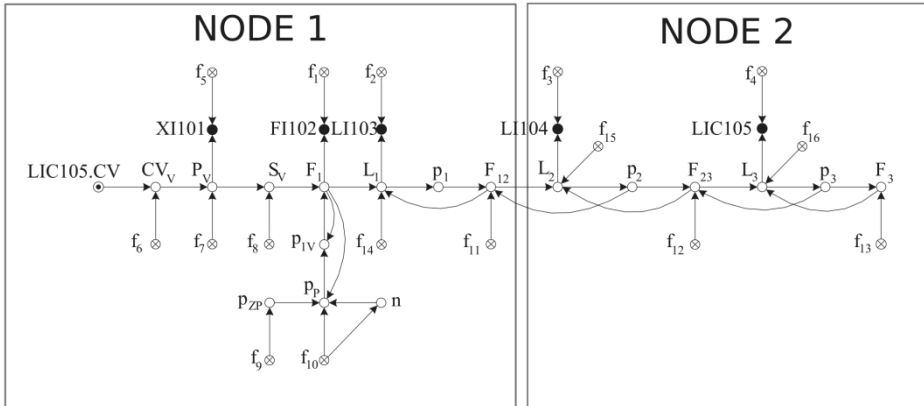


Fig. 9. Division of the process graph from Fig. 1 onto nodes of HAZOP analysis

GP graph shows that, independently on the causes taking place in the node 1, the clogging of the pipe between tanks 2 and 3 as well as the clogging of the outlet of tank 3 can be a causes of tank 1 overflow. Whereas, the too low level in the tank 1 can arise as a result of leakage from the tanks in node 2.

4 Summary

The aim of this paper was to present quantitative process model in the form of cause-and-effect graph with its numerous applications. Special feature of that model is direct consideration of fault influence onto the cause-and-effect relations in the process. Therefore, it is particularly useful in the design of advanced diagnostic systems for complex industrial processes. It allows to find structures of models used for fault detection and to define fault-symptoms relationship. It can be also used as a tool for optimal sensor placement and to select diagnostic tests.

The GP graph is also useful in the alarm analysis and designing simple, alarm based diagnostic systems. Finally, it was shown that the graph of a process is a suitable tool for HAZOP process risk analysis. It allows to take into account all causes of process parameter deviations, thus increases the completeness of the analysis.

5 Bibliography

1. Blanke M., Kiannaert M, Lunze M. and Staroswiecki M. Diagnosis and Fault-Tolerant Control. Berlin: Springer-Verlag, 2003.
2. Dunjo J., Fthenakis V., Vilchez J.A., Arnaldos J.: Hazard and operability (HAZOP) analizys. A literature reviw. Journal of Hazardous Materials, 173 (2010) 19-32.
3. Gatnar E. Methods of qualitative modeling (in Polish). Akademicka Oficyna Wydawnicza PLJ, Warszawa 1994.
4. Iri M., Aoki K., O'Shima E., Matsuyama H., An algorithm for diagnosis of system failures in the chemical process. Computers & Chemical Engineering, Vol.3., 1979, 489-493.

5. Kościelny J. M., Ostasz A., Application of Causal Graph GP for Description of Diagnosed Process, 5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes – SAFEPROCESS'2003, Washington, USA June 9-11, 2003, pp. 879–884.
6. Kościelny J.M., Syfert M., Rostek K., Szyber A.: Fault Isolability with Different Forms of Faults-Symptoms Relation. *Int. J. Appl. Math. Comput. Sci.* 2016, Vol. 26, No. 4, 815–826, (DOI:<http://dx.doi.org/10.1515/amcs-2016-0058>).
7. Montmain J. Leyval L., Causal Graphs for Model Based Diagnosis. IFAC Symposium on Fault Detection, Supervision and Safety for Technical Process – SAFEPROCESS'1994, Espoo, Finland, Vol.1, 347-355.
8. Ostasz A.: Cause-and-effect graph of the process and its application for determining the set of residuals and diagnostic relations (in Polish). PhD. thesis., Warsaw University of Technology, Faculty of Mechatronics, Warszawa 2007
9. Rostek K.: Influence of multi-valued diagnostic signals on optimal sensor placement, *Journal of Physics: Conference Series* 783, 2017, 13th European Workshop on Advanced Control and Diagnosis (ACD 2016), 17–18 November 2016, Lille, France
10. Shiozaki J, Matsuyama H., Tano K., O'Shima, Fault Diagnosis of Chemical Processes by the Use of Signed Directed Graphs: Extension to Five-Range Patterns of Abnormality. *International Chemical Engineering*, Vol. 37, No. 4, 1985, 651-659.
11. Szyber A.: The method of selecting the set of sensors for diagnostics of industrial processes based on cause-and-effect graph (in Polish). PhD. thesis, Warsaw University of Technology, Faculty of Mechatronics, 2015.
12. Szyber A.: Sensor Placement for Fault Diagnosis Using Graph of a Process, *Journal of Physics: Conference Series* 783, 2017, 13th European Workshop on Advanced Control and Diagnosis (ACD 2016), 17–18 November 2016, Lille, France
13. Szyber A., Ostasz A., Kościelny J.M.: Graph of a Process - a new tool for finding model's structures in model based diagnosis. *IEEE TRANSACTIONS ON SYSTEM, MAN, AND CYBERNETICS: SYSTEMS.* (DOI: 10.1109/TSMC.2014.2384000)
14. Tabor Ł.: Modified method of constructing cause-and-effect graphs with use of historical signal time series (in Polish). *Pomiary Automatyka Kontrola*, 2012, R.58, nr.1, 101-104.
15. Thoma J. and Bouamama B. O., *Modelling and Simulation in thermal and Chemical Engineering*. Berlin: Springer-Verlag, 2000.
16. Venkatasubramanian V, Rengaswamy R, Yin K, Kavuri SN. A review of process fault detection and diagnosis, Part II: Qualitative model and search strategies. *Computers and Chemical Engineering* 2003;27:313–326.

The dynamics of the straw combustion process in the batch-fired straw boiler

Wojciech Kreft

wkref@agh.edu.pl

AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering
Department of Automatics and Biomedical Engineering
Al. Mickiewicza 30, 30-059 Krakow, Poland

Abstract. The paper presents the straw combustion process in the biomass boiler in terms of its dynamics. Straw combustion takes place in a specially adapted chamber of the boiler. The analyzed biomass boiler has also a water jacket, which stores heat received from the burning straw. The thermal energy stored in the water jacket is used to heat the building.

Author shows the original model of the straw combustion process which is based on the assumption that the loaded straw has the shape of a cube, and it still maintains this shape during combustion. In this model, it was also assumed that the combustion speed is proportional to the current burning straw surface, air mass flow and burning speed factor, which is characterized for a particular type of straw. The paper presents a mathematical model of the batch-fired straw boiler. That model was simulated in MATLAB/Simulink. The model time data was matched to the real time data of combustion thus making identification of some system parameters.

Keywords: biomass, combustion, modelling, renewable energy sources

1 Introduction

Nowadays more and more attention is given to questions of environmental protection. The most discussed issues are threats caused by emissions generated by energy conversion processes. Large amounts of harmful pollutants (oxides of carbon, nitrogen and sulphur as well as particulates, dioxins and other toxins) are introduced to atmosphere in connection with combustion of fossil fuels. In particular, a man-made emission of CO_2 is considered as the main reason for global warming [1, 2, 3]. International community tries to commit all states in the world to reduce emissions of greenhouse gases. One of the most important steps taken in this direction was the international treaty called the Kyoto Protocol which was concluded under auspices of the United Nations in 1997 and entered into force in 2005 [4].

According to the International Energy Agency (IEA) standpoint various forms of biomass, i.e. solid biomass, biogas and liquid biofuels, are recognized as renewable energy sources. Generally, biomass is defined as organic matter, especially

plant matter, that can be converted to fuel and is therefore regarded as a potential energy source.

The paper concerns the combustion of straw for water heating. Straw is a by-product of corn production. Traditionally, it is used as a fodder and bedding for animals or it is utilized as fertilizer (comes directly under the plough). The use of straw for heating purposes is not yet widespread, however, presently it gains momentum. Heating value of straw is about 16 MJ/kg. To compare, heating value of steam coals (thermal coals) is about 27 MJ/kg. From the point of view of energy content of 1.5 ton of straw is roughly equivalent to 1.0 ton of an average steam coal [5].

Firing with straw for water heating is carried out in purposely designed boilers which can be used for heating farms, small housing estates or public buildings.

This work deals with mathematical description of straw combustion in the batch-fired straw boiler.

2 The straw combustion process in the batch-fired boiler

An outline of the analysed boiler is shown in Figure 1. The boiler consists of the combustion chamber of cubic shape and the water jacket. The cubic bale of pressed straw is stoked into the combustion chamber. The batch of straw is manually set on fire through a special small pipe. Combustion air is forced by a fan and blown into a surface of the straw bale to sustain its burning. Water circulates continuously between the boiler and the rest of the heating system. Water is warmed up in the water jacket.

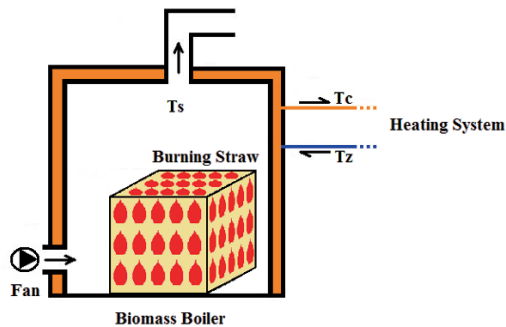


Fig. 1. The scheme of the batch-fired straw boiler

There is a few ways to model the phenomenon of straw combustion in the boiler. To analyse in detail, one should describe the physical and chemical phenomena occurring during the combustion process. The physical processes are heat and mass transfer, and the chemical processes are chemical reactions during combustion of straw.

From a physical point of view, straw combustion in the boiler means a thermal energy emanation in the combustion chamber and the progressive lowering of the

straw mass for gas (exhaust) and ash. The ash created from burned straw represents only about 4% of its original mass [5].

There are various ways of modelling the combustion process in the batch-fired straw boiler [6, 7, 8, 9, 10]. However, the models for practical purposes (e.g., for the purpose of control the entire heating system) does not need to describe precisely the physical and chemical phenomena occurring in the batch-fired straw boilers, but there is sufficient to take into account the most important aspects. To do this, one should analyse static and dynamic properties of the system.

In general, analysing the combustion process one can conclude that the mass flow of gases discharged from the combustion chamber to the chimney, is the sum of the mass flows of the supplied air (using a fan) and substances releasing from the straw during the process of combustion. Undoubtedly, the combustion speed depends on the flow of supplied air and the kind of fuel. The combustion takes place on the surface of the straw block, because that surface is the only place where there is a contact between fuel and air (oxygen), that are necessary to maintain the combustion process. Since the combustion takes place on the surface, the grater is the burning surface, the greater is the combustion speed. The straw combustion causes a mass loss from currently burning surface. Consequently, during combustion, the straw mass is steadily decreasing. The change in straw mass during the combustion process can be represented by differential equation (1). This equation describes the straw combustion process with the following assumptions:

- Straw is loaded in the form of a cube and its shape is maintained during combustion
- The mass of ash is negligibly small in relation to the original mass of straw
- The combustion speed is proportional to the air flow and the current burning surface

$$\frac{d}{dt}M_s(t) = -5\alpha F_p(t) \left(\frac{M_s(t)}{\rho_s} \right)^{\frac{2}{3}} \quad (1)$$

where:

- t – time,
- $M_s(t)$ – straw mass,
- $F_p(t)$ – air mass flow,
- ρ_s – straw density,
- α – straw burning speed factor.

The scheme of the analysed combustion process is shown in Figure 2. To clarify the equation (1), there is necessary to describe the relationship between the surface and the volume of the cube. The volume of straw cube is $V(t) = \frac{M_s(t)}{\rho_s}$. On the other hand is $V(t) = (a(t))^3$, where $a(t)$ is the length of the straw cube edge. The

burning area is $S(t) = 5(a(t))^2$, because 5 faces of the straw cube is burning. Consequently, $S(t) = 5(V(t))^{2/3}$, thus $S(t) = 5\left(\frac{M_s(t)}{\rho_s}\right)^{2/3}$.

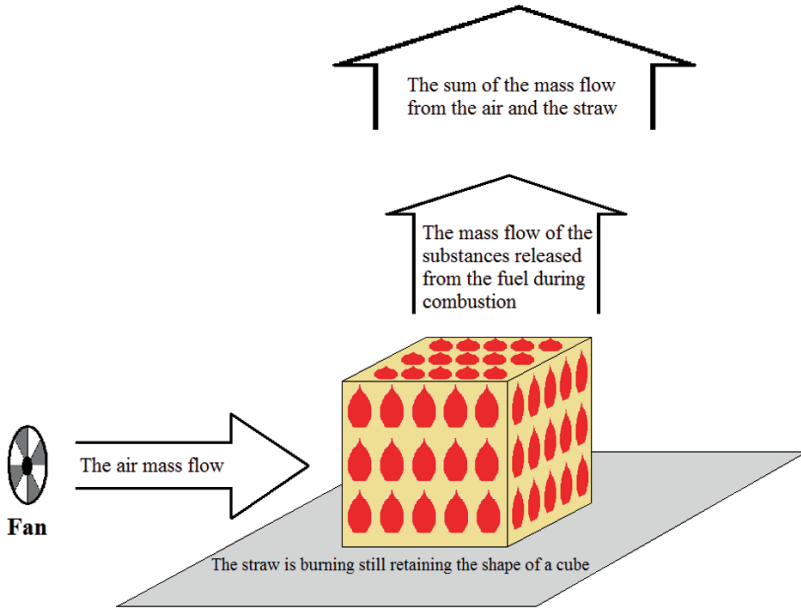


Fig. 2. The scheme of straw combustion and mass balances in the boiler during combustion

If the straw combustion speed is proportional to the current burning surface and the air flow, this can be written in the formula (2).

$$\frac{d}{dt} M_s(t) = -\alpha S(t) F_p(t) \tag{2}$$

The negative sign in equation (2) indicates that the burning straw causes a mass loss, and thus the time derivative of the straw mass is negative. Substituting formula for $S(t)$ to the equation (2), we get the equation (1).

The straw combustion process should be supplemented by the equation for emerging thermal energy during combustion process. This problem is connected with the heating value of the burning straw. The heating value determines how much thermal energy is obtained from the straw mass unit combustion. Thus, the thermal energy per time unit (i.e. power) released from straw during combustion is equal to the straw mass burned in the same time unit multiplied by the straw heating value. The value of the straw mass burned per time unit has the opposite sign than the value of the time derivative of the straw mass. Hence, the thermal power released from the straw during combustion is described by the formula (3).

$$P(t) = -\lambda \frac{d}{dt} M_s(t) \quad (3)$$

where:

t – time,

$P(t)$ – emerging thermal power during combustion process,

$M_s(t)$ – straw mass,

λ – straw heating value.

2.1 The solution of the differential equation for the straw combustion

The equation (1) can be solved by converting to the form (4).

$$M_s(t)^{-\frac{2}{3}} \frac{d}{dt} M_s(t) = -\frac{5\alpha F_p(t)}{\rho_s^{\frac{2}{3}}} \quad (4)$$

Integrating formula (4), we obtain (5).

$$\int_0^t M_s(t)^{-\frac{2}{3}} \frac{d}{dt} M_s(t) dt = -\frac{5\alpha}{\rho_s^{\frac{2}{3}}} \int_0^t F_p(t) dt$$

$$3M_s(t)^{\frac{1}{3}} = -\frac{5\alpha}{\rho_s^{\frac{2}{3}}} \int_0^t F_p(t) dt + 3M_s(0)^{\frac{1}{3}}$$

$$M_s(t) = \left(-\frac{5\alpha}{3\rho_s^{\frac{2}{3}}} \int_0^t F_p(t) dt + \sqrt[3]{M_s(0)} \right)^3 \quad (5)$$

The physical meaning of the equation (5) is only for the time $t \in [0, t_k]$, where t_k is the time of straw complete combustion. From the time t_k , the straw mass is equal zero. To determine t_k , we must solve the equation (6).

$$M_s(t_k) = 0 = \left(-\frac{5\alpha}{3\rho_s^{\frac{2}{3}}} \int_0^{t_k} F_p(t) dt + \sqrt[3]{M_s(0)} \right)^3 \quad (6)$$

$$-\frac{5\alpha}{3\rho_s^{\frac{2}{3}}} \int_0^{t_k} F_p(t) dt + \sqrt[3]{M_s(0)} = 0$$

$$\sqrt[3]{M_s(0)} = \frac{5\alpha}{3\rho_s^{\frac{2}{3}}} \int_0^{t_k} F_p(t) dt$$

$$\frac{3\rho_s^{\frac{2}{3}}}{5\alpha} \sqrt[3]{M_s(0)} = \int_0^{t_k} F_p(t) dt = \tilde{F}_p t_k \quad (7)$$

where \tilde{F}_p is the average value of the air mass flow on the time interval $t \in [0, t_k]$. Deriving t_k from the equation (7), we obtain the equation (8).

$$t_k = \frac{3\rho_s^{\frac{2}{3}}}{5\alpha \tilde{F}_p} \sqrt[3]{M_s(0)} \tag{8}$$

2.2 Straw combustion with a constant air mass flow

Assuming that $F_p(t) = F_p$, then statement (8) takes the form (9).

$$t_k = \frac{3\rho_s^{\frac{2}{3}}}{5\alpha F_p} \sqrt[3]{M_s(0)} \tag{9}$$

It is worth noting that the straw complete combustion occurs in a finite time.

2.3 Straw combustion with a constant speed

It is worth to consider a case of changing the flow $F_p(t)$ in such a manner that the straw combustion speed is maintained until complete combustion. Then, the thermal power supplied during combustion to the boiler is constant in time. Then the equation (1) takes the form (10).

$$D = -5\alpha F_p(t) \left(\frac{M_s(0) + Dt}{\rho_s} \right)^{\frac{2}{3}} \tag{10}$$

where D is a predetermined straw combustion speed (its value is negative because of the mass loss). Deriving $F_p(t)$ from the equation (10), we obtain the equation (11).

$$F_p(t) = \frac{D\rho_s^{\frac{2}{3}}}{-5\alpha(M_s(0) + Dt)^{\frac{2}{3}}} \tag{11}$$

The equation (11) can be expressed by $F_p(t)$ instead of D. For this purpose, in the equation (10), we substitute $t = 0$ and substitute derived D to the equation (11). Then we obtain the equation (12).

$$F_p(t) = \frac{F_p(0)M_s(0)^{\frac{2}{3}}}{\left[M_s(0) - 5\alpha F_p(0) \left(\frac{M_s(0)}{\rho_s} \right)^{\frac{2}{3}} t \right]^{\frac{2}{3}}} \tag{12}$$

The formula (13) presents the statement (12) in the integral form.

$$\int_0^t F_p(t) dt = \frac{3\rho_s^{\frac{2}{3}}}{-5\alpha} \left[\left(M_s(0) - 5\alpha F_p(0) \left(\frac{M_s(0)}{\rho_s} \right)^{\frac{2}{3}} t \right)^{\frac{1}{3}} - M_s(0)^{\frac{1}{3}} \right] \quad (13)$$

From the condition for the straw complete combustion, we obtain the condition (14) for the time of straw complete combustion.

$$t_k = -\frac{M_s(0)}{D} = \frac{\rho_s^{\frac{2}{3}}}{5\alpha F_p(0)} \sqrt[3]{M_s(0)} \quad (14)$$

An interesting observation is that $\int_0^{t_k} F_p(t) dt$ has a finite value, however, $F_p(t_k) \rightarrow \infty$. Comparing (14) with (8), we obtain an interesting conclusion that $\tilde{F}_p = 3F_p(0)$.

3 Thermal phenomena in the batch-fired straw boiler

Using the equations (1) and (3) one can build a model of thermal processes occurring in the analysed boiler. This model for the batch-fired straw boiler, can be described in the form of the ordinary differential equation system (15).

$$\left\{ \begin{array}{l} V\rho_{sp}c_{sp} \frac{d}{dt} T_{sp}(t) = P(t) + K_{s-w}(T_c(t) - T_{sp}(t)) + F_p c_{sp}(T_p(t) - T_{sp}(t)) \end{array} \right. \quad (15a)$$

$$\left\{ \begin{array}{l} M_w(t)c_w \frac{d}{dt} T_c(t) = F_w c_w (T_z(t) - T_c(t)) + K_{s-w}(T_{sp}(t) - T_c(t)) \end{array} \right. \quad (15b)$$

$$\left\{ \begin{array}{l} \frac{d}{dt} M_s(t) = -5\alpha \cdot F_p \left(\frac{M_s(t)}{\rho_s} \right)^{\frac{2}{3}} \end{array} \right. \quad (15c)$$

$$\left\{ \begin{array}{l} -\lambda \frac{d}{dt} M_s(t) = P(t) \end{array} \right. \quad (15d)$$

where:

t – time,

$M_s(t)$ – current straw mass,

$T_p(t)$ – ambient temperature,

$T_{sp}(t)$ – exhaust temperature,

$T_c(t)$ – hot water temperature (temperature of water flowing out of the water jacket),

$T_z(t)$ – cold water temperature (temperature of water flowing into the water jacket),

$\rho_{sp}(t)$ – exhaust density,

$P(t)$ – emerging thermal power during combustion process,

F_p – air mass flow,

F_w – water mass flow through the water jacket,

M_w – water jacket mass,

α – straw burning speed factor,

λ – straw heating value,

ρ_s – straw density,

V – combustion chamber volume,

c_{sp} – exhaust and air specific heat capacity,

c_w – water specific heat capacity,

K_{s-w} – factor related to heat transfer between combustion chamber and water jacket.

The equations (15a) and (15b) describe the balance of mass and thermal energy in the boiler respectively for the combustion chamber and the water jacket. Equation (15c) is the same as the equation (1), but for the purpose of further analysis there is assumed that the air mass flow F_p is constant in time. Equation (15d) is the same as the equation (3), which determines the emerging thermal energy during combustion process.

The mathematical model of the batch-fired straw boiler described by the equation system (15) has been implemented in MATLAB/Simulink as a simulation model.

Table 1. presents a few selected model parameters whose values are obtained by identification of the model based on experimental data of the hot and cold water temperature. As the burning straw was previously stored in a warehouse in the form of cubes, so the humidity on the edge could be significantly less than deep inside. Therefore, for the identification there were proposed two different straw burning speed factors α_1 and α_2 and their switching time t_c .

Figure 3. shows the time responses of the hot water temperature of model and experiment (red and blue charts) and time response of cold water temperature of experiment (green chart).

Table 1.

The identified parameters based on the model and experiment of combustion

No.	Parameter	Optimal value
1.	α_1	0.8065 [1/m ²]
2.	α_2	0.1612 [1/m ²]
3.	t_c	800.5959 [s]
4.	Performance index	800.5959

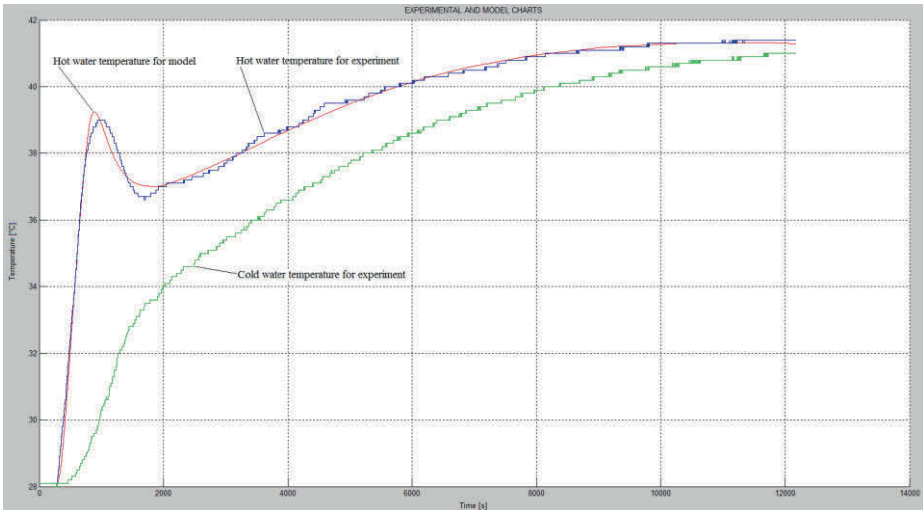


Fig. 3. The time responses of the hot water temperature of model and experiment (red and blue charts) and time response of cold water temperature of experiment (green chart). For practical reasons, the registration of the experimental data has been started before straw inflammation

4 Conclusions

In this paper there is presented a description of the straw combustion process in the batch-fired straw boiler apart from the chemical process phenomena. This description based only on the mass balance and the thermal energy balance, as from the practical point of view this approach seems to be sufficient. The proposed model of the combustion process in batch-fired straw boiler is from a mathematical point of view very interesting because of exponent $2/3$ in the expression of the combustion process model. Because of mentioned form of the equation describing the straw burning process, the straw complete combustion occurs in a finite time. This issue is due to the nonlinearity of the equation of the straw combustion process. The physical processes described by linear equations, for example a half-life process in nuclear physics give different results in time. In the latest example the time response asymptotically approaches zero, but not reaching it in a finite time.

The description of other phenomena of the straw boiler has been described with ordinary differential equations, because there was no need of describing these phenomena in a more complicated way.

Acknowledgements

I am greatly indebted to Professor Wojciech Grega and Professor Mariusz Filipowicz for their help in experimental part of this work. The financial support from the Ministry of Science and Higher Education is gratefully acknowledged.

References

- [1] J.R. Barker, M.H. Ross, An introduction to global warming, *American Journal of Physics* 67 (1999) 1216 – 1226
- [2] E. F. Guallart, U. Schuster, N. M. Fajar, O. Legge, P. Brown, C. Pelejero, M.-J. Messias, E. Calvo, A. Watson, A. F. Ríos, F. F. Pérez, Trends in anthropogenic CO₂ in water masses of the Subtropical North Atlantic Ocean, *Progress in Oceanography* 131 (2015) 21 – 32
- [3] E. Specht, T. Redemann, N. Lorenz, Simplified mathematical model for calculating global warming through anthropogenic CO₂, *International Journal of Thermal Sciences* 102 (2016) 1 - 8
- [4] United Nations, Kyoto Protocol to the United Nations Framework Convention on Climate Change, 1997
- [5] A. A. Bhuiyan, J. Naser, Thermal characterization of coal/straw combustion under air/oxy-fuel conditions in a swirl-stabilized furnace: A CFD modelling, *Applied Thermal Engineering* 93 (2016) 639–650
- [6] S. K. Kær, Straw combustion on slow-moving grates—a comparison of model predictions with experimental data, *Biomass and Bioenergy* 28 (2005) 307–320
- [7] B. Miljković, I. Pešenjanski, M. Vičević, Mathematical modelling of straw combustion in a moving bed combustor: A two dimensional approach, *Fuel* 104 (2013) 351–364
- [8] B. S. Repić, D. V. Dakić, A. M. Erić, D. M. Djurović, A. D. Marinković, S. Dj. Nemoda, Investigation of the cigar burner combustion system for baled biomass, *Biomass and Bioenergy* 58 (2013) 10 – 19
- [9] Y. B. Yang, R. Newman, V. Sharifi, J. Swithenbank, J. Ariss, Mathematical modelling of straw combustion in a 38 MWe power plant furnace and effect of operating conditions, *Fuel* 86 (2007) 129–142
- [10] H. Zhou, A. D. Jensen, P. Glarborg, P. A. Jensen, A. Kavaliauskas, Numerical modeling of straw combustion in a fixed bed, *Fuel* 84 (2005) 389–403

Modelling, Simulation and Optimization of the Wavemaker in a Towing Tank

Marcin Drzewiecki

Gdansk University of Technology, Faculty of Electrical and Control Engineering

marcin.drzewiecki@pg.gda.pl

<http://eia.pg.edu.pl/>

CTO S.A. Ship Design and Research Centre, Szczecinska 65, Gdansk, Poland

marcin.drzewiecki@cto.gda.pl

<http://cto.gda.pl>

Abstract. The paper analyses the problem of experimental identification (frequency response), modelling and optimization of the towing tank wavemaker in the Scilab/Xcos environment.

The experimental identification of the objects (the towing tank wavemaker placed in the hydrodynamic laboratory of the CTO S.A. Ship Design and Research Centre (CTO)) and the implementation of the models in the simulation environment, enable to perform:

1. tuning of the cascade PID controller (Astrom-Hagglund relay method),
2. checking the stability (Routh-Hurwitz criterion, Nyquist criterion) of the wavemaker with tuned controller,
3. evaluating the regulatory quality and simulating of work of the optimized (Ziegler-Nichols method) system of the wavemaker.

The above works and the achieved results are further described and presented in the content of this paper.

Keywords: towing tank; cascading PID controllers; modelling, simulation and optimization.

1 Introduction

During the tests with a ship model on a sea wave, where marine conditions are modeled in the towing tank, it is important to obtain a good realization of the designed wave.

The CTO towing tank is an object with 270 m length, 12 m width and 6 m depth. The waves in the towing tank are generated by the wavemaker with a rigid flap with single articulation above channel bed, as shown in Fig. 1. The rigid flap is moved by a hydraulic cylinder (controlled in outer loop of the cascading PID), which is driven by an electrohydraulic servo valve (controlled in inner loop of the cascading PID), as it shown in Fig. 2. The anti-windup loop has not been implemented because the variables do not reach the threshold actuators (external safety system protects from excessive opening of the valve or overswing of the rigid flap).

The relationship between the range (peak to peak) of the paddle moves e and wave height (peak to peak) HW , for the mentioned type of the wavemaker is given by the Biesel's transfer function (BTF) (1) [1], where water depth h and height of the articulation of the rigid flap above the channel bed h_0 are consistent with Fig. 1. The k is the hydrodynamic constant calculated for deep water using (2) where f is a frequency of the wave and g is the gravitational acceleration. The BTF was validated for the CTO towing tank and results of the measurements are shown in Fig. 3.

$$BTF = \frac{HW}{e} = \frac{2}{k(h - h_0)} \left(\frac{\sinh kh((h - h_0)k \sinh kh - \cosh kh + \cosh kh_0)}{\sinh kh \cosh kh + kh} \right) \tag{1}$$

$$k = \frac{(2\pi f)^2}{g} \tag{2}$$

Due to validated (1), the proper control of paddle moves e (using the cascading PID controllers) is crucial for obtaining the expected waves height HW in the CTO towing tank.

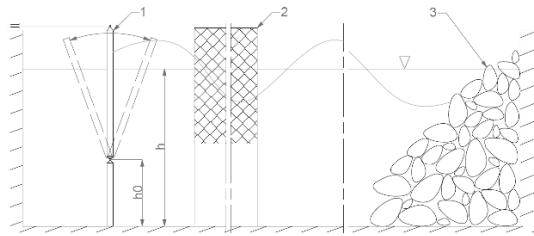


Fig. 1. Longitudinal profile of the deepwater towing tank with the wavemaker rigid flap 1, waveguides 2 for better propagation of the wave and an artificial beach 3 for dumping the waves.

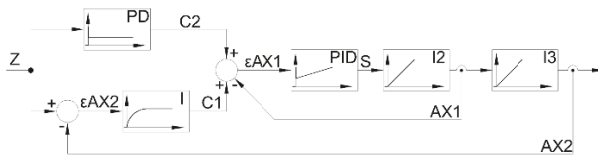


Fig. 2. The structure of cascading PID controllers: a slave-PID controller of the electrohydraulic servo valve I2 and a master-PID controller of the position of the rigid flap of the wavemaker, driven by hydraulic cylinder I3.

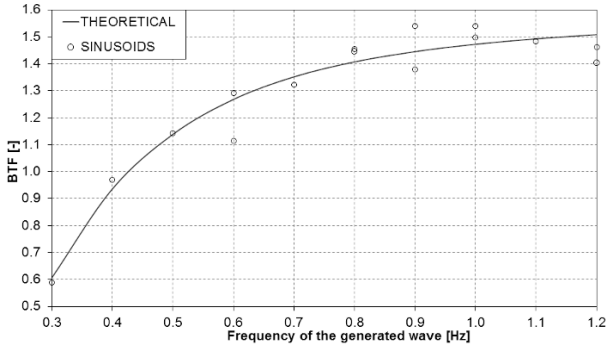


Fig. 3. The BTF for the CTO towing tank - theoretical, calculated using (1) and measured for generated sinusoidal waves.

2 Objective and Scope

Hitherto the cascading PID controllers (after modernization in 2015 [2]) were working under the parameters shown in Tab. 1. These parameters were obtained by the step-response (Fig. 4) of the analog control system, designed in 1974.

Table 1. Precedent parameters of the PID controllers

Module	Param.	Value
I	K_i	0.48
	T_i	0.29 s
PD	K_p	0.35
	K_d	0.63
	T_d	0.92 s
PID	K_p	0.61
	K_i	0.45
	T_i	0.21 s

In response to growing customer expectations, it was necessary to obtain better regulatory quality. To achieve this goal, the identification of the actuators (the servo valve and the hydraulic cylinder), using the frequency response method, was made. Identified actuators were implemented to the simulation environment (Xcos/Scilab) and tuning of the cascading PID controllers, using Astrom-Hagglund relay method, was done.

For tuned controllers, the stability was checked: analytically, by the Routh-Hurwitz method and by simulation, using the Nyquist method. For the validated control system, the better regulatory quality was proven.

The optimized model of the towing tank with the wavemaker for further simulation testing, was made.

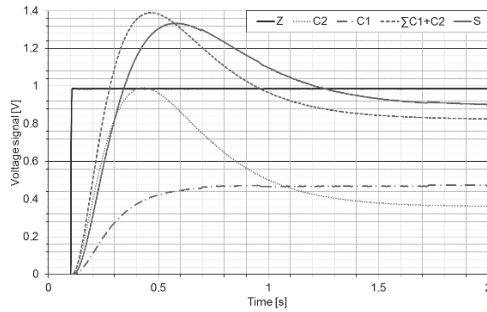


Fig. 4. The step-response characteristics of the analog control system under the precedent parameters (symbols are in accordance with Fig. 2).

3 Solution

3.1 Identification

Looking at the Fig. 2: sinusoidal signals with the amplitude equal to 1 V and with the frequency span from 0.3 Hz to 1.2 Hz (with 0.1 Hz resolution), were given as the S . For each point, the input signal S , the servo valve position signal $AX1$ and the rigid flap position signal $AX2$, was measured and recorded. Sampling frequency was equal to 25 Hz. A measuring amplifier HBM Spider8 4.8 kHz and a measurement software HBM Catman Professional 4.5 were used.

On the basis of the measurements, identification of the two actuators: the electrohydraulic servo valve and the hydraulic cylinder, was made.

Servo valve. On the basis of the measured S and $AX1$, the Nyquist plot for the servo valve was determined and shown in Fig. 5.

The servo valve was taken as an integrator with S at the input and $AX1$ at the output with linear relationship in the work area. For such assumptions, the parameter of the transfer function (3) was obtained as a mean value from values calculated for each point of the plot in Fig. 5.

$$G_1(s) = \frac{1}{Tc1 \cdot s} = \frac{10.15}{s} \quad (3)$$

Hydraulic cylinder. On the basis of the measured $AX1$ and $AX2$, the Nyquist plot for the hydraulic cylinder was determined and shown in Fig. 6.

The hydraulic cylinder was taken as an integrator with $AX1$ at the input and $AX2$ at the output with linear relationship in the work area. The inertia of the rigid flap was negligible and was omitted. For such assumptions, the transfer function (4) was obtained as a mean value from values calculated for each point of the plot in Fig. 6.

$$G_2(s) = \frac{1}{Tc2 \cdot s} = \frac{2.47}{s} \quad (4)$$

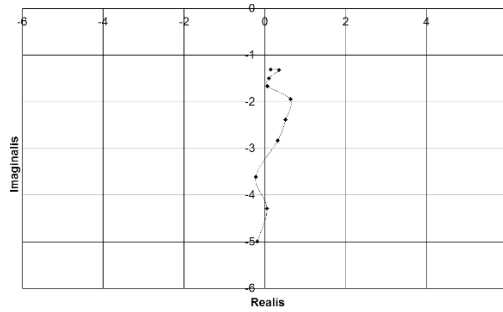


Fig. 5. The Nyquist plot for the electrohydraulic servo valve.

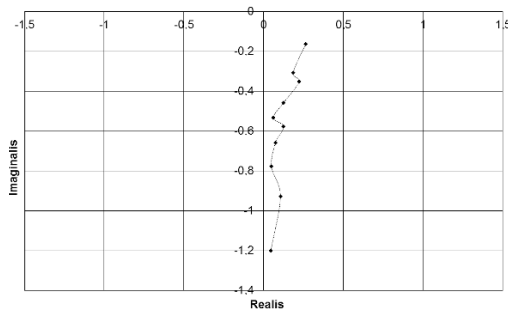


Fig. 6. The Nyquist plot for the hydraulic cylinder.

3.2 Tuning

The identified actuators were implemented to the simulation environment (Xcos/Scilab).

For simulated cascading control loops, the tuning of the slave-PID, works in the inner loop and the tuning of the master-PID, works in the outer loop, respectively, were done. The Astrom-Hagglund relay method and Ziegler-Nichols parameters [3], were used.

Slave-PID. The servo valve, identified as (3), was implemented to the tuning loop model, shown in the Fig. 7.

Based on the simulation characteristics of the model, the ultimate gain K_u and the ultimate period T_u , were obtained (Tab. 2). Basing on the ultimate parameters, the optimized (in sense of the Ziegler-Nichols method) parameters of the slave-PID controller were calculated and shown in the Tab. 2.

The structure of the inner loop with the optimized slave-PID controller is shown in Fig. 9.

Master-PID. The hydraulic cylinder, identified as (4), was implemented to the tuning loop model, shown in the Fig. 9.

On the basis of the simulation characteristics of the model, the ultimate gain K_u and the ultimate period T_u , were obtained (Tab. 2). Basing on the ultimate parameters, the optimized (in sense of the Ziegler-Nichols method) parameters of the slave-PID controller were calculated and shown in the Tab. 2.

The structure of the outer loop with the optimized master-PID controller is shown in Fig. 10.

Table 2. Optimized parameters of the PID controllers, based on K_u and T_u .

Param.	Slave-PID	Master-PID
K_u	1.277	1.214
T_u	0.394 s	1.557 s
K_p	0.766	0.728
T_i	0.197 s	0.779 s
T_d	0.0473 s	0.187 s

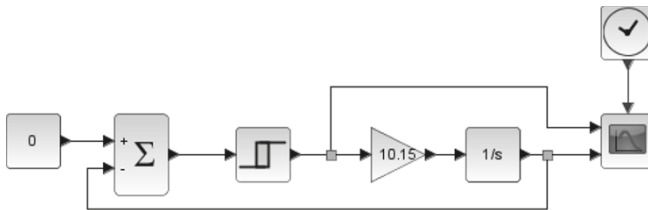


Fig. 7. The slave-PID tuning loop of the Astrom-Hagglund relay method.

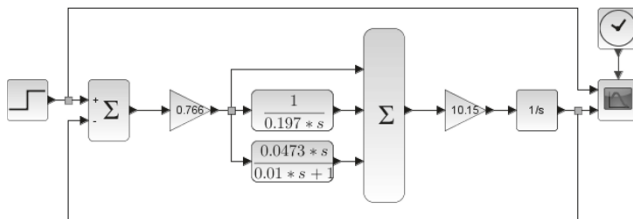


Fig. 8. The structure of the inner loop with the optimized slave-PID controller.

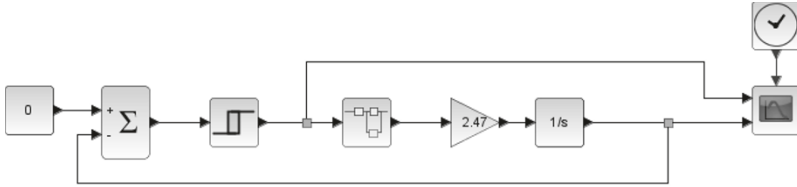


Fig. 9. The master-PID tuning loop of the Astrom-Hagglund relay method.

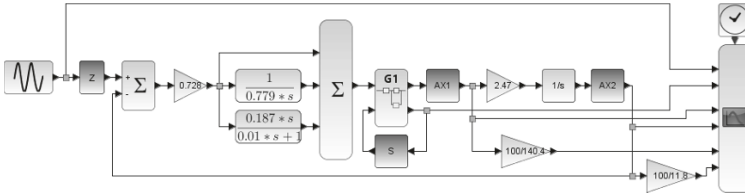


Fig. 10. The structure of the outer loop with the optimized master-PID controller.

3.3 Stability

It was necessary to confirm the stability of the designed control system.

The stability was tested in the analytical way under the Routh-Hurwitz criterion and in the simulation way under the Nyquist criterion.

Routh-Hurwitz criterion. The criterion for inner loop was confirmed. The Hurwitz matrix (5) was constructed on basis of the structure in Fig. 8 and calculated using the parameters in (3) and Tab. 2. The determinant of the (5) will always be greater than zero. It means that system is structurally stable.

The criterion for outer loop was confirmed. The Hurwitz matrix (6) was constructed on basis of the structure in Fig. 10 and calculated using the parameters in (3) and Tab. 2. The determinant of the (6) will be greater than zero for parameters in Tab. 3. It means the system is conditionally stable.

$$H1 = \begin{bmatrix} Kp1 \cdot Ti1 & 0 \\ Ti1 \cdot (Tc1 + Kp1 \cdot Td1) & Kp1 \end{bmatrix} \tag{5}$$

$$H2 = \begin{bmatrix} s3 & s1 & 0 & 0 \\ s4 & s2 & s0 & 0 \\ 0 & s3 & s1 & 0 \\ 0 & s4 & s2 & s0 \end{bmatrix} \tag{6}$$

where:

$$s4 = Kp1 \cdot Ti1 \cdot Td1 \cdot Kp2 \cdot Ti2 \cdot Td2$$

$$s3 = Ti1 \cdot Ti2 \cdot (Kp1 \cdot Td1 + Tc1 + Kp1 \cdot Td1 \cdot Kp2 + Kp1 \cdot Kp2 \cdot Td2)$$

$$s2 = Kp1 \cdot (Ti1 \cdot Ti2 + Ti1 \cdot Td1 \cdot Kp2 + Ti1 \cdot Kp2 \cdot Ti2 + Kp2 \cdot Ti2 \cdot Td2)$$

$$s1 = Kp1 \cdot (Ti2 + Ti1 \cdot Kp2 + Kp2 \cdot Ti2)$$

$$s0 = Kp1 \cdot Kp2$$

Nyquist criterion. The implemented model of the wavemaker with the designed control system (Fig. 10) was tested under the Nyquist criterion. Using an Xcos/Scilab instant tools, the Nyquist characteristic was plotted (for Z as the input and $AX2$ as the output signal) as shown in Fig. 11. The critical point $(-1.0, 0.0)$, marked in Fig. 11, is always on the left side of the characteristic rising from the low frequency equal to 0.1 Hz to the high frequency, equal to 2.0 Hz. It means that tested close loop system (the wavemaker with the cascading PID regulators) is stable.

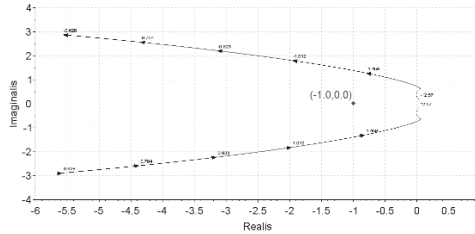


Fig. 11. The Nyquist characteristic of the optimized wavemaker.

3.4 Regulatory Quality

For validation of the designed stable system, the regulatory quality was examined by simulation.

The comparison of the regulatory quality of the optimized PID controllers (with structure shown in Fig. 10, under parameters as in Tab. 2) with the precedent PID controllers (with structure shown in Fig. 2, under parameters as in Tab. 1) for the working system were done.

The simulation of step-response for the system of wavemaker working with the both variants were done. The step-response of the optimized system is shown in Fig. 12. The step-response of the precedent system is shown in Fig. 13. The parameters of the regulatory quality (RQP) of the both systems are juxtaposed in Tab. 3. As it is shown, for the optimized control system: the time setting and the rise time was two times longer, the oscillation appeared slightly, the overshoot was seven times smaller, the stable setpoint was reached.

3.5 Optimized model

Using the BTF (1) and optimized (in sense of the Ziegler-Nichols method) model of the wavemaker (Fig. 10), the model of the whole object (the towing tank with the wavemaker) was carried out.

It allowed to carry out simulation testing of the optimized object under the setpoint waves. The example of the simulation of working wavemaker in the towing tank is shown in Fig. 14, where form of the basin wave ($f=1.0$ Hz) is shown as HW .

Table 3. Juxtaposition of the regulatory quality parameters (RQP) for the precedent and for the optimized (in sense of the Z-N method) cascading PID control system.

RQP	Optimized	Precedent
Time setting	0.875 s	0.494 s
Rise time	0.674 s	0.333 s
Time adjustment	4.608 s	∞
Overshoot	0.211	1.548
Oscillating	0.139	0

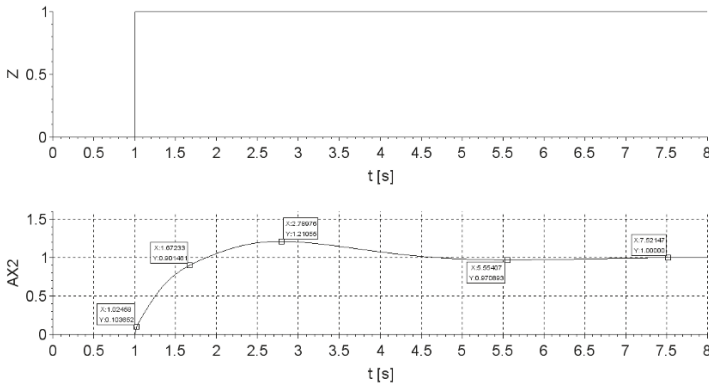


Fig. 12. The step-response of the system with optimized cascading PID controllers.

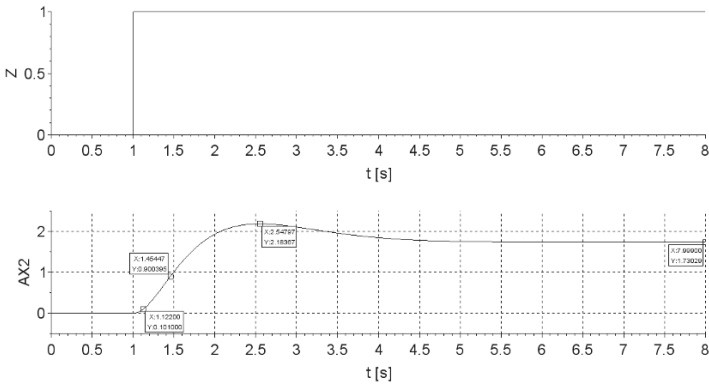


Fig. 13. The step-response of the system with precedent cascading PID controllers.

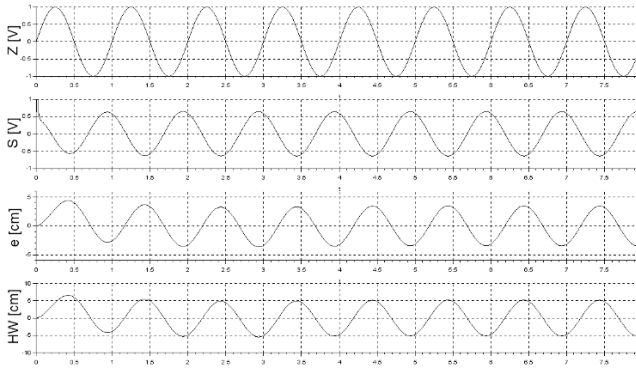


Fig. 14. The simulation of the towing tank with the working optimized wavemaker.

4 Conclusion

In the scope of the works, the new cascading control system was designed for the identified actuators of the wavemaker.

The system was optimized using the Astrom-Hagglund relay method and the Ziegler-Nichols method. The optimized system was examined under the Routh-Hurwitz and under the Nyquist criterion. The regulatory quality was examined. The designed, optimized and validated control system has the capacity to reach the stable setpoint (in contrast to precedence) and the smaller (than precedence) overshoot. The rest of the RQP are satisfactory.

The work done in the simulation environment allowed to implement the optimized cascading PID controllers of the wavemaker. It enables to obtain better realisation of the wave which meets the new expectations of the CTO customers.

The important advantage lies in the use of the optimized model of the towing tank with wavemaker in place of the real object. It will be used to search more advanced methods to control waves (directly or nondirectly) in the simulation environment, without costly use of the towing tank.

Acknowledgments. The research was financed by the budget of the Ministry of Science and Higher Education of the Republic of Poland, earmarked for statutory activity of CTO S.A. Ship Design and Research Centre.

References

1. Biesel, F.Suquet, F.: Les appareils generateurs de houle en laboratoire. La Houille Blanche. 161-163 (1951).
2. Drzewiecki, M.: Modernizacja kaskadowego ukladu regulacji wywoływacza fal basenu modelowego. Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdanskiej. 39-42 (2015).
3. Cheng-ching, Y.: Autotuning of PID controllers Relay Feedback Approach. Springer-Verlag Berlin Heidelberg, New York (1999).

Modeling dynamical systems using neural networks and random linear projections

Ewa Skubalska-Rafajłowicz¹

Wrocław University of Science and Technology
Faculty of Electronics, Department of Computer Engineering
ewa.rafajlowicz@pwr.edu.pl

Abstract. We focus our attention on two stable models of nonlinear dynamic systems (external dynamic approach): NFIR (nonlinear finite impulse response) and simple version of NARX (nonlinear autoregressive model with external inputs), and their linear counterparts. The main idea investigated in the paper is to project the vector of past inputs u_n 's onto random directions drawn uniformly from the unit sphere, (instead of estimated) and select only those projections that are relevant for proper neural networks based models.

Keywords: dynamical system modeling, neural networks, random projections

1 Introduction

Our aim in this paper is to propose and investigate a new approach to modeling dynamical systems by neural networks and linear systems focusing on neural networks with random projections. Other approaches to modeling dynamical systems without using random projections are well known (see, e.g., [5], [6], [9], [10] among many others).

We start with a time-invariant continuous time non-autonomous dynamical system with exogenous (external) inputs, i.e.,

$$x'(t) = F(x(t), u(t)), \quad y(t) = H(x(t)), \quad x \in R^D, \quad u \in R^n, \quad y \in R,$$

or discrete time system:

$$x_{n+1} = f(x_n, u_n), \quad y_n = h(x_n).$$

We assume that observations are corrupted by noise (usually white):

$$y_n = h(x_n) + \epsilon_n.$$

This noise may be treated as a part of the system. In the noise-free framework observations $y_k = h(x_k)$ (or $y(t) = H(x(t))$) can be measured precisely.

Modeling of nonlinear dynamical systems is performed in two modes. First, in the simulation mode when only inputs of the system are available. Second, in the prediction mode when also last measurements y_k, \dots, y_{k-l} are given.

Neural networks are (general) nonlinear black-box structures [5], [9], [10].

Classical neural network architectures (feed-forward structures) are multi-layer perceptrons (MLP) with one (or more) hidden layers and sigmoid activation functions. There are two types of neural networks used for modeling of dynamical systems. Networks with lateral and/or feedback connections, such as for example Hopfield networks (associative memory) or Ellman nets (context sensitive) are artificial dynamical system themselves. Another type are feed-forward neural networks with external dynamics [9].

We focus our attention on these two stable models known as external dynamic approach:

1. NFIR (nonlinear finite impulse response)

$$y_{k+1}^{\hat{}} = g(u_k, \dots, u_{k-m}),$$

where $g : R^{(m+1)n} \rightarrow R$ is a continuous (and smooth) function.

2. NARX (nonlinear autoregressive model with external inputs)

$$y_{k+1}^{\hat{}} = G(y_k, \dots, y_{k-l}, u_k, \dots, u_{k-m}),$$

where $G : R^{(m+1)n+l+1} \rightarrow R$ is a continuous (and smooth) function.

G or g are represented in the system model by a MLP network, which is a real valued function on R^d

$$g_M(\theta, w; x) = v_0 + \sum_{j=1}^M \theta_j \Psi(\langle w_j, x \rangle), x \in \mathcal{K} \subset R^d,$$

where \mathcal{K} is a compact set. Activation function $\Psi(t)$ is a sigmoid (S-shape) function. The logistic function $\frac{1}{1+\exp(-t)}$ or hyperbolic tangent functions $\tanh(t) = \frac{1-\exp(-2t)}{1+\exp(-2t)}$ are usually used as activation functions in MLP. Neural networks are usually non-linear in the parameters.

If we allow a net to grow with the number of observations, then they are sufficiently rich to approximate smooth functions [3]. In practice, finite network structures are considered, leading to a parametric (weights) optimizing approach. Learning (training) is a selecting of weights, on the basis of examples (input-output pairs), using usually nonlinear LMS. The Levenberg-Marquardt local optimization algorithm is the most popular choice for MLP training. Nevertheless, some simplifications may lead to linear-in the parameter network structures.

Having a learning sequence (x_n, y_n) , $n = 1, 2, \dots, N$, the weights $\bar{w}, \bar{\theta}$ are usually selected by minimization of

$$\sum_{n=1}^N [y_n - g_M(\bar{w}, \bar{\theta}; x_n)]^2 \tag{1}$$

w.r.t. $\bar{w}, \bar{\theta}$.

This is frequently complicated - mostly not due to spurious local minima, but rather due to the existence of many saddle points.

Our idea is to replace \bar{w} in

$$g_M(x; \bar{w}, \bar{\theta}) = \theta_0 + \sum_{j=1}^M \theta_j \varphi(\langle w_j, x \rangle), \quad (2)$$

by randomly selected vectors \bar{s}_j 's, say, and replace x by past inputs

$$u_n, u_{n-1}, \dots, u_{n-r}$$

and possibly by past outputs $y_n, y_{n-1}, \dots, y_{n-r}$, which converts (2) into dynamic models with outer dynamics.

In the next section some motivation of using random projections are given. Next, NFIR and NARX models with projections are discussed. The main results of simulations for the Lorenz system are presented in Section 4.

2 Motivations for using random projections

Random projections [4], [8], [12], in signal processing termed sometimes as sketching, are widely considered to be one of the most potential methods of dimensionality reduction.

In the random projection method, the original high-dimensional observations are projected onto a lower-dimensional space using a suitably scaled random matrix with independent, typically, normally distributed entries.

Random projections are closely related to the Johnson-Lindenstrauss lemma [4], which states that any set A , say, of N points in an Euclidean space can be embedded in an Euclidean space of lower dimension ($\sim O(\log N)$) with relatively small distortion of the distances between any pair of points from A .

The idea of dimensionality reduction using random projections as a first layer of neural networks was proposed by Arriaga and Vempala in [1] and developed by the author in the context of the multi-layer feed-forward with sigmoidal activation functions [11].

In this section, we will consider the case of the well known, simple finite impulse response (FIR) model:

$$\hat{y}_n = \sum_{j=1}^J \alpha_j u_{n-j} + \epsilon_n, \quad n = 1, 2, \dots, N, \quad (3)$$

where y_n are outputs observed with the noise ϵ_n 's, while u_n 's form an input signal, which is observed (or even designed) in order to estimate α_j 's. Let us suppose that our system has a long memory - needs $J \approx 10^3$ for adequate modeling, e.g., chaotic systems. Is it reasonable to estimate $\sim 10^3$ parameters, even if the number of observations N is very large ?

The alternative idea is to project vector of past u_n 's onto random directions and select only those projections that are relevant for a proper modeling.

This idea allows us to reduce the dimensionality of a model and simultaneously approximately retaining geometry structure of the data forming inputs to the model.

3 Models NFIR and NARX with random projections

For T denoting the transposition, define:

$$\bar{u}_n = [u_{n-1}, u_{n-2}, \dots, u_{(n-r)}]^T.$$

Above, $r \geq 1$ is treated as large (hundreds or more) since we discuss models with long memory.

For $n=(r+1), (r+2), \dots, N$, we obtain

$$\hat{y}_n = g[(\bar{s}_1^T \bar{u}_n), \dots, (\bar{s}_K^T \bar{u}_n)] + \epsilon_n, \tag{4}$$

or if g is an MLP network with one hidden layer:

$$\hat{y}_n = \sum_{k=1}^K \theta_k \varphi(\underbrace{\bar{s}_k^T \bar{u}_n}_{int.proj.}) + \epsilon_n. \tag{5}$$

ϵ_n 's are i.i.d. random errors, having distribution with zero mean and finite variance; $E(\epsilon_n) = 0, \sigma^2 = E(\epsilon_n)^2 < \infty$.

When observations are properly scaled, we can use $\varphi(t) = t$ obtaining a simple linear model with random projections.

Random projection vectors $\bar{s}_k = s_k / \|s_k\|$ are normalized r -dimensional random normal vectors $s_k \sim \mathcal{N}(0, I_r)$. Clearly, s_k 's are mutually independent from ϵ_n 's.

It should be noted that however large K is, it is smaller than the input of model dimension, i.e., $K \ll r = dim(\bar{u}_n)$.

The NFIR model (5) looks similar to the projection pursuit regression (PPR), but there are some important differences. First, directions of projections \bar{s}_k 's are drawn at random uniformly from the unit sphere, instead of being estimated. Second, φ is given in the sense that it is not estimated from the data.

Model NARX is here seen as an extension of the NFIR model. It contains also an autoregressive (AR) part based on previous outputs of the dynamic system. These outputs could be also randomly projected. In this paper we assume that the number of output regressors R is small and they are not projected.

$$\hat{y}_n = G[(\bar{s}_1^T \bar{u}_n), \dots, (\bar{s}_K^T \bar{u}_n), y_{n-1}, y_{n-2}, \dots, y_{n-R}] + \epsilon_n. \tag{6}$$

4 Simulation study - chaotic Lorenz system perturbed by PRBS

Now, let us consider the well known chaotic Lorenz system [7], [2] perturbed by (interpolated) pseudo-random binary sequence (PRBS) $u(t) \in \{-1, 1\}$:

$$\dot{x}(t) = 100 u(t) - 5(x(t) - y(t)) \tag{7}$$

$$\dot{y}(t) = x(t) (-z(t) + 26.5) - y(t) \tag{8}$$

$$\dot{z}(t) = x(t)y(t) - z(t) \tag{9}$$

with IC $x(0) = z(0) = 0, y(0) = 1$. We want to design the system behaviour model in order to predict:

A) to $x(t_n)$ from $\chi_n = x(t_n) + \epsilon_n$,

B) $y(t_n)$ from $\eta_n = x(t_n) + \epsilon_n$,

without using the knowledge about (7)-(9). The system behavior in the form of phase plot is depicted in Fig. 1. We have assumed that output $x(t)$ (or $y(t)$) is

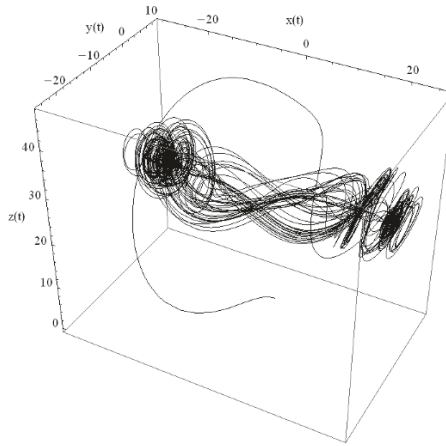


Fig. 1. Phase plot of the Lorenz system perturbed by PRBS

measured with additional noise $\mathcal{N}(0, 0.1)$ and the signal is sampled with $\tau = 0.01$.

5.1 Prediction of $x(t)$ coordinate of the Lorenz system using FIR model with random projections.

The estimated model for $x(t)$ simulation is of the form

$$x_n = \sum_{k=1}^K \theta_k (\bar{s}_k^T \bar{u}_n) + \epsilon_n, \tag{10}$$

where the number of random projections is $K = 150$ based on $r = 2000$ of past inputs ($r = \dim(\bar{u}_n)$) that are projected by \bar{s}_k , where \bar{s}_k is $s_k \sim \mathcal{N}(0, I_r)$ – normalized to 1.

Estimated model output vs learning data is shown in Fig. 2 (left panel). MSE error for training is 0.433. One step ahead prediction simulated using the estimated model, i.e., output vs testing data (4000 predictions are recorded) is presented in Fig. 2 (right panel).

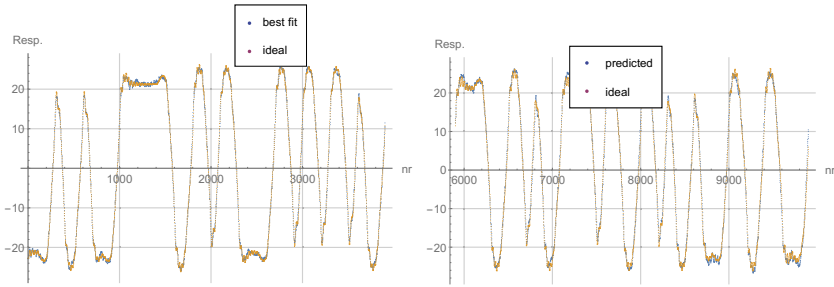


Fig. 2. Left panel) $x(t)$ component of the Lorenz system estimated by the FIR-model with random projections – a learning sequence. Right panel) $x(t)$ component of the Lorenz system predicted by the FIR-model with random projections – the whole testing sequence.

We have obtained a very accurate prediction using a relatively simple model and information about inputs only. MSE error for testing (averaged over 4000 samples) is 1.489. It is clear that the Lorenz system’s $x(t)$ component is relatively easy to predict when the is system perturbed by PRBS. It does not behave as a chaotic signal.

5.2 Prediction of $y(t)$ coordinate of the Lorenz system using a FIR model with random projections.

Let’s consider a real challenge: prediction of the second, i.e., $y(t)$ component. The $y(t)$ coordinate signal is really chaotic as one can see in Fig. 3. It shows the part of the signal used for learning the system model. The true output signal without additional noise looks very similar.

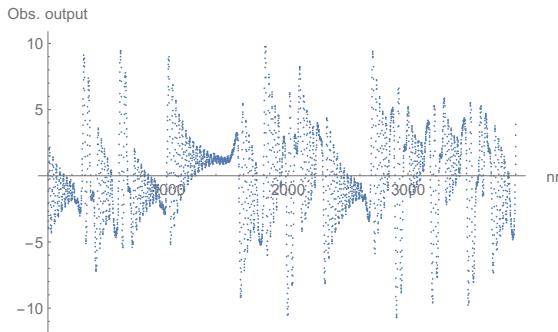


Fig. 3. The Lorenz system perturbed by PRBS- noisy $y(t)$ component – a part used for learning.

As in the case of predicting $x(t)$ component we have used the same model structure but with different parameters. Namely, $r = 500$ – the number of past

inputs ($r = \dim(\bar{u}_n)$) that are projected by $K = 100$ random projections \bar{s}_k where \bar{s}_k is $s_k \sim \mathcal{N}(0, I_r)$ – normalized to 1.

We have used the same input signals as in case A). 8000 noisy observations of $y(t)$ component was generated. ~ 4000 was used for learning and ~ 4000 for testing prediction abilities. The observations started at the same time as in the case of predicting $x(t)$ output signal, but a smaller number of previous PRBS's inputs was used, i.e., only 500, not 2000.

Fig. 4 (left panel) provides a comparison of the model FIR response vs learning data. The first 1000 observations of estimated $y(t)$ coordinate are visualized. The fit for training data is not perfect, but retains almost all oscillations. MSE obtained for the whole training sequence is equal to 3.96. Figure 4 (right panel) presents the first 1000 results of prediction $y(t)$ coordinate based on the last 1000 projected input signals. MSE obtained for the whole testing sequence is equal to 6.63. The prediction is far from precise, but still retains a general shape of the highly chaotic sequence. It should be stressed that each prediction is made on the basis of the last 1000 input values, not taking into account previous output measurements.

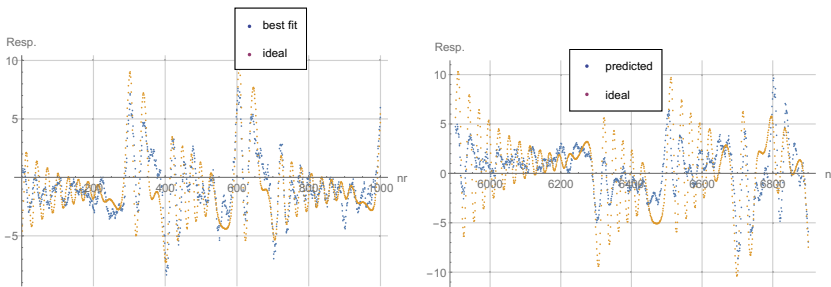


Fig. 4. Left panel) $y(t)$ component of the Lorenz system estimated by the FIR-model with random projections – first 1000 observations from the learning sequence. Right panel) $y(t)$ component of the Lorenz system predicted by the FIR-model with random projections – first 1000 observations taken from the testing sequence.

5.3 Prediction of $y(t)$ using neural network model NFIR with random projections.

The same experiment with respect to $y(t)$ coordinate prediction we have performed using MLP neural network with two hidden layers as a nonlinear function of $r = 500$ inputs signal projected onto $K = 100$ dimensional space. The computations were done using Mathematica 11.0 and structure of the network designed by the system is as follows: 100 input nodes, two hidden layers of 24 neurons each, one linear output neuron. All neuron's activations functions were $\tanh(\cdot)$. L_2 regularization factor equals 0.1.

The results for the whole learning data are shown on Fig. 5 (left panel). Figure 5 (right panel) presents results of prediction $y(t)$ coordinate based on the last

(with respect to the moment of prediction) 1000 projected input signals. MSE

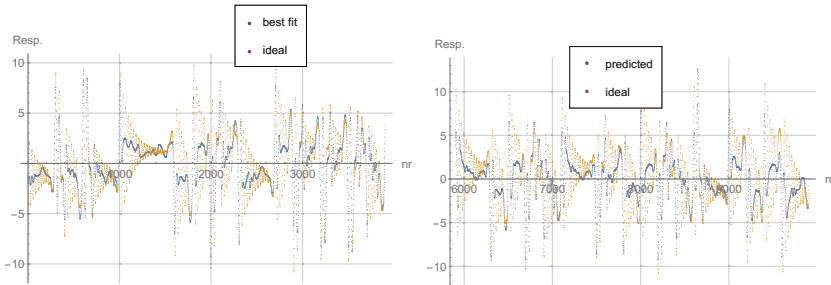


Fig. 5. Left panel) The Lorenz system perturbed by PRBS- model response for $y(t)$ coordinate vs learning data (neural NFIR-model with random projections). Right panel) $y(t)$ component of the Lorenz system predicted by the neuronal NFIR-model with random projections – the testing sequence.

for the training and testing sequence was equal to 2.96 and 6.84, respectively.

5.4 Prediction of $y(t)$ coordinate of the Lorenz system using the ARX and neural network based NARX model with random projections of inputs.

This time we have utilized the simplified version of the ARX model with only two previous output measurements, i.e., $y(n - 1)$ and $y(n - 2)$. The model has the form (linear model):

$$\hat{y}_n = \sum_{k=1}^K \theta_k (\bar{s}_k^T \bar{u}_n) + \theta_{K+1} y_{n-1} + \theta_{K+2} y_{n-2} + \epsilon_n, \tag{11}$$

and

$$\hat{y}_n = G[(\bar{s}_1^T \bar{u}_n), \dots, (\bar{s}_K^T \bar{u}_n), y_{n-1}, y_{n-2}] + \epsilon_n. \tag{12}$$

For clarity of presentation we have assumed that $r = 500$ and $K = 98$. In such a way the linear model has as previously 100 parameters and the neural networks model 100 input nodes and 27 nodes in each hidden layer. The learning and testing data were also the same. Linear model (11) provided very accurate one-step ahead predictions of $y(t)$ coordinate with MSE equal to 0.061 (for testing). In the case of neural NARX (12)s predictions were slightly less accurate with MSE of one-step ahead predictions on test data equal to 0.79. MSE for learning were 0.054 and 0.41, respectively. Fig. 6 shows one step ahead prediction of $y(t)$ component of the Lorenz system. Predictions for linear model (11) are shown on the left. Predictions for neural network model (12) are shown on the right.

Finally, we have used the previous models for simulation. This means that the lastly predicted output values are plug-in into the model (11) or (12) instead of output measurements. Such models are known as Output Error (OE) models (see for example [9]). The subsequent predictions on the previous (here the last two)

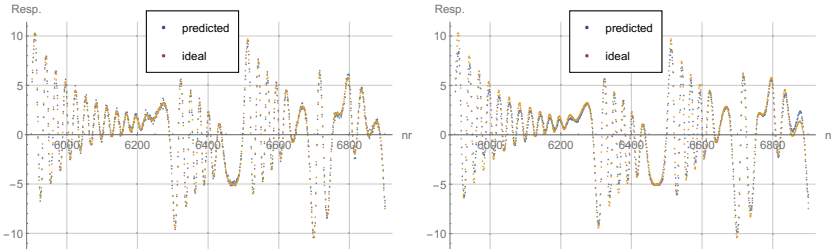


Fig. 6. One step ahead prediction of $y(t)$ component of the Lorenz system. On the left – predictions for linear model (11). On the right – predictions for neural network model (12).

predictions and projected system's inputs (as in FIR and NFIR) form input data for the models. Thus, OE models work in the simulation mode. Fig. 7 presents 1000 first subsequent predictions of $y(t)$ in comparison to the test measurements. The left panel shows results for the linear OE model and the right panel shows corresponding predictions obtained by neural NOE. MSE for OE was equal to

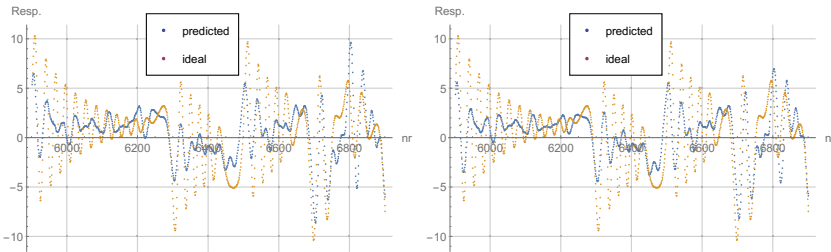


Fig. 7. simulations of $y(t)$ component of the Lorenz system. On the left – predictions for linear model (11). On the right – predictions for neural network model (12).

7.37. MSE for NOE was very similar and equals 7.32. It is easy to check that in both cases model outputs are close to the true measurements and only fast changing impulse-like signals are underestimated.

5 Final remarks and conclusions

Projections of a long sequence of past inputs plus LSQ provide an easy-to-use method of predicting sequences having complicated behaviors. In our simulations linear models are slightly better as an adequate nonlinear neural networks models, but it should not be a rule. It only indicates that even in the case of highly non-linear dynamic systems linear models with random projections should also be taken into account. The choice of $r = \dim(\bar{u}_n)$ is definitely important. Here

it was done by trial and error, but one can expect that Akaike's or Rissanen's criteria will be useful. The selection of K – the number of projections is less crucial. One can project the vector of past inputs u_n 's (and possibly y_n) onto a larger number of random directions and select only those projections that are relevant for proper modeling. One can also consider a mixture of projections having different lengths as a more robust approach.

References

1. Arriaga, R., Vempala S.: An algorithmic theory of learning: Robust concepts and random projection, *Proc. of FOCS, New York, 1999*: 616-623, 1999.
2. Haykin, S.(Eds): Kalman Filtering and Neural Networks, John Wiley and Sons, Inc., New York, 2001.
3. Haykin, S.: Neural networks and learning machines, 3rd ed., Pearson Education, Inc., Upper Saddle River, New Jersey, 2009.
4. Johnson, W. B., Lindenstrauss, J.: Extensions of lipshitz mapping into Hilbert space, *Contemporary Mathematics* **26**: 189-206, 1984
5. Juditsky A., Hjalmarsson H., Benveniste A., Delyon B., Ljung L., Sjoberg J. and Zhang Q.: Nonlinear Black-box Models in System Identification: Mathematical Foundations, *Automatica* 31(12): 1725–1750, 1995.
6. Ljung,L.: System Identification -Theory for the User, Prentice-Hall, N.J. 2nd edition, 1999.
7. Lorenz, E.: Deterministic nonperiodic flow, *Journal of the Atmospheric Sciences*, 20 (2): 130141,1963.
8. Matousek, J.: On variants of the JohnsonLindenstrauss lemma. *Random Structures and Algorithms*, 33(2): 142–156, 2008.
9. Nelles O.: Nonlinear system identification: from classical approaches to neural networks and fuzzy models, Springer-Verlag, Berlin Heidelberg 2001.
10. Sjberg, J.,Q. Zhang, L.Ljung, A.Benveniste, B.Delyon, P.-Y.Glorennec, H.Hjalmarsson, and A.Juditsky: Non-linear Black-box Modeling in System Identification: a Unified Overview, *Automatica*, 31:1691–1724, 1995.
11. Skubalska-Rafajowicz, Ewa: Neural networks with sigmoidal activation functions-dimension reduction using normal random projection, *Nonlinear Analysis: Theory, Methods and Applications*,71 (12)e1255–e1263,2009.
12. Woodruff D.P.: Sketching as a Tool for Numerical Linear Algebra. *Foundations and Trends in Theoretical Computer Science* 10(1–2): 1–157, 2014.

Designing the process model of steam superheating in a power boiler and the adaptive control system which controls this process

Mateusz Jabłoński

Wrocław University of Science and Technology, 50-370 Wrocław, Poland,
mateusz.jablonski@pwr.edu.pl

Abstract. In this paper the process model of steam superheating system, localized inside a power boiler; and the advanced, cascade control system which is used to maintain a steam temperature are proposed. Model and control loop have been implemented and investigated in Distributed Control System and in Matlab as well.

Keywords: Power Plant, Superheater, DCS, Matlab, Model, Control, Optimization, PID

1 Introduction

The steam superheating process is realized in heat exchangers named steam superheaters. They are made of many combined metal tubes in which steam flows. Superheaters are located inside the boiler on the top and connected on the one side with a drum and on the second side with the main steam output valve. Exchangers are serial connected, on the same pipe. Inside the power boiler, there may be a few stages of superheating. It depends of boiler construction. Additionally, between each superheater, there is located a water attemperator to cool down the steam, if its temperature grows up. Sprays allow to stabilize a steam temperature on a set point and do not let to destroy the boiler [1], [2].

2 Modeling approaches of the steam superheating system

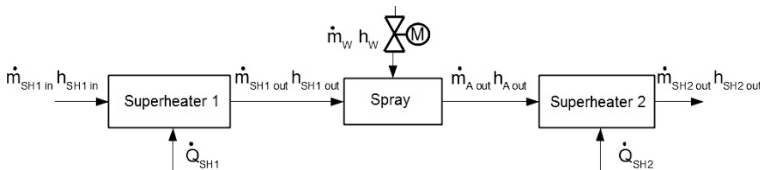


Fig. 1. General structure of the system

Figure 1 shows the ideological scheme of the steam superheating system which contains two separated superheaters and one attemperator between them [5], [6].

2.1 Mathematical model of the system

Superheater 1. The input steam, which is coming from the drum, may be described by the following variables: $\dot{m}_{SH1in}[\frac{kg}{s}]$ as mass flow and $h_{SH1in}[J]$ as specific enthalpy. The output steam of superheater 1 is described by: $\dot{m}_{SH1out}[\frac{kg}{s}]$ as a mass flow and $h_{SH1out}[J]$ as a specific enthalpy. Heat flow from gas to steam in superheater 1 is described by $\dot{Q}_{SH1}[\frac{J}{s}]$. The output steam parameters are equal to the steam parameters inside the superheater. Therefore, it is not necessary to define extra variables except the volume of steam in superheater 1 $V_{SH1}[m^3]$. Density of steam inside the superheater $\rho_{SH1out}[\frac{kg}{m^3}]$ is not constant and it depends of temperature and pressure. Equation 1 describes a dynamics of the first superheater [1], [6].

$$\rho_{SH1out}(t)V_{SH1}\dot{h}_{SH1}(t) = \dot{Q}_{SH1}(t) + \dot{m}_{SH1in}(t)h_{SH1in}(t) - \dot{m}_{SH1out}(t)h_{SH1out}(t) \quad (1)$$

The change of steam specific enthalpy inside the exchanger is equal to the change of output steam specific enthalpy $\dot{h}_{SH1} = \dot{h}_{SH1out}$. Mass flow of input steam is equal to mass flow of output steam too $\dot{m}_{SH1in} = \dot{m}_{SH1out}$. Equation 2 describes the change of specific enthalpy of steam inside the superheater 1 \dot{h}_{SH1out} .

$$\dot{h}_{SH1out}(t) = \frac{\dot{Q}_{SH1}(t) + \dot{m}_{SH1out}(t)[h_{SH1in}(t) - h_{SH1out}(t)]}{\rho_{SH1out}(t)V_{SH1}} \quad (2)$$

Attemperator. The spraying valve is located outside of the boiler and installed on the steam pipe. The parameters of the input steam to the attemperator (mass flow and specific enthalpy) are equal to the parameters of the output steam from superheater 1 $\dot{m}_{SH1out}[\frac{kg}{s}]$ and $h_{SH1out}[J]$. Injected water can be described by $\dot{m}_w[\frac{kg}{s}]$ as mass flow and $h_w[J]$ as specific enthalpy. The output steam parameters from attemperator are mass flow $\dot{m}_{Aout}[\frac{kg}{s}]$ and specific enthalpy $h_{Aout}[J]$. The volume of steam inside the spray chamber is marked as $V_A[m^3]$. Equation 3 describes a dynamics of the attemperator [1], [6].

$$\rho_{Aout}(t)V_A\dot{h}_A(t) = \dot{m}_{SH1out}(t)h_{SH1out}(t) + \dot{m}_w(t)h_w(t) - \dot{m}_{Aout}(t)h_{Aout}(t) \quad (3)$$

Regarding the fact that the output mass flow from attemperator is equal $\dot{m}_{Aout} = \dot{m}_{SH1out} + \dot{m}_w$ and the change of steam specific enthalpy inside the attemperator is equal to the change of output steam specific enthalpy $\dot{h}_A = \dot{h}_{Aout}$, the change of steam enthalpy inside the attemperator \dot{h}_{Aout} may be described by equation 4.

$$\dot{h}_{Aout}(t) = \frac{\dot{m}_{SH1out}(t)[h_{SH1out}(t) - h_{Aout}(t)] + \dot{m}_w(t)[h_w(t) - h_{Aout}(t)]}{\rho_{Aout}(t)V_A} \quad (4)$$

Superheater 2. The input steam parameters: mass flow and specific enthalpy, are equal to the attemperator output steam parameters $\dot{m}_{Aout}[\frac{kg}{s}]$, $h_{Aout}[J]$. Heat transfer from gas to steam in superheater 2 is marked by $\dot{Q}_{SH2}[\frac{J}{s}]$. Density

of the output steam is described by $\rho_{SH2out}[\frac{kg}{m^3}]$ and the volume of superheater 2 is $V_{SH2}[m^3]$. A dynamics of the second exchanger is represented by equation 5 [1], [6].

$$\rho_{SH2out}(t)V_{SH2}\dot{h}_{SH2}(t) = \dot{Q}_{SH2}(t) + \dot{m}_{Aout}(t)h_{Aout}(t) - \dot{m}_{SH2out}(t)h_{SH2out}(t) \tag{5}$$

Due to the fact that $\dot{h}_{SH2} = \dot{h}_{SH2out}$ and $\dot{m}_{SH2in} = \dot{m}_{SH2out}$, the change of the output steam enthalpy \dot{h}_{SH2out} is described by equation 6.

$$\dot{h}_{SH2}(t) = \frac{\dot{Q}_{SH2}(t) + [\dot{m}_{SH1out}(t) + \dot{m}_w(t)][h_{Aout}(t) - \dot{h}_{SH2out}(t)]}{\rho_{SH2out}(t)V_{SH2}} \tag{6}$$

Full mathematical model can be represented by the system of equations 7.

$$\begin{cases} \rho_{SH1out}(t)V_{SH1}\dot{h}_{SH1out}(t) = \dot{Q}_{SH1}(t) + \dot{m}_{SH1out}(t)[h_{SH1in}(t) - h_{SH1out}(t)] \\ \rho_{Aout}(t)V_A\dot{h}_{Aout}(t) = \dot{m}_{SH1out}(t)[h_{SH1out}(t) - h_{Aout}(t)] + \dot{m}_w(t)[h_w(t) - h_{Aout}(t)] \\ \rho_{SH2out}(t)V_{SH2}\dot{h}_{SH2}(t) = \dot{Q}_{SH2}(t) + [\dot{m}_{SH1out}(t) + \dot{m}_w(t)][h_{Aout}(t) - \dot{h}_{SH2out}(t)] \end{cases} \tag{7}$$

The first line of the system of equations is related to the first superheater, the second one describes the spray attemperator and the third one is related to the second heat exchanger. Each addend represents the energy flow of each medium: steam, water and gas. The sums are related to dynamic energy changes inside each element: the first superheater, spray attemperator and the second one [1], [2], [3], [6].

2.2 Model implementation in Matlab/Simulink

Figure 2 shows the realization of heat exchanger implemented in Simulink.

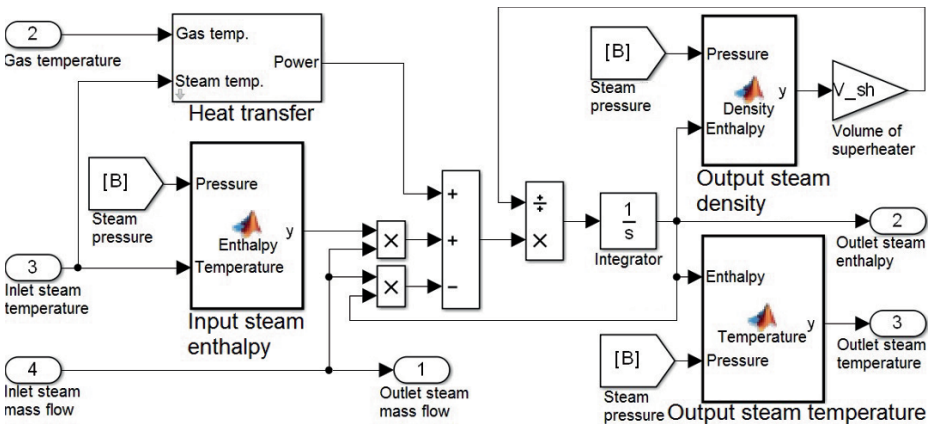


Fig. 2. Supeheater implementation in Simulink

To realize it, *XSteam* library has been used to calculate steam parameters - enthalpy, temperature, pressure and density. The library is based on the enthalpy-entropy (h-s) chart.

System calculates the output steam parameters - enthalpy h_{SHout} and temperature T_{SHout} , according to the input values - gas temperature T_G , input steam temperature T_{SHin} , steam pressure P_{SH} and steam flow $\dot{m}_{SHin} = \dot{m}_{SHout}$. The resultant value of the heat transfer from gas to steam \dot{Q}_{SH} , heat flow in input steam $h_{SHin} * \dot{m}_{SHin}$ and heat flow in output steam $h_{SHout} * \dot{m}_{SHout}$ (negative) is integrating to calculate total energy inside superheater. This value is divided by total steam mass in exchanger (dynamically updated because of density changes in block *output steam density*) to determine specific enthalpy of output steam h_{SHout} . It is necessary to calculate a temperature of output steam T_{SHout} as well. The temperature will be used by the steam temperature control system.

The attemperator model has been realized in the same way like the superheater model.

At the end all structures (superheater 1, attemperator and superheater 2) have been combined to one full steam superheating model according to figure 1.

2.3 Dynamics analysis

Figure 3 presents model responses to disturbance of input steam temperature (on the left) and gas temperature (on the right).

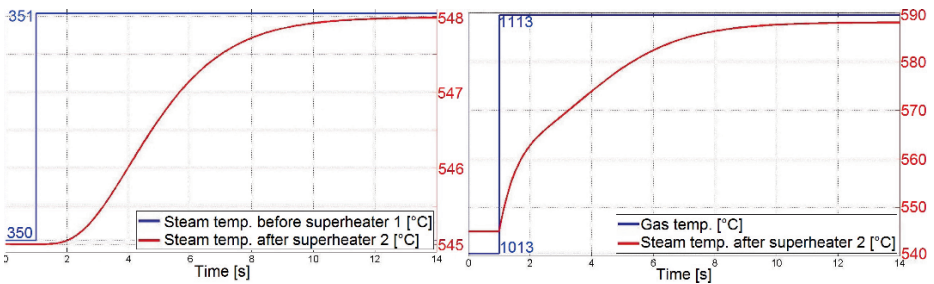


Fig. 3. Model responses to disturbance of input steam and gas temperature (around superheater 1 and 2)

Response to the input steam temperature change has a characteristic of higher order object because inertial objects - superheater 1, attemperator and superheater 2 are serial connected. Change of the input steam temperature is not equal to the output steam temperature change due to fact that presented model of steam superheating is nonlinear. Each change of steam temperature or pressure causes density of steam change which is not linear - changes according to the steam tables.

Increase of gas temperature caused higher heat transfer from gas to steam flowing through both superheaters. Therefore, temperature of steam increased according to the second figure. Response to flows disturbances has been shown on figure 4.

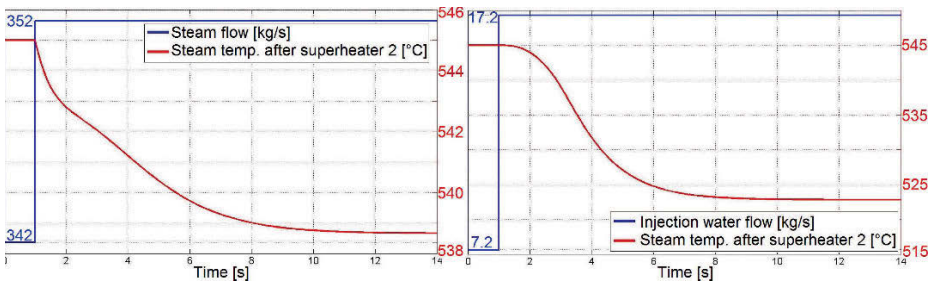


Fig. 4. Model responses to disturbance of steam and water flow

On the left chart there is a model response to steam flow increase. output steam temperature decreased because less heat from gas has been absorbed by steam. In conclusion to maintain specific enthalpy of steam it is necessary to increase gas temperature together with mass flow of steam.

Increase of injection water flow by opening the valve caused decrease of output steam temperature due to fact that water enthalpy is significantly less then steam enthalpy. Water injection caused that mass flow of steam increased but its specific enthalpy decreased. It has a direct influence on the output steam temperature.

Time constant of model mainly depends of superheaters volume and number of them. Larger heat exchangers need more time to reach a steady state.

Model dynamics also depends on the injection valve construction. There are many valves with different flow characteristics. Size of the valve and dynamics of servomechanism installed on it can impact to the dynamics of whole process.

3 Advanced control system of output steam temperature

One of the main parameters in the power boiler to stabilize is temperature of the output steam. Operator is able to set the setpoint of steam temperature and implemented control loop has to maintain set value despite of disturbances - changes of the boiler load, soot on steam pipes, changes of steam parameters (density, pressure and mass flow) and so on. Steam temperature keeping is very important because if too high value can destroy the boiler and causes huge monetary losses. Furthermore, output steam temperature impacts on unit efficiency, so it has to be maintain the best possible.

Classic control system is built of PID controllers in cascaded configuration. Additional elements which are usually added to the control loop are feed-forward

signals from boiler load and mass flow of steam. Another issue is PID controllers tuning. Their parameters should be dynamically adapted to the operating point because of object nonlinearity [6].

3.1 Implementation in Emerson Ovation Distributed Control System

Figure 5 presents the realization of cascaded control loop with PID parameters optimization in one of the most used DCS in power engineering - Emerson Ovation DCS v3.5.1.

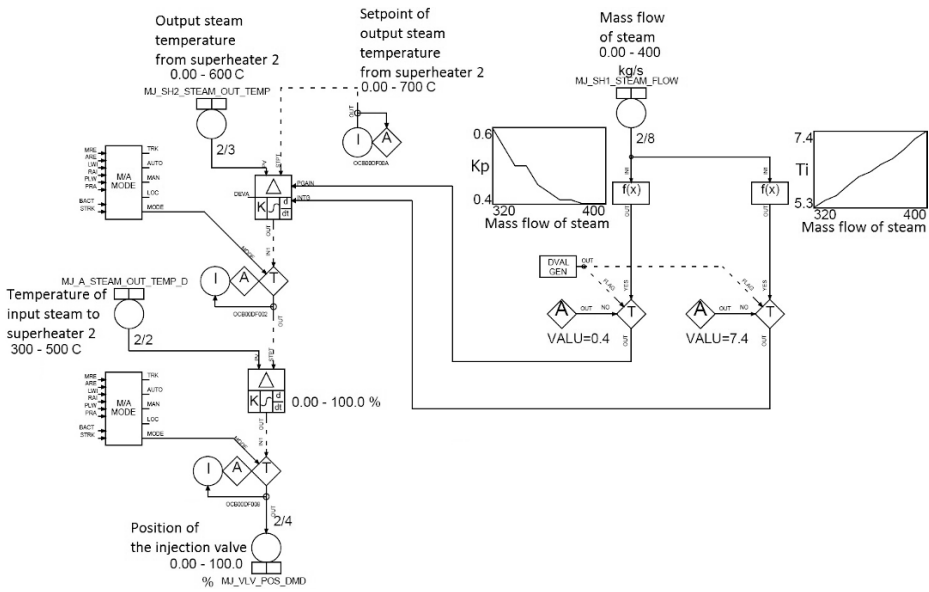


Fig. 5. Control logic of cascaded PID control system in Emerson Ovation DCS v3.5.1

Presented control system is used in each conventional power plant, where superheating system takes a place and in any case, cascaded control loop is used. Therefore, control logic presented on figure 5 is able to be used in real power plant. The system was tested and tuned with model presented in section 1. The type of controllers used to realize control logic is IND - independent algorithm. The PID transmittance is $G_{PID}(s) = K_p + T_i \frac{1}{s} + T_d s$.

Two PID controllers are connected in cascade. The first one (on the bottom) directly controls the injection valve by position setting at the output according to the steam temperature measured at the inlet to superheater 2. Its setpoint value is connected to the second PID controller - master (on the top). This controller sets the value of temperature at the inlet to superheater 2 which should be reached to maintain the temperature at the outlet. The outlet temperature is set by an

operator and it is the setpoint of the master PID. The outlet temperature is connected to the master PID as process value. This structure allows to prevent the output temperature change before it will change. The sign that the output temperature is going to change is the input temperature change which is noticed earlier than output one [6].

Operator can change the operating mode of PID from automatic to manual if it is necessary in any time. This function is realized by M/A Mode block. In manual mode operator may set manually the position of the injection valve.

3.2 PID parameters optimization

To obtain the best regulation in different operating points, optimization of PID parameters has been performed. For this purpose PID has been tuned in a few operating points - in different unit load. Parameters for each operating points have been used to create the functions which are correlation between unit load (steam flow) and PID parameters. These functions are connected to master PID (proportional and integration pins) on figure 5. It allows to dynamically change PID parameters in real time to perform the best regulation. Results of optimization are shown in table 1. Optimization process has been performed in Emerson Ovation DCS on the model of steam superheating system implemented in DCS which is equal to the model in Matlab presented in section 1 (all parameters are preserved).

Load[%]	Steam[kg/s]	Kp	Ti	Reg time [s]		ISE	
				Disabled	Enabled	Disabled	Enabled
80	320	0,6	5,3	45	35	49,66	37,67
82	328	0,55	5,5	44	35	46,25	37,75
84	336	0,5	5,63	43	36	46,17	37,55
86	344	0,5	5,9	42	36	43,73	37,35
88	352	0,45	6,15	42	36	42,89	37,72
90	360	0,43	6,28	41	36	40,81	37,75
92	368	0,41	6,5	40	37	40,53	37,61
94	376	0,41	6,65	40	37	39,65	37,8
96	384	0,4	6,9	39	38	38,24	37,36
98	392	0,4	7,2	39	38	38,1	37,71
100	400	0,4	7,4	38	38	37,44	37,45

Table 1. PID controller optimization results

A quality of optimization was tested with the following criterions: Integral Square Error (ISE) and regulation time. In this case ISE means integral of square

difference between setpoint and process value of the output steam temperature - absolute error. According to table 1 if optimization is enabled, ISE and regulation time are almost constant but if optimization is disabled they are not. Without optimization system is correctly tuned only for full boiler load [4].

3.3 Test of the model and control system on the real data

Implemented control system with process model has been tested on the real data exported from the power plant which is localized in the United States of America. This data has been used for the model identification. Results of the test are shown on figure 6. It can be noticed that the highest difference between set temperature and measured is about 15 °C. It occurs when the unit load (steam flow) immediately drops down.

The system correctly controls the injection valve. When the unit load drops, the system closes the valve. Otherwise the valve is more opened.

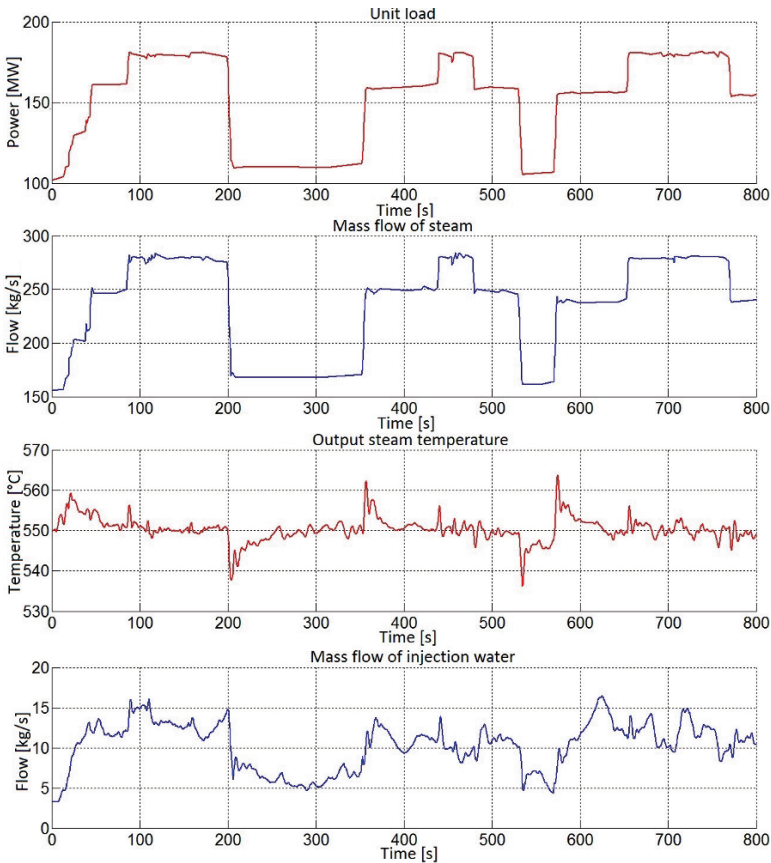


Fig. 6. System tests on the real data exported from american power plant

Above data has been exported from working power plant in USA and connected to the presented model in Matlab.

4 Conclusion

Implemented model of steam superheating can simulate situations which occurs on real power plant. The model successfully approximates reactions on load changes, steam parameters changes and injection water parameters changes.

Cascaded PID system correct controls the output steam temperature. To reach higher performance of the system, optimization of PID parameters has been done. After optimization process regulators work properly on different operation points.

The behaviour of the system on the real data was correct. Cascaded control loop has maintained the set value of the output steam temperature even if significantly unit load drops occurred.

References

- [1] Kruczek, S.: *Kotły*. Oficyna Wydawnicza Politechniki Wrocławskiej. 11–18, 298–309 (2001)
- [2] Laudyn, D., Pawlik, M., Strzelczyk, F.: *Elektronie*. Wydawnictwo Naukowo-Techniczne Warszawa. 32–35, 124–126 (2000)
- [3] Wiśniewski, S.: *Termodynamika techniczna*. Wydawnictwo Naukowo-Techniczne Warszawa. 13–65 (1993)
- [4] Łysakowska, B., Mzyk, G.: *Komputerowa symulacja układów automatycznej regulacji w środowisku Matlab/Simulink*. 93–107 (2005)
- [5] Hlava, J., Opálka, J., Johansen, T. A.: *Model predictive control of power plant superheater - comparison of multi model and nonlinear approaches*. Technical University of Liberec., Norwegian University of Science and Technology Trondheim. 311–312 (2013)
- [6] Li, K., Chan, K. H., Ydstie, B. E.: *Adaptive Inventory Control of Superheater Systems*. Department of Chemical Engineering Carnegie Mellon University Pittsburgh. 1–4 (2008).
- [7] Song, X. L., Liu, Ch., Song, Z., Song, X. F.: *Robust PID control for steam superheater*. College of Electrical Engineering and Informational Science, Hebei University of Science and Technology Shijiazhuang. 988–991 (2004)

Identification of Discrete-Time Model of Active Magnetic Levitation System

Kamil Czerwiński, Maciej Ławryńczuk

Institute of Control and Computation Engineering, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland, tel. +48 22 234-71-24
kamil.czerwinski@emerson.com, M.Lawrynczuk@ia.pw.edu.pl

Abstract. This paper describes model identification of an active magnetic levitation laboratory process. Magnetic levitation is a nonlinear, open-loop unstable and time-varying dynamical system whose modelling and control are difficult. Theoretical model takes into account set of parameters which are difficult to identify. The proposed approach assumes discrete-time form of the model the parameters of which are tuned using nonlinear optimisation methods. The obtained model is more accurate and can be directly used in model-based control algorithms. Results of real experiment are demonstrated for the INTECO MLS2EM laboratory set-up.

Key words: Magnetic levitation, identification, unstable and nonlinear system, discrete-time model.

1 Introduction

MAGnetic LEVitation system (MAGLEV), also named magnetic suspension, is a physical phenomenon in which a levitating ferromagnetic object is suspended with no support other than magnetic fields, without any contact with the surrounding field [6, 7]. The constantly generated magnetic force is used to counteract the effects of the gravity force. Due to the fact that the theory of electromagnetism is well developed, this phenomenon is used in many applications, including MAGLEV trains, contactless melting, magnetic bearings and for product display purposes. Rapid development in electronics and advanced control theory makes practical applications in which magnetic levitation is used possible.

From the theoretical point of view, the MAGLEV is a nonlinear, open-loop unstable dynamic process whose modelling and control are difficult. In the simplest, and quite typical, configuration magnetic levitation systems are controlled as SISO systems, using only one electromagnet. Such systems (MLS1EM) [5] use electromagnetic force generated by applying the required voltage to the electromagnet and one-axis distance sensor [6]. In more advanced configurations (MLS2EM) the second electromagnet located can be utilised to generate the external gravity force, of a different nature than that generated by the first one. Alternatively, both electromagnets can work together in one axis as the active

magnetic bearing [6]. In all mentioned configurations a real-time controller must be utilised in order to generate the force(s).

In order to design the controller, a precise dynamic model of the levitation process is necessary. Two main approaches are possible:

- a) first-principle mathematical model development,
- b) black-box nonlinear model identification.

In the first approach the structure of the model results from the well-known fundamental laws governing the process, but the parameter must be tuned for a specific application [2, 8]. In the second one the structure of the model is chosen arbitrarily, an example neural model of the levitation process is described in [1]. The black-box may result in a model which does not work properly in the full operating range.

In this work the problem of tuning the parameters of the first-principle dynamic model of a laboratory magnetic levitation system produced by INTECO is studied [2]. Tuning of the theoretical model is necessary since advanced control algorithms exploit a model to calculate on-line the optimal value of the manipulated variable. When the properties of the model are significantly different from those of the real process, the controller is likely to generate a wrong control policy. Since the process is open-loop unstable, closed-loop identification with a Proportional-Derivative (PD) controller is used. Using a set of data measured in experiments, the parameters of the model are tuned by a global nonlinear optimisation procedure. It makes it possible to significantly increase accuracy of the initial model. Starting with a continuous-time model, a discrete-time one is obtained using the backward Euler method. Additionally, the discrete model is extended to take into account the rocking effect, which can be treated as a disturbance model.

2 Laboratory Magnetic Levitation System

The INTECO MLS2EM laboratory process is presented in the Fig. 1. It consists of: a frame, two electromagnets, a ferromagnetic sphere, a position sensor and coil current sensors. Fig. 2 depicts the general configuration of the process. Power and communication interfaces allow to design and run experiments in real-time directly from MATLAB/Simulink environment. In this configuration the user designs the controller directly from MATLAB/Simulink. It is next compiled into the C programming language using the MATLAB Coder and Simulink Coder toolboxes. During on-line control the resulting code, which contains the software implementation of the control algorithm, is run on the PC computer. The basic objective of the process is to generate voltage applied to one or both electromagnets in such a way that the ferromagnetic object levitates. The object has a shape of a sphere. Its vertical position is determined by a position sensor and coil current is also measured. The values of the position and of the current are next used to tune the model.

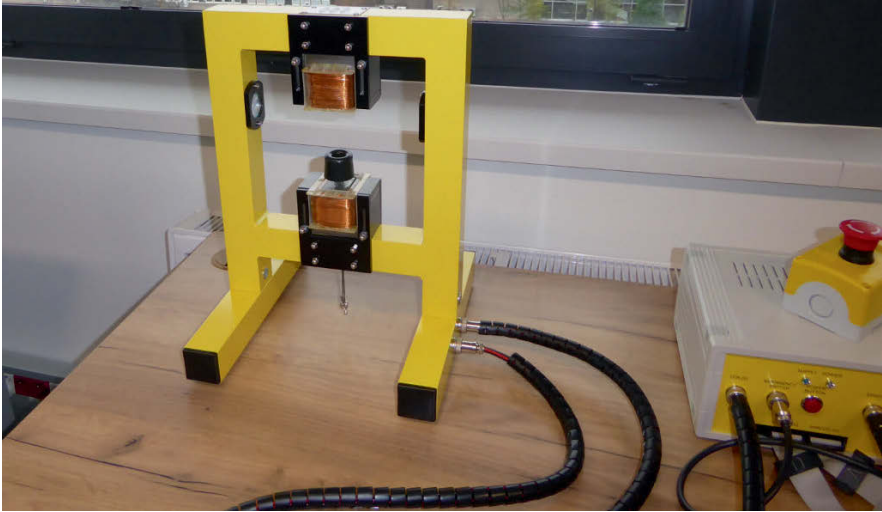


Fig. 1: The MLS2EM laboratory process

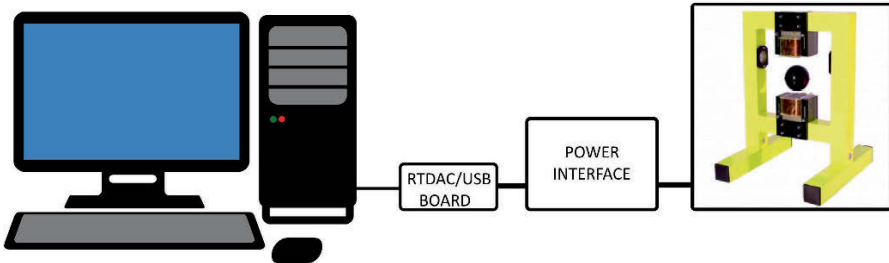


Fig. 2: Configuration of the MLS2EM laboratory process

In general, two configurations of power actuators designed to the MLS2EM process are possible [6], which can have same impact on model quality (and the possible discrepancy between the model and the process). The first configuration bases on the hardware current controller, the second one on Pulse Width Modulation (PWM) control, which is less effective. It is because the PWM control assumes constant period of the signal with a variable duty cycle. Moreover, noise in PWM control can be more important. The laboratory process used in the experiments described in this work uses the PWM control scheme and it is connected to the PC computer which acts as a controller by an USB universal serial port. The highest supported frequency of the PWM signal is 1kHz.

3 Continuous-Time Model

The mathematical dynamic model of the process should include mechanical construction, electrical phenomena and behaviour of the sensor. The schematic diagram of the described system is shown in Fig. 3.a. The mechanical representation of the system using stiffness k and damping c parameters is shown in Fig. 3.b. The theoretical mathematical model of the MLS2EM process is well-known [2, 5, 8]. It consists of four first-order differential equations corresponding to the mechanical and electrical compounds [2]

$$\begin{cases} \frac{dx_1(t)}{dt} = x_2(t) \\ \frac{dx_2(t)}{dt} = -\frac{F_{em1}(t)}{m} + g + \frac{F_{em2}(t)}{m} \\ \frac{dx_3(t)}{dt} = \frac{1}{f_i(x_1(t))} (k_i u_1(t) + c_i - x_3(t)) \\ \frac{dx_4(t)}{dt} = \frac{1}{f_i(x_d - x_1(t))} (k_i u_2(t) + c_i - x_4(t)) \end{cases} \quad (1)$$

where

$$F_{em1}(t) = x_3^2(t) \frac{F_{emP1}}{F_{emP2}} \exp\left(-\frac{x_1(t)}{F_{emP2}}\right) \quad (2)$$

and

$$F_{em2}(t) = x_4^2(t) \frac{F_{emP1}}{F_{emP2}} \exp\left(-\frac{x_d - x_1(t)}{F_{emP2}}\right) \quad (3)$$

are the forces generated by the first electromagnet (EM1) and the second one (EM2), respectively. The characteristics of two power actuator units are described by equations

$$f_i(x_1(t)) = \frac{f_{iP1}}{f_{iP2}} \exp\left(-\frac{x_1(t)}{f_{iP2}}\right) \quad (4)$$

for $i = 1, 2$. The state variables are: x_1 – the distance of the object (sphere) from the surface of the upper electromagnet, x_2 – the velocity of the object, x_3 – the current in upper coil (EM1), x_4 – the current in bottom coil (EM2). F_{emP1} and F_{emP2} denote the forces generated by the upper and the bottom coils, respectively. Gravitational acceleration is denoted by g , the mass of the object is m . The manipulated variables, i.e. the PWM duty cycles applied to the first and the second electromagnets are denoted by u_1 and u_2 , respectively. Actuator characteristics for upper and bottom coils use damping c_i and stiffness k_i factors, where $i = 1, 2$. Current functions for both coils are defined by f_{iP1} , f_{iP2} parameters, where $i = 1, 2$. In laboratory system both electromagnets are the same so in the model we can use only one set of actuator parameters for both electromagnets, the same as in [2]. The distance between upper and bottom electromagnet minus sphere diameter is denoted by x_d .

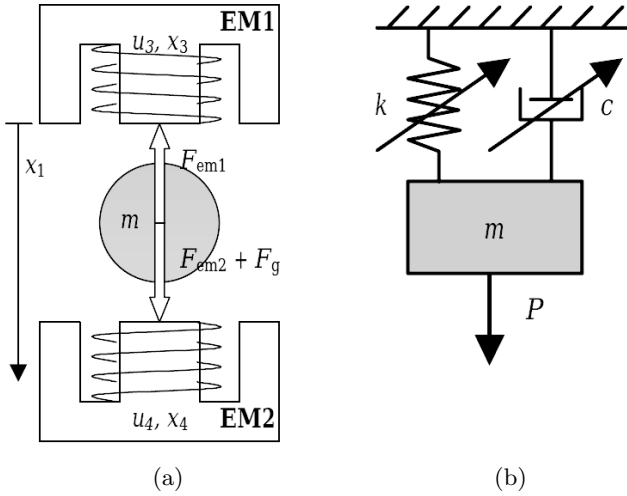


Fig. 3: Active magnetic suspension: a) MLS diagram, b) mass-spring-damper equivalent of upper EM [6]

4 Model Tuning and Modification

Differences between behaviour of the original theoretical model (1) and that of the real laboratory MLS2EM process are inevitable. In order to increase model accuracy, in this work a specialised tuning procedure is proposed. Additionally, a disturbance model is also used.

During the experiments it is assumed that:

- a) sensor and actuator characteristic in laboratory set-up is identified using the procedure recommended by the producer of the process, as described in [2],
- b) only the upper electromagnet is controlled,
- c) the bottom electromagnet is present and its duty cycle is at bottom limit during the experiment,
- d) a feedback PD controller is used,
- e) stability of the closed-loop is in practice guaranteed by a proper tuning of the controller parameters,
- f) only one ferromagnetic object is used during a single cycle of tuning procedure,
- g) the object is levitating without any external disturbances,
- h) the object holder is mounted on the bottom electromagnet.

During the experiments, in order to generate the data next used for model tuning, the (vertical) location set-point is changed in the whole operating range at random times. Identification signal should contain the suitable frequency components. The sampling time is 1ms, which means that process input and output variables as well as the output set-point are collected as frequently as 1000 times per second.

4.1 Close-Loop Model Tuning Using Particle Swarm Optimisation Algorithm

It is a very important decision to choose a proper set of parameters from the process model (1) which may be modified during the model identification procedure. After excluding some parameters which are known or could be directly measured (m, g, x_d), one has 6 parameters which must be determined by the optimisation routine: $k_i, c_i, f_{iP_1}, f_{iP_2}, F_{emP_1}, F_{emP_2}$. For each parameter its minimal and maximal value must be defined. This is required, because the purpose of optimisation is only to tune the physical parameters of the model, full global optimisation with unconstrained parameters may lead to parameters which are far from the physical configuration of the process. Model parameters are calculated as a result of an optimisation process during which the errors between the recorded data and model output is minimised. The minimised cost-function is hence

$$E = \sum_{n=1}^N (y_{\text{data}}(n) - y_{\text{model}}(n))^2 \quad (5)$$

where $N = 50000$ is the length of the data set (data is recorded during 50 seconds of process operation), $y_{\text{data}}(n)$ is the process output (position) measured and recorded in the real-time experiment, $y_{\text{model}}(n)$ is the model output calculated for the process input signal (the manipulated variable) used in the real experiment. Since the process model defined by Eqs. (1) is nonlinear, the minimised cost-function (5) is nonlinear. That is why the parameters of the model are determined using Particle Swarm Optimisation (PSO) [3] which is a heuristic optimisation algorithm, often used when it is necessary to solve multimodal, highly nonlinear optimisation tasks. The PSO algorithm is able to give good results in numerous applications in which global optimisation is necessary. In the existing work, other global optimization algorithms have not been tested.

During optimisation of model parameters the differential equations defining the fundamental model (1) must be solved. It has been verified experimentally that MATLAB/Simulink implementation is very time consuming. Single simulation of 50 seconds of real-time takes 59 second. Because the PSO optimisation algorithm requires calculation of the cost-function (5) thousands of times, calculations are very slow in MATLAB/Simulink. That is why the whole optimisation program has been written in C programming language. In that approach single simulation of 50 seconds of real-time takes 2 second.

4.2 Discrete Nonlinear Model Using Euler Discretisation

In this work the backward Euler method is in order to approximate continuous-time derivatives by means of discrete-time differences

$$\frac{dx_i(t)}{dt} \approx \frac{\Delta x}{\Delta t} = \frac{x_i(k+1) - x_i(k)}{t(k+1) - t(k)} = \frac{x_i(k+1) - x_i(k)}{T_s} \quad (6)$$

Where T_s is the sampling time. Using the Euler method of discretisation (6), the continuous-time model (1) becomes

$$\begin{cases} x_1(k+1) = x_2(k)T_s + x_1(k) \\ x_2(k+1) = -\frac{T_s}{m} \left(\tilde{F}_{em1}(k) + \tilde{F}_{em2}(k) \right) + T_s g + x_2(k) \\ x_3(k+1) = \frac{T_s}{\tilde{f}_i(x_1(k))} (k_i u_1(k) + c_i - x_3(k)) + x_3(k) \\ x_4(k+1) = \frac{1}{\tilde{f}_i(x_d - x_1(k))} (k_i u_2(k) + c_i - x_4(k)) + x_4(k) \end{cases} \quad (7)$$

where

$$\tilde{F}_{em1}(k) = x_3^2(k) \frac{F_{emP1}}{F_{emP2}} \exp\left(-\frac{x_1(k)}{F_{emP2}}\right) \quad (8)$$

$$\tilde{F}_{em2}(k) = x_4^2(k) \frac{F_{emP1}}{F_{emP2}} \exp\left(-\frac{x_d - x_1(k)}{F_{emP2}}\right) \quad (9)$$

and for $i = 1, 2$

$$\tilde{f}_i(x_1(k)) = \frac{f_{iP1}}{f_{iP2}} \exp\left(-\frac{x_1(k)}{f_{iP2}}\right) \quad (10)$$

4.3 Modification of Discrete-Time Model (Rocking Phenomena)

The PSO fundamental model of the process ((1) or (7)) does not take into account that in practice during levitation the ferromagnetic object moves in 3D space. The position sensor gives measurement only in one dimension (i.e. vertical position), hence any movement in perpendicular direction has a big impact on the position measurement. It is easy to noticed that such a phenomenon, called rocking, has a harmonic nature with two frequency components. The low frequency is responsible for rocking and the high frequency is responsible for vibrations and noise. In order to take into account the rocking phenomenon, the discrete-time dynamic model (7) is extended by adding a simple disturbance model consisting of trigonometric functions. In place of the first equation from the model (7), the following equation is used

$$x_1(k+1) = x_2(k)T_s + x_1(k) + F_1(\sin(F_2k)) + G_1(\cos(G_2k)) \quad (11)$$

where F_1 , F_2 , G_1 and G_2 are additional parameters.

5 Results of Experiments

The INTECO MLS2EM laboratory process with the RTDAC/USB card is used in the real-time experiments. The ferromagnetic sphere with mass $m = 0.039$

kg and diameter $d = 0.05$ m is used as the levitating object. As it has been told, the process is open-loop unstable which means that no experiments are possible without a controller. The PD controller is used with the proportional gain $K_p = 55$ and the time constant of the differential part $T_d = 4$, $u_0 = 0.35$ is the value of the manipulated variable (process input), which is applied as a constant bias. Control law of PD controller is hence

$$u(k) = K_p e(k) + T_d \left(\frac{e(k) - e(k-1)}{T_s} \right) + u_0 \quad (12)$$

where e is the control error, i.e. between the value of the process output and its set-point.

Tab. 1 details the 6 optimised parameters of the model. For each of them the initial values are given. Since for optimisation using the PSO algorithm box constraints of all model parameters are used, their minimal and maximal values are given. The range of model parameters are chosen in experimental way, taking into account the convergence and efficiency of the algorithm. Finally, the optimised values of model parameters are given. In this work the PSO algorithm with 14 individuals and 30 generations is used. For such a configuration the average optimisation time is 25 minutes.

Table 1: Model parameters: minimal and maximal values, the initial values as well as the optimised final ones

Parameter	Initial value	Minimal value	Maximal value	Optimised value
k_i	2.5165	2.4	2.6	2.5168
c_i	0.0243	0.01	0.03	0.01545
f_{iP_1}	1.4142×10^{-4}	1×10^{-4}	1.8×10^{-8}	1.36×10^{-4}
f_{iP_2}	4.5626×10^{-3}	4×10^{-3}	5×10^{-3}	4.4651×10^{-3}
F_{emP_1}	1.7521×10^{-2}	1×10^{-2}	2×10^{-2}	1.395×10^{-2}
F_{emP_2}	5.8231×10^{-3}	5.3×10^{-3}	6.2×10^{-3}	6.2×10^{-3}

Fig. 4 compares graphically the data used for identification vs. the output of the initial model and the output of the tuned one. One may notice that the signal generated by the rudimentary model is different than the data and optimisation of model parameters makes it possible to significantly reduce model error. The error defined by Eq. (5) is $E = 6.5 \times 10^{-2}$ and $E = 4.34 \times 10^{-4}$ for the initial and tuned model, respectively. Next, the continuous-time model is discretised. The error of the discrete-time model is $E = 4.396 \times 10^{-4}$. Finally, the rocking phenomenon is taken into account, according the Eq. (11). It makes it possible to further reduce the model error to $E = 4.2404 \times 10^{-4}$. The additional parameters are chosen experimentally, the following values are used: $F_1 = 1 \times 10^{-5}$, $F_2 = 8.9 \times 10^{-3}$, $G_1 = 9 \times 10^{-7}$, $G_2 = 9 \times 10^{-2}$. As depicted in Fig. 5, the modified discrete-time tuned model gives the process trajectory very similar to the real data.

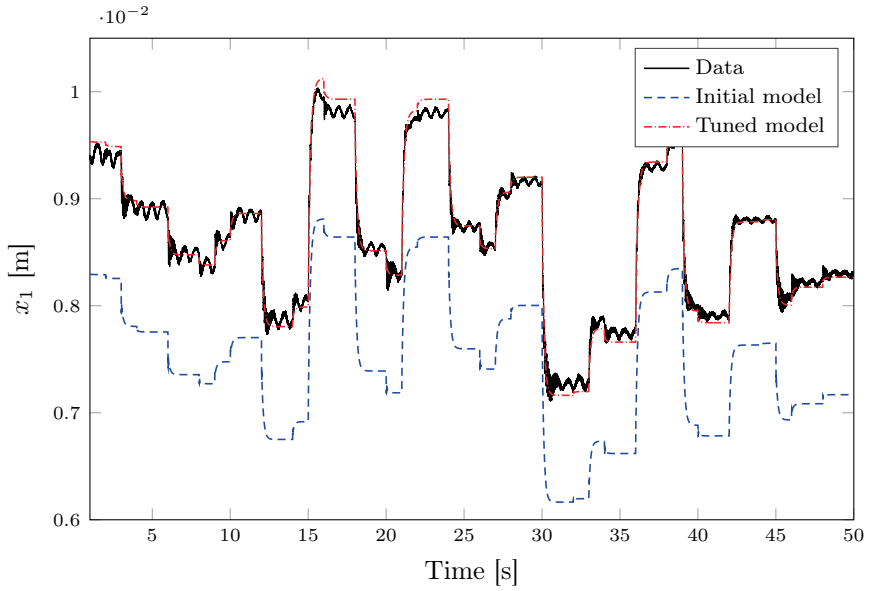


Fig. 4: The data vs. the output of the initial model and the output of the tuned one

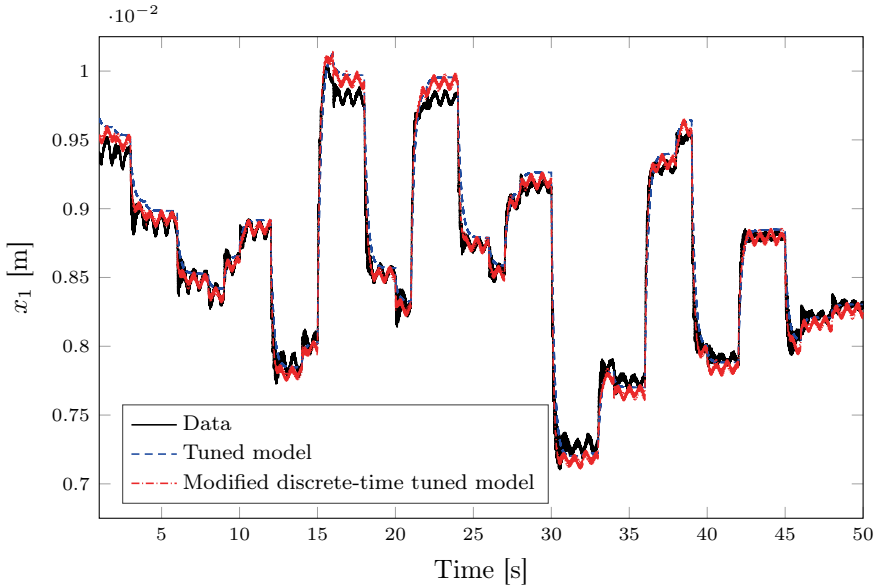


Fig. 5: The data vs. the output of the tuned model and the output of the modified discrete-time tuned one

6 Conclusions

This work considers the problem of finding a precise nonlinear dynamic model of the levitation process. For model identification a set of data recorded during real-time experiments is used. The process is open-loop unstable, which means that a controller (PD one) must be used during process operation. The output of the rudimentary fundamental model is significantly different than the data. In order to tune the model its parameters are tuned using the PSO nonlinear optimisation algorithm. Next, the model is modified by adding a rocking model to describe small movements of the levitation object not modelled by the fundamental laws. As presented in this work, the output of the modified tuned model is very similar to the data recorded during real process operation. In future works it is planned to incorporate the obtained model in model-based controllers, in particular in Model Predictive Control (MPC) algorithms.

References

1. Antić, D., Milovanović, M., Nikolić, S., Milojković, M., Perić, S.: Simulation model of magnetic levitation based on NARX neural networks. *International Journal of Intelligent Systems and Applications* 5, 25–32 (2013)
2. MLS2EMUSB2, Magnetic Levitation System Users Guide, INTECO Ltd. Kraków, 2015,
3. Kennedy, J., Eberhart, R.C.: *Swarm Intelligence*. Morgan Kaufmann (2001)
4. Piłat, A.: Modelling, investigation, simulation, and PID current control of Active Magnetic Levitation FEM model, The 18th International Conference on Methods & Models in Automation & Robotics (MMAR), Miedzyzdroje, Poland, pp. 299–304 (2013)
5. Piłat, A.: Active magnetic levitation systems (in Polish), *Pomiary Automatyka Robotyka* 12, 146–149 (2011)
6. Piłat, A.: Testing performance and reliability of magnetic suspension controllers. *IFAC Proceedings Volumes*, vol. 42, issue 13, pp. 164–167 (2009)
7. Sinha, P.K.: *Electromagnetic Suspension. Dynamics & Control*. London, Peter Perginus Ltd. (1987)
8. Yu, W., Li, X.: A magnetic levitation system for advanced control education. *IFAC Proceedings Volumes*, vol. 47, issue 3, pp. 9032–9037 (2014)

Part IX
Fault Detection, Security and Diagnostics

Cyber Security Provision for Industrial Control Systems

Marek Amanowicz and Jacek Jarmakiewicz

Military University of Technology, gen. Sylwestra Kaliskiego 2, 00-908 Warsaw, Poland,

Abstract. The paper gives an overview of the infrastructure of the electric power control system. It describes the basic threats caused by unauthorized actions and provides examples of the energy management system infrastructure vulnerabilities and their possible consequences. An architecture of a system that offers an effective protection of the Industrial Control Systems from cyber threats is described in some details. The test scenarios used for the system verification and the results of experiments performed both in Supervisory Control and Data Acquisition testbeds and in a operational power control substation are also presented.

Keywords: cyber security, Industrial Control System (ICS), power control system (PCS), Supervisory Control and Data Acquisition (SCADA)

1 Introduction

Information technology plays an extraordinary role in developing so called smart electrical grid. However, it should be noticed that reliability, stability and security of power systems is an essential factor for proper operation of many other critical systems. The use of digital information and automated and interactive technologies makes such infrastructure expose to cyber-threats.

Industrial Control Systems (ICS) that play an important role in achieving reliability and stability of electric power systems become often victims of cyber-attacks. Such unauthorized activities are performed by various cybercrime or hostile organizations that exploit the weakest points, like people mistakes or vulnerabilities of the technical components. The attack at the Ukrainian electric power companies in 2015 shows the reality of such threats and potential consequences of long-term lack of power supply. It also confirms increasing complexity of such advanced persistent threats (APT) [1]. Successfully performed attack may direct impact on other control systems, which confirms the analysis of incidents presented in the Repository of Industrial Security Incidents [2].

The statistics presented in [3], [4] show that majority of computer incidents in critical infrastructure apply to electricity generation and industry sectors. Furthermore, the analysis of identified vulnerabilities confirms increasing efficiency of the detection process, however it also indicates growing complexity of the industrial systems. This requires elaboration the decision support mechanisms that increase the efficiency of the unauthorized activities detection in Industrial

Control Systems and offer effective tools to maintain secure and reliable operation of the systems in the presence of APT. Some examples of potential solutions were discussed in [8], [9].

The rest of the paper is organized as follows. Section 2 gives a generic overview of the energy management system infrastructure. The basic vulnerabilities of the Industrial Control Systems are discussed in Section 3. Section 4 depicts an architecture of the cyber security protection system and its basic components, which is followed in Section 5 by test scenarios used for the system validation and the results of experiments performed both in Supervisory Control and Data Acquisition (SCADA) testbed and in operational power control substation. The basic features of the proposed solution and its potential implementation areas for cyber security decision support are summarized in conclusions.

2 Energy management system infrastructure

The continuous provision of electric power to the consumers with appropriate quality parameters and in agreed quantity requires the efficient and proper work of a complex power generation, transmission and distribution systems. The main task of the energy services is to maintain constantly the appropriate settings of the system parameters in order to generate and disseminate the right amount of electricity to secure the ever-changing load of energy consumers. The process of energy supply is realized by power plants within less than 30 seconds from the time the demand occurred. This process is controlled in real-time by the Central Control System (CCS). Generators are activated in power plants by the Load Frequency Control (LFC) system, which is responsible for frequency and power control. The CCS operates in real-time in a feedback-loop (Fig. 1). The LFC mechanism supervises the Area Control Error (ACE) metric (1) to keep its value close to zero.

$$ACE = \Delta P + K \cdot \Delta f \quad (1)$$

where:

$\Delta P = P - P_0$; P - assigned area change load [MW]; P_0 - fixed area change load [MW]; $\Delta f = f_0 - f$; f - temporary frequency; f_0 - nominal frequency; K - system constant [MW/Hz].

The required behavior of central power regulator is achieved using proportional and integral (PI) characteristic of control loop according to the equation (2):

$$\Delta P = -\beta \cdot ACE - \frac{1}{T} \int ACE \cdot dt \quad (2)$$

where:

β - coefficient gain of the regulator.

The power control is performed using Inter-Control Center Communications Protocol (ICCP) or SCADA protocols IEC 60870-5-104, IEC 61850 which are carried within IPv4 packets. Turbines are controlled from CCS by sending commands through a separated communication network, which is disconnected from

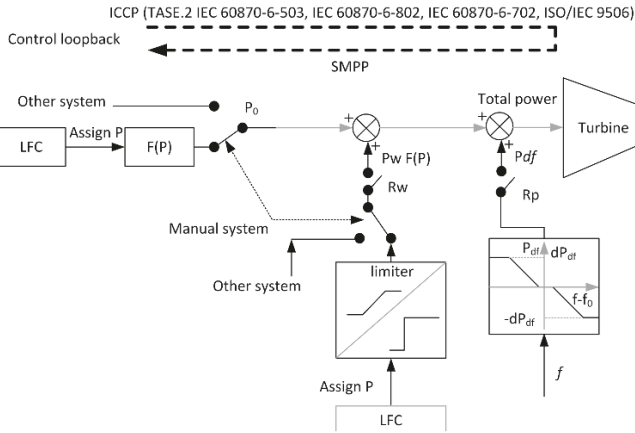


Fig. 1: Electricity generation control [5]

the Internet (Fig. 2). Telecommunication cables are suspended on poles along with high voltage cables. The network is based on the Ethernet standards with VLANs. The exchange of commands and responses from the generators and energy consumption readouts is performed by Control and Supervision Substation (CSS). Control commands from CCS are directed through switches and routers to WAN and are transferred to target routers of the power stations [6]. Control

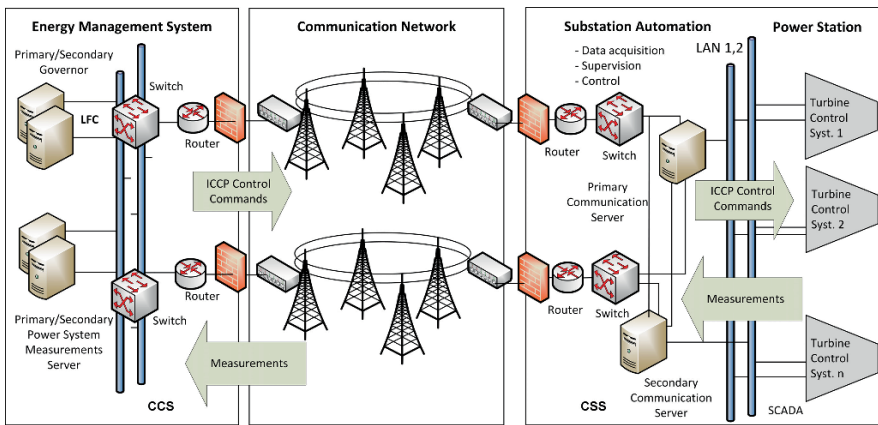


Fig. 2: Infrastructure of electricity generation control

data are sent to CSS station and then forwarded to servers, which send the commands to generators' drivers. The control and regulation of communication equipment and automation systems at the fields is performed from CSS. At the CSS of the power station, the traffic flows through separated Virtual Routing and

Forwarding VLAN networks. Within CCS, the field controllers are responsible for switching processes of electrical circuits, protection of circuits and cooperation with power stations. Commands from CCS are delivered to CSS through routers which are used for readouts from Intelligent Electronic Devices (IEDs). The entire CSS is protected by physical security system. Control commands are processed by SCADA systems in CSS power station.

The energy management system infrastructure often becomes a target of cyber-attacks. If power control system is compromised, an attacker could utilize it to cause catastrophic damage or outages directly or by exploiting paths to critical end devices or connected SCADA systems. The network protocols are one of the most critical parts of power system operations responsible for retrieving information from field equipment, generators and for sending control commands. Despite their key function, the communication protocols have rarely incorporated the security measures, including security against unintentional errors, power system equipment malfunctions, communications equipment failures, or deliberate sabotage.

3 Vulnerabilities of the Industrial Control System

Vulnerability of the ICS refers to the inability of a system to withstand the effects of a hostile environment, which is the result of weaknesses or flaws in technical infrastructure of the system. It should be noticed that vulnerability of the ICS strongly depends on its configuration in a concrete environment, the types of applied control devices and their implementations.

The most popular and available source of information on ICS vulnerabilities becomes the US governmental ICS-CERT¹, which assists control systems vendors and asset owners/operators to identify security vulnerabilities and develop mitigation strategies that strengthen their cyber security posture and reduce the risk. The ICS-CERT database contains the rich set of alerts that provide timely notification to critical infrastructure owners and operators concerning threats or activity with the potential to impact critical infrastructure computing networks. A significant part of these alerts refer to SCADA systems and Programmable Logic Controllers (PLC) that are widely used in ICS.

In FY 2015, ICS-CERT received and responded to 295 incidents² focused on critical infrastructure sectors. Majority of incidents (57%) were reported by Critical Manufacturing Energy and the Water Systems Sectors. The following exemplary vulnerabilities of ICS clearly show the scale of threat and potential risk:

- vulnerability of OpenSSL (Secure Socket Layer) application (commonly known as Heartbleed) identified in the ICS systems implementations of many vendors³;

¹ <https://ics-cert.us-cert.gov/>

² <https://ics-cert.us-cert.gov/Year-Review-2015>

³ <https://ics-cert.us-cert.gov/advisories/ICSA-14-156-01>

- vulnerability of RuggedCom ROS-based and ROX-based devices. An attacker could exploit this vulnerability to perform a POODLE (Padding Oracle On Downgraded Legacy Encryption) attack to force a targeted user and server application to switch to an insecure version of Transport Layer Security (TLS) for secure HTTP communications for the two parties. A successful exploit could allow the attacker to conduct a man-in-the-middle attack and intercept sensitive information between client and server⁴;
- multiple zero-day vulnerabilities for several leading ICS hardware PLCs⁵. The affected PLCs are used to control functions in critical infrastructure in the chemical, energy, water, nuclear, and critical manufacturing sectors;
- multiple vulnerabilities affecting Electric Modicon Quantum PLC⁶ that are exploitable through backdoor accounts, malformed HTTP or FTP requests, or cross-site scripting (XSS). This exploit module retrieves stored username and passwords for the webserver login and an additional password that may be used to modify control operations via the web interface.

It should be noticed that the basic issues in achieving the ICS security deal both with the presence of ICS vulnerabilities and availability of the tools that allow identification of these weaknesses. It is not true that advanced, complex attacks can be executed by untrained people. However, it is true that information on vulnerability of the ICS software and control devices is generally available. What is more, the detailed information on the ICS implementation available in tender documents can also be exploit for hostile activity. This means that even moderately educated IT or automation service engineer can identify and exploit the ICS vulnerabilities.

4 Architecture of the ICS cyber security protection system

In response to the threats mentioned above, an advanced system for the ICS protection from cyber threats was developed within the research project *Cyber security provision system for critical infrastructure* (No. DOBR/0074/R/ID1/2012/03) co-sponsored by the Polish National Centre for Research and Development.

The functional architecture of the system, which is shown in Fig. 3a is composed of a set of software modules that can be deployed within the ICS infrastructure in an arbitrary way, i.e.: hub module (HM), which is the main logical component of the system, protection modules (PM), which types and quantity depend on the needs, adaptation modules (AM) that allow joining and integration of external protection components, network modules (NM) that allow monitoring and control of the network infrastructure and engineering access control module (EAM) that allow monitoring and control of all technical service activities. In case of small system all components can be placed on a single computer.

⁴ <https://ics-cert.us-cert.gov/advisories/ICSA-14-051-03B>

⁵ <https://ics-cert.us-cert.gov/alerts/ICS-ALERT-12-020-01>

⁶ <https://ics-cert.us-cert.gov/alerts/ICS-ALERT-12-020-03B>

In large and spatially distributed objects, several computers located depending on the needs can be used for the system deployment. In any case, the system is controlled by a single hub of secure communications.

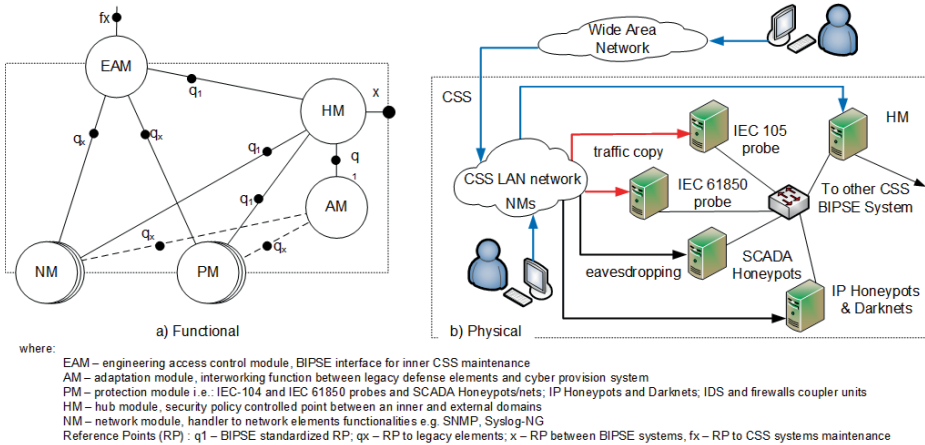


Fig. 3: Architecture of the ICS cyber security system

The ICS cyber security system provides:

- analysis of the network traffic, searching for threats and anomalies;
- detection of malicious actions using specially designed IED-emulating probes;
- correlation of events identified by sensors;
- automated detection and tracking of threats, giving appropriate countermeasures;
- security management of the ICS infrastructure, including both station and technological communications;
- cyber situational awareness of the whole monitored ICS (SIEM⁷ -like).

In particular, the ICS cyber security protection system applies the following security countermeasures shown in Fig. 3b:

- anomaly detection and filtering of management and control IP traffic transferring IEC protocols (IEC 105 and IEC 61850 probes);
- SCADA hardware emulation, Honeypots and Darknets (SCADA Honeypots, IP Honeypots, Darknets);
- monitoring of the status of the protected infrastructure and secure storage of information (HM);
- secure communications with the central SIEM and the user interface (GUI) (HM);

⁷ SIEM - Security Information and Event Management

- advanced access control with the use of Attribute-Based Access Control (ABAC) model and security policies (HM);
- audit and traceability of all management operations and identification of potential unauthorized operations (HM);
- encryption of internal management messages.

Detection of different types of reconnaissance activities are performed by decoy devices. Honeypot SCADA module enables location within the control network additional fictional devices that do not participate in any real control processes. All traffic addressed to such entities is logged that enables the adversary detection as well as identification of types of performed actions. The presented version of the system supports the control protocol IEC 61850 MMS, however it could be easily adapted to support other ones. Any idle IP address (unused by the real devices) can be monitored by Darknet probe. Contrary to other decoy devices, it does not interact with the attacker and only register his activities in hidden way. This prevents from any kind of reconnaissance activities including attacks performed without the knowledge of the network topology.

5 System verification

The cyber security protection system was examined in 3 different environments, i.e. in the laboratory testbed (Fig. 4a) resembling the architecture of power substation, in the Laboratory of Distributed Generation at Institute of Electrical Power Engineering of the Lodz University of Technology (Fig. 4b), as well as in operational power substation (Fig. 4c).

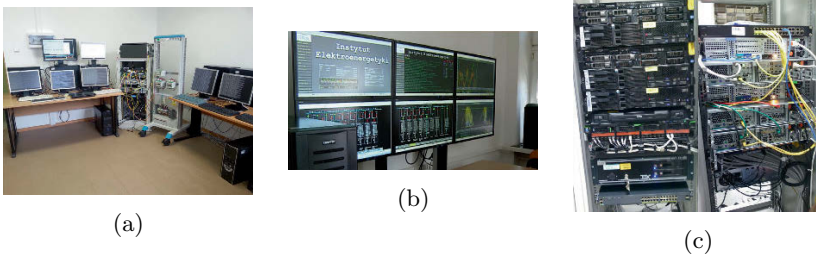


Fig. 4: ICS cyber security system implementation in: (a) laboratory testbed, (b) Laboratory of Distributed Generation, (c) operational power substation

The experiments were designed to verify the ability of the system to detect cyber-attacks and to protect against them, as well as to adjust the sensitivity of the developed probes and decoys.

The efficiency of threat detection was verified by the following tools [7]:

- probes based on Snort and Bro software that are adapted for analysis of the SCADA protocols (e.g. IEC 60870-5-104) in order to detect anomalies in the power control and management systems;
- commercial IDS/IPS probes that were previously purchased and are currently used in the power control and management network;
- standard Honeypots, SCADA Honeypots and Darknets for monitoring and logging of all of the suspicious activities in ICS network;
- Mediation Device developed to normalize the messages obtained from the other security systems and elements;
- SIEM system gathering, analyzing and aggregating information received from above mentioned elements;
- databases gathering the history of power control and management conditions;
- cyber security Visualization and Management System processing data produced in SIEM in real-time;
- engineering access control system for monitoring and control of all technical service activities, including video registration.

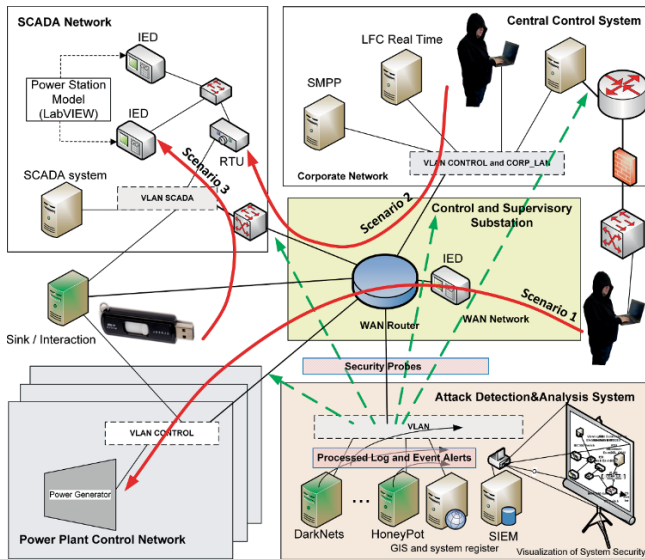


Fig. 5: Test scenarios

The exemplary scenarios (Fig. 5) define the following directions of attacks:

- a) from the Internet and over WAN with the use of unauthenticated and unauthorized measures by intruders;
- b) from the enterprise network, the attacks coming from authorized users of this network who, due to various reasons, attack the power control system;

- c) from the control network by persons who know the effects of the attacks and due to personal and/or external reasons conduct attacks on the infrastructure;
- d) from the control network by users who are not aware of the threats, authorized to resources (e.g. during a software update a malware is installed and transferred along with the useful software).

Let us consider the attack scenario from the Internet (ad. a) with the use of PSTN or from the direction of the enterprise network, in which the firewall protecting the control network access was breached (Fig. 5, scenario 1).

Table 1: Test results for attack scenario 1

(a)	<code>[root@pl.bipse.mut.gen1 ~]#nmap -Pn -p 1-100 --min-rate 1000 172.15.102.104</code>				
	Event Time	Phy Element	CSS	Domain	Msg
	2015-04-01 13:02:27.797	probe-2hnp	si2	mut	Many rejected TCP packets for Ip Addr Pair
	2015-04-01 13:02:23.971	ssin1	si2	mut	TCP packet rejected
(b)	<code>[root@pl.bipse.mut.gen1 ~]#wget test@172.15.102.100 / --ask-password</code> 100%[=====>] 204 -.K/s w 0s 2015-04-01 13:27:42 (10,4 MB/s) - saved 'index.html' [204/204]				
	Event Time	Phy Element	CSS	Domain	Msg
	2015-04-01 13:27:42.589	probe2-hnp	si2	mut	TCP connection terminated
	2015-04-01 13:27:42.589	probe2-hnp	si2	mut	TCP connection terminated
	2015-04-01 13:27:42.108	probe2-hnp	si2	mut	TCP connection set-up
	2015-04-01 13:27:42.108	probe2-hnp	si2	mut	TCP connection set-up

In this case, the symptoms of attack are traces left by the control station environment recognition applications. In this scenario, the attacker poorly acquainted with the environment, has to find out the control station structure, functions performed by the devices, their addresses and implemented protocols. This requires generation of two-directional monitoring traffic from and to the station. Collected features of the observed traffic are then sent toward the attacker. Such actions lead to increase of the traffic volume that could be detected on the routers. Scanning of addresses and/or ports may be detected by SCADA decoys Darknets (Tab. 1.a) and Honeypots (Tab. 1.b), emulating operation of substation.

The test results confirm detection of unauthorized operations on the security elements allowing the security administrator to make an appropriate decision. Some other results of the system verification are available in [7].

6 Conclusions

The positive results of the ICS cyber security protection system validation allowed for its installation in the power station of the Polish Transmission System Operator. Exhaustive tests performed in operational environment confirmed that the system meets all functional requirements. Its specific features like modular and scalable architecture, closed-loop reaction to detected threats, expanded engineering access control subsystem, and lack of negative impact on security and reliability of the protected object allow the system adaptation both to small and large-scale implementations. The ICS cyber security protection system can be also adapted to other critical infrastructure environments, such as fuel or water supply systems.

The advanced concept of the system covers a trusted multi-domain cooperation when the domains share the identified threat information building a cybersecurity situational awareness.

Acknowledgment

This work is based upon research project supported by the Polish Ministry of Science and Higher Education via the Military University of Technology under the contract number 917/2016/WAT.

References

1. Special Security Report, Ukraine Cyber Attack Analysis, RADIFLOW, http://radiflow.com/wp-content/uploads/2015/12/Ukraine_cyber_attack_report.pdf
2. Repository of Industrial Security Incidents Online Incident Database, <http://www.risidata.com/Database>
3. ICS-CERT Monitor Newsletter, October-December 2013, <https://ics-cert.us-cert.gov/monitors/ICS-MM201312>
4. ICS-CERT Monitor Newsletter, May-August 2014, <https://ics-cert.us-cert.gov/monitors/ICS-MM201408>
5. PSE Operator: Requirements for LFC implementation in power plants, Konstancin-Jeziorna, 1-32 (2011)
6. Jarmakiewicz J. et al.: Development of Cyber Security Testbed for Critical Infrastructure, IEEE Explore DOI:10.1109/ICMCIS.2015.7158686
7. Jarmakiewicz J. et al.: Evaluation of the cyber security provision system for critical infrastructure, Journal of Telecommunications and Information Technology, No. 4/2015, 22-29 (2015)
8. Kruczkowski M., Niewiadomska-Szynkiewicz E., Kozakiewicz A.: FP-tree and SVM for Malicious Web Campaign Detection, Intelligent Information and Database Systems, Pt II Book Series: Lecture Notes in Artificial Intelligence, Vol: 9012, pp: 193-201 (2015)
9. Kozakiewicz A., Felkner A.; Kruk T.J.: Critical information infrastructures security, Lecture Notes in Computer Science, vol: 5508 (2009)

Distributed design of sensor network for abnormal state detection in distributed parameter systems

Damian Kowalów and Maciej Patan

Institute of Control and Computation Engineering
University of Zielona Góra
ul. Szafrana 2, 65-516 Zielona Góra, Poland
{d.kowalow, m.patan}@issi.uz.zgora.pl

Abstract. The problem of measurement effort distribution for detection of the abnormal state of distributed parameter system monitored with sensor network is considered. The measurement strategy is formulated in terms of maximizing the power of parametric hypothesis test related to the nominal system state. Then, using communication schemes based on the class of so-called gossip algorithms a computational procedure for optimizing the measurement effort over the sensor network is proposed. Finally, the presented fault detection approach is verified on the example of convective-diffusion process.

1 Introduction

Together with increasing complexity and quality demands imposed on the modern control systems, the Fault Detection and Isolation (FDI) for dynamical systems become very important area of research, especially in the context of real engineering applications. In spite of the fact, that there is a vast amount of contributions developed for lumped parameters systems [3, 6, 20], there still just a few approaches dedicated to the distributed-parameter systems (DPSs) exists. Most often, it is impossible to observe the system states over the entire spatial domain. Going further, it is required to properly select the observational data and design the sensor network in terms of spatial configuration of its nodes. Although some dedicated techniques of practical sensor placement techniques has been developed for stationary case [9, 29, 18, 22, 14, 8, 7], scanning sensor networks [18, 28, 12, 15] or moving observations [24, 23, 5, 29, 17, 31] almost all are based on the centralized approaches [27, 26]).

One of the main issues during developing applications for DPSs is determination of proper relation between good performance of system and the strategy of gathering the measurement data in order to achieve the high reliability of diagnosis. Here, the centralized approach based on application of the experimental design theory for sensor allocation developed for stationary sensors [18], scanning and mobile sensor networks [13, 18, 32, 19, 11, 25] can be further extended towards

the fully distributed computational scheme. Additionally some techniques can be successfully used also in optimum design for neural networks[10].

This work focus on the distributed optimization of experimental effort over the sensor network for fault detection in DPSs. All sensors are assumed to take measurements continuously in time and are in located at a given finite set of sites. Each sensor node has a non-negative weight assigned to it, representing the proportion of total measurement effort spent at this node. That way, we are able to reduce the complexity of observational system and determine the most informative sensor nodes, which are further used to store data. This solution is somewhat similar to the classical concept of optimum experimental design theory for lumped systems [1, 33].

The delineated approach can be easily tailored to the framework of so-called *gossip algorithms* in which each node communicates with no more than one neighbor at each time instant [13]. In effect, the resulting exchange-type algorithm for sensor scheduling is working in a fully decentralized way and is very easy to implement.

2 Abnormal state detection problem

Parameter estimation is one of most popular techniques of fault detection [6]. It is especially useful in situations when system parameters have a strong physical interpretation coming from proper modelling of process variables.

Let $y = y(x; t; \theta)$ is a scalar state of spatio-temporal object in point $x \in \Omega \subset \mathbb{R}^d$ and time $t \in T = [0; tf], tf < \infty$. Unknown constant θ is m - dimensional parameter vector. Additionally N denotes observations provided by ℓ stationary point-wise sensors

$$z_m^j(t) = y(\chi^j, t; \theta) + \varepsilon(\chi^j, t), \quad x^j \in X, \quad t \in T, \\ j = 1, \dots, n \quad (1)$$

where $z_m^j(t)$ is the scalar output, χ^j stands for an element selected from among a given *a-priori* set of sensor locations $X = \{x^1, \dots, x^\ell\}$ and $\varepsilon(\chi^j, t)$ denotes the measurement noise which is assumed that is zero-mean, Gaussian, spatial uncorrelated and white [30], i.e. $E[\varepsilon^j(t)] = 0$ and $\text{var}(\varepsilon^j(t)) = \sigma^2$ and $\sigma > 0$ is standard deviation of the measurement noise.

The unknown parameter vector θ is estimated through minimization of LS criterion

$$\mathcal{J}(\theta) = \sum_{j=1}^N \int_T |z^j(t) - \hat{y}(x^j, t; \theta)|^2 dt, \quad (2)$$

where $\theta \in \Theta_{\text{ad}}$ and Θ_{ad} is the set of admissible parameters and $\hat{y}(\cdot, \cdot; \theta)$ denotes the system model response corresponding to a given θ . Additionally, vector $\hat{\theta}$ minimizing $\mathcal{J}(\theta)$ stands for the estimate of the true value of θ^* .

Fault diagnostics in most examples consists of comparison of estimates with corresponding known nominal values to find statistically significant perturbations. Then, to detect abnormal situation, some thresholding techniques can be applied [18, 16, 32, 19], with assumption that θ^* is the nominal value of θ corresponding to normal system performance. This way it is possible to test a simple statistical hypothesis imposed on the system parameters $H^0 : \theta = \theta^*$.

Taking into account the generalization of likelihood function [4]:

$$\mathcal{L}(z|\theta) = \left(\frac{1}{2\pi\sigma^2} \right)^{N/2} \exp \left(-\frac{1}{2\sigma} \mathcal{J}(\theta) \right), \quad (3)$$

with $\Theta_0 = \{\theta \in \Theta_{\text{ad}} : \theta = \theta^*\}$ the generalized log-likelihood ratio takes the form:

$$\lambda(z) = 2 \ln \frac{\sup_{\theta \in \Theta_{\text{ad}}} \mathcal{L}(z|\theta)}{\sup_{\theta \in \Theta_0} \mathcal{L}(z|\theta)} = \mathcal{J}(\tilde{\theta}) - \mathcal{J}(\hat{\theta}), \quad (4)$$

where

$$\hat{\theta} = \arg \min_{\theta \in \Theta_{\text{ad}}} J(\theta), \quad \tilde{\theta} = \arg \min_{\theta \in \Theta_0} J(\theta). \quad (5)$$

Generalized log-likelihood ratio test is very popular in statistic and if null hypothesis H^0 is not rejected, the sequence $\lambda(z)$ for $N \rightarrow \infty$ is weakly convergent to χ^2 random variable on m degrees of freedom[4] (Thm. 3.6.1, p 55).

Therefore, it is possible to compare the value of $\lambda(z)$ with threshold k_γ where γ representing a a fixed range of model uncertainty. Threshold level is obtained from chi-squared distribution with m degrees of freedom, then fault detection is adequate to:

$$S = \begin{cases} S^1 & \text{if } \lambda(z) \geq k_\gamma \quad (\text{reject } H^0), \\ S^0 & \text{if } \lambda(z) < k_\gamma \quad (\text{accept } H^0), \end{cases} \quad (6)$$

where k_γ is ' γ -quantile' of the distribution.

If H^0 will be rejected it indicates detection of unexpected state of system. Separate possibilities could be as a result: type I error - incorrect rejection of a true null hypothesis and type II error where is observed a failure to reject a false null hypothesis. In faults diagnostic, there is a connection between these errors and probability of a false alarm and missed detection respectively.

3 Optimal measurement problem

It can be shown that the power of the presented hypothesis test (i.e. the probability of not committing the Type II error) can be increased by taking a large number of measurements N or, alternatively, by maximizing the D-optimality criterion [16]:

$$\Psi(M) = -\log \det(M), \quad (7)$$

where M is defined as so-called *average* Fisher Information Matrix[12]

$$M = \frac{1}{Nt_f} \sum_{j=1}^N \int_0^{t_f} g(\chi^j, t) g^\top(\chi^j, t) dt, \quad g(x, t) = \left(\frac{\partial(x, t; \theta)}{\partial \theta} \right)_{\theta=\theta^0}^\top. \quad (8)$$

So-called *sensitivity* vector $g(x, t)$ describe what is influence of the each parameter for system behavior, where θ^0 is a prior estimate to the unknown parameter vector θ .

Sensor location optimality problem is defined as

$$\Psi[M(\chi^1, \dots, \chi^N)] \rightarrow \min, \quad (9)$$

where $\chi^j, j = 1, \dots, N$ is belong to set X .

It possible to reformulate problem to operate on locations (*design*) x^1, \dots, x^ℓ in place of χ^1, \dots, χ^ℓ with r_1, \dots, r_ℓ as the numbers of replicated measurements and p_1, \dots, p_ℓ as a their weights. Therefore

$$\xi_N = \{(x^1, p_1), (x^2, p_2), (x^\ell, p_\ell)\}, \quad (10)$$

where $p_i = r_i/N$, $N = \sum_{i=1}^\ell r_i$ is *exact design* of experiment, p_i is proportional of observations performed at x^i locations of sensors.

Therefore FIM:

$$M(\xi_n) = \sum_{i=1}^\ell p_i \frac{1}{t_f} \int_0^{t_f} g(x^i, t) g^\top(x^i, t) dt, \quad (11)$$

and $\sum_{i=1}^\ell p_i = 1$ to receive proper probability distribution on X .

Using (10) it is possible to redefine an optimal design solution to the optimization problem

$$\xi^* = \arg \min_{\xi \in \Xi(X)} \Psi[M(/xi)], \quad (12)$$

where $\Xi(X)$ is set of all probability distributions on $X = \{x^1, \dots, x^\ell\}$.

4 Distributed sensor selection optimization problem

Taking into account the problem where sensor nodes have fixed spatial positions a computational algorithm can be proposed using the mapping $\mathcal{T} : \Xi(X) \rightarrow \Xi(X)$ where

$$\mathcal{T}\xi = \left\{ \left(x^1, p_1 \frac{\phi(x^1, \xi)}{m} \right) \dots, \left(x^\ell, p_\ell \frac{\phi(x^\ell, \xi)}{m} \right) \right\}, \quad (13)$$

and design ξ^* is D-optimal if it is a fixed point of the mapping \mathcal{T} , i.e. $\mathcal{T}\xi^* = \xi^*$. This idea leads to decentralized configuration algorithm for sensor network which is distributed generalization of [21] for the classical optimum experimental design problem consisting in iterative computation of D-optimum design on finite set which was extended also in [30, 13].

Let $r = 0, 1, 2 \dots$ denotes discrete time index, which partition the continuous configuration time axis for time intervals $Z_r = [z_{r-1}, z_r)$ then it is possible to map \mathcal{T} to iteratively improve the experimental effort distribution. Global information matrix $M(\xi)$ cannot be calculated independently of the other network nodes. Therefore (11) gives that information matrix is a weighted average of local information matrices:

$$M_i = \frac{1}{t_f} \int_0^{t_f} g(x^i, t)g^T(x^i, t) dt. \tag{14}$$

This means that algorithm is related to problem of distributed averaging on a network[34, 2].

One well know technique for distributed averaging is so-called *gossip* scheme. In its classical variation it is assumed that at the k reconfiguration time slot the i -th sensor contacts some neighbor node j with probability Q_{ij} and the pair $(i-j)$ is randomly and independently selected. During each time node value stored in the single node convergence to average of all nodes. Assuming $M_k(\xi^{(r)})$ will be estimate of global FIM maintained by k -th sensor at time slot Z_r , then

$$M_k(\xi^{(r)}) \leftarrow \frac{p_i M_i(\xi^{(r)}) + p_j M_j(\xi^{(r)})}{p_i + p_j}, \quad k \in \{i, j\} \tag{15}$$

where \leftarrow is an update operator. To avoid situation when weights trends to zero value during time slots can be introduced concept of running consensus

$$M_i(\xi^{(r)}) \leftarrow \frac{r-1}{r} M_i(\xi^{(r)}) + \frac{1}{r} M_i, \tag{16}$$

where first term enforces consensus among nodes (average information form network) and second accounts for the increase in total contribution of the local node. Idea of the presented approach is derived in the Algorithm 1.

Algorithm 1 Distributed optimization of experimental effort. Indexes i and j denote, respectively, data from local repository and obtained from neighbour. Function *EXCHANGE* is responsible for both sending and receiving data to/from connector neighbor (order depending on who initiated communication)

- 1: **procedure** EXCHANGE_PROTOCOL
 - 2: EXCHANGE($M_j(\xi)$, $M_i(\xi)$) ▷ sends and receives
 - 3: EXCHANGE(p_i and p_j) ▷ FIM weights
 - 4: $p \leftarrow p_i + p_j$ ▷ store old weights for normalization
 - 5: Update $M_i(\xi) \leftarrow (p_i M_i(\xi) + p_j M_j(\xi)) / (p_i + p_j)$
 - 6: Calculate $\phi(x^k, \xi) = \text{tr}[M_i^{-1}(\xi)M_k]$, $k \in \{i, j\}$
 - 7: Update $p_k \leftarrow p_k \phi(x^k, \xi)$
 - 8: $p_i \leftarrow p \cdot p_i / (p_i + p_j)$ ▷ normalization p_i 's sum up to 1
 - 9: Update $M_i(\xi) \leftarrow \frac{r-1}{r} M_i(\xi) + \frac{1}{r} M_i$
 - 10: **end procedure**
-

At first step ($r = 0$) and each node starts with global FIM estimate $M_i(\xi^{(r)})$ on a basis of local information matrix M_i with non-zero weights to exclude rejection of potentially good information point, next during asynchronous configuration time slots Z_r , nodes start communication. Consequently optimal solution is compared with independent weight update for each network node and stochastic convergence of gossip algorithms[34] but it need to admit that initial sensor configuration have big influence on convergence rate.

5 Simulation example

For better understanding of the presented approach, the problem of sensor configuration for parameter estimation and fault detection process of air pollutant transport over a given urban area is presented. Inside normalized spatial area $\Omega = (0, 1)^2$ active pollution source is present. Pollutant spatial concentration $y = y(x, t)$ is observed over the normalized time interval $T = (0, 1]$ and is presented by advection-diffusion equation:

$$\begin{aligned} \frac{\partial y(x, t)}{\partial t} + \nabla \cdot (v(x, t)y(x, t)) \\ = \nabla \cdot (\alpha(x)\nabla y(x, t)) + f(x), \quad x \in \Omega \end{aligned} \quad (17)$$

subject to the boundary and initial conditions:

$$\frac{\partial y(x, t)}{\partial n} = 0, \text{ on } \Gamma \times T, \quad (18)$$

$$y(x, 0) = y_0, \text{ in } \Omega, \quad (19)$$

where $y(x, t)$ is pollutant concentration, $v(x, t)$ is wind velocity field $\alpha(t)$ is spatial-varying diffusion coefficient, and f denotes pollutant sources $f(x) = 50 \exp(-50||x - c||^2)$ located at the point $c = (0.3, 0.3)$. Partial derivative of y with respect to the outward normal to the boundary Γ is represented by $\partial y/\partial n$. The mean spatio-temporal changes of wind velocity field over the area were approximated by $v(x, t) = (2(x_1 + x_2 - t), x_2 - x_1 + 2t)$. Assumed functional form of the spatial-varying diffusion coefficient is $a(x) = \theta_1 + \theta_2 x_1 x_2 + \theta_3 x_1^2$. Main goal of the research is to identify intensity of pollutant emission from source and detection of significant changes of turbulent diffusion coefficient through θ_1 parameter changes. Estimation of parameters are based of measurement data from monitoring stations and to simplify fault simulation a several scenario was used which are described below. As an application implementation MATLAB program was written and simulated using PC station with Intel Core i5 processor (2.5GHz, 12GB RAM) running Windows 10. Vector of base parameters describing system was $\theta_1^0 = 0.05$, $\theta_2^0 = 0.01$, $\theta_3^0 = 0.005$. Calculations were done with a finite element method for spatial mesh composed of 168 triangles, 132 nodes. In the presented example, it is assumed that the sensors are located at the points of triangulation mesh excluding points lying on the boundary, cf. Fig. 2). The goal is to determine the weights for all $N = 132$ possible positions of sensors nodes

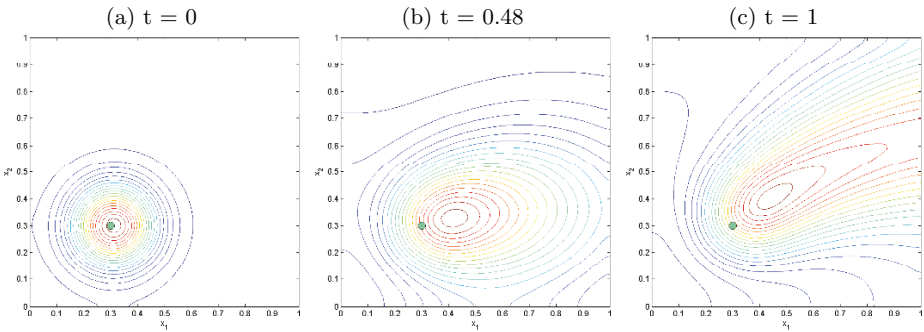


Fig. 1. Temporal changes in the wind velocity field and pollutant concentration (green point – pollution source)

which leads to D-optimum least-squares estimates of parameters θ . Experiment assumed that communication between each pair of sensors is possible without taking into account the distance (the graph is fully connected). In Fig. 2 are presented two stages of algorithm where in Fig. 2(a) is presented one of initial state and in (b) D-optimal solution.

Once the observation strategy is established it is possible to make a hypothesis test (6). The threshold limit was chosen at level 95%, which corresponds to $k_\lambda = 3.8$ and give only 5% chance to achieve inappropriate results. Also it should be mentioned that rejection hypothesis is not conclusive from statistical point of view. Examples of simulating failures are presented in Fig. 3 where different scenarios of both the abrupt and incipient faults were investigated. In three from four examples failure was property recognized, only in a case of very small deviation of θ_1 from nominal value (fig. 3 (b)) hypothesis that failure occurred was rejected with 95% but is acceptable with 89% level ($k_\lambda = 2.5$). In this kind of problems need to be taken into account two-stage decision system or different approach.

6 Concluding remarks

In the presented work, the problem of optimal weight selection representing the importance of information gathered from sensor nodes with fault identification using log-likelihood ratio test has been presented. Main point was to adequate adaptation of problem to distributed systems. With very rapid development routines for distributed information fusion designed for sensor, peer-to-peer or wireless ad-hoc networks it is possible to implement presented ideas in real system also taking into account simplicity and time of computation. Further researches will be focused on more complex monitoring systems i.e. scanning or mobile sensor networks and extension to the iterative learning control schemes dedicated for the fault detection.

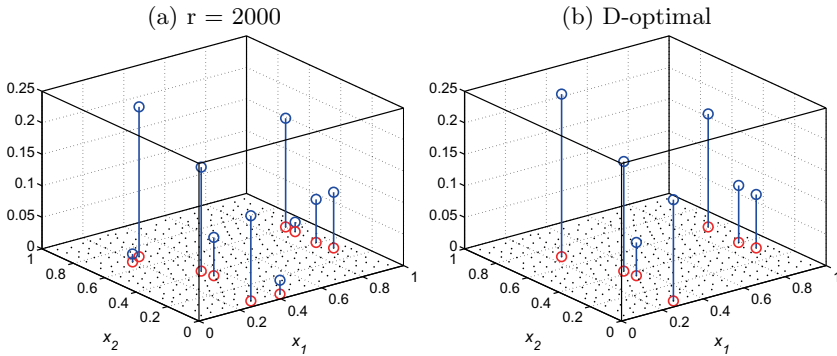


Fig. 2. Allocation of active sensors (red circles) with experimental effort (blue stem plot) in consecutive stages of network configuration (a) and final D-optimal configuration ($\ln(\det(M)) = 17.1263$) (b).

References

1. A. Atkinson, A. Donev, and R. Tobias. *Optimum experimental designs, with SAS*, volume 34. Oxford University Press, 2007.
2. S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE/ACM Transactions on Networking (TON)*, 14(SI):2508–2530, 2006.
3. L. H. Chiang, R. D. Braatz, and E. L. Russell. *Fault detection and diagnosis in industrial systems*. Springer Science & Business Media, 2001.
4. G. C. Goodwin and R. L. Payne. *Dynamic system identification: experiment design and data analysis*. 1977.
5. A. Jeremic and A. Nehorai. Landmine detection and localization using chemical sensor array processing. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 47(11):3185, 1999.
6. J. Korbicz and J. M. Kościelny. *Modeling, diagnostics and process control: implementation in the diaster system*. Springer Science & Business Media, 2010.
7. D. Kowalów and M. Patan. Optimal sensor selection for model identification in iterative learning control of spatio-temporal systems. In *Methods and Models in Automation and Robotics (MMAR), 2016 21st International Conference on*, pages 70–75. IEEE, 2016.
8. D. Kowalów, M. Patan, W. Paszke, and A. Romanek. Sequential design for model calibration in iterative learning control of dc motor. In *Methods and Models in Automation and Robotics (MMAR), 2015 20th International Conference on*, pages 794–799. IEEE, 2015.
9. A. Nehorai, B. Porat, and E. Paldi. Detection and localization of vapor-emitting sources. *IEEE Transactions on Signal Processing*, 43(1):243–253, 1995.
10. K. Patan, M. Patan, and D. Kowalw. Optimum training design for neural network in synthesis of robust model predictive control. In *55th IEEE Conference on Decision and Control - CDC 2016*, pages 3401–3406, Las Vegas, USA, 2016. IEEE Explore.
11. M. Patan. A parallel sensor scheduling technique for fault detection in distributed parameter systems. In *Euro-Par 2008-Parallel Processing*, pages 833–843. Springer, 2008.

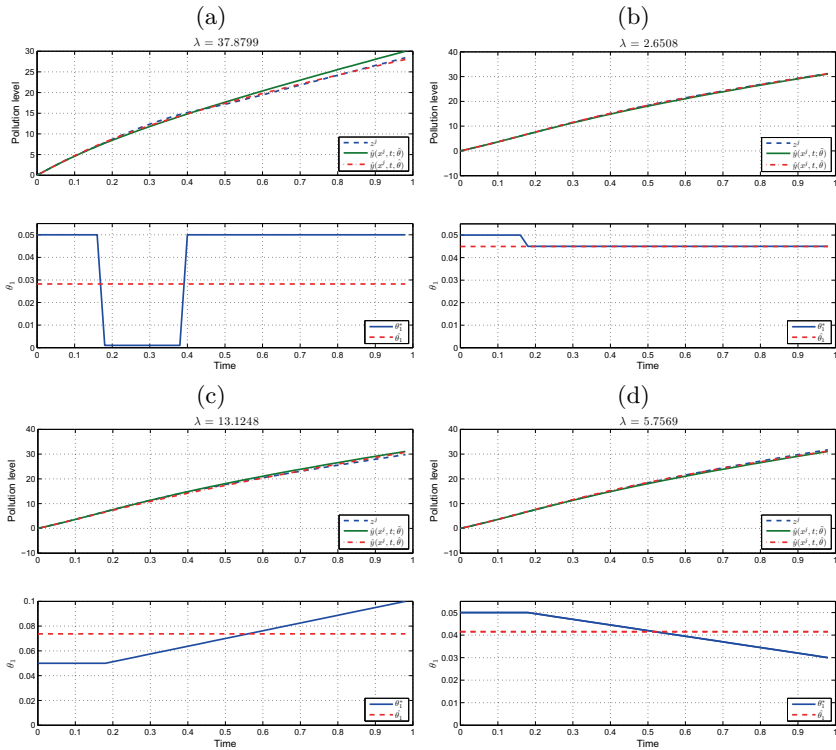


Fig. 3. Failure simulation for θ_1 with different strategies and levels of failure (a)- temporary, (b) - constant, (c) - increasing, (d) - decreasing.

12. M. Patan. Distributed scheduling of sensor networks for identification of spatio-temporal processes. *International Journal of Applied Mathematics and Computer Science*, 22(2):299–311, 2012.
13. M. Patan. *Optimal sensor networks scheduling in identification of distributed parameter systems*, volume 425. Springer Science & Business Media, 2012.
14. M. Patan and D. Kowalów. Robust sensor scheduling via iterative design for parameter estimation of distributed systems. In *Methods and Models in Automation and Robotics (MMAR), 2014 19th International Conference On*, pages 618–623. IEEE, 2014.
15. M. Patan and D. Kowalów. Distributed configuration of sensor network for fault detection in spatio-temporal systems. In *Journal of Physics: Conference Series*, volume 783, pages 1–12. IOP Publishing, 2017.
16. M. Patan and K. Patan. Optimal observation strategies for model-based fault detection in distributed systems. *International Journal of Control*, 78(18):1497–1510, 2005.
17. M. Patan, C. Tricaud, and Y. Chen. Resource-constrained sensor routing for parameter estimation of distributed systems. In *Proc. 17th IFAC World Congress*, 2008.

18. M. Patan and D. Ucinski. Optimal activation strategy of discrete scanning sensors for fault detection in distributed-parameter systems. In *Proceedings of the 16th IFAC world congress, Prague, Czech Republic*, pages 4–8, 2005.
19. M. Patan and D. Uciński. Configuring a sensor network for fault detection in distributed parameter systems. *International Journal of Applied Mathematics and Computer Science*, 18(4):513–524, 2008.
20. R. J. Patton, P. M. Frank, and R. N. Clark. *Issues of fault diagnosis for dynamic systems*. Springer Science & Business Media, 2013.
21. A. Pázman. *Foundations of optimum experimental design*, volume 14. Springer, 1986.
22. N. Point, A. V. Wouwer, and M. Remy. Practical issues in distributed parameter estimation: Gradient computation and optimal experiment design. *Control Engineering Practice*, 4(11):1553–1562, 1996.
23. B. Porat and A. Nehorai. Localizing vapor-emitting sources by moving sensors. *Signal Processing, IEEE Transactions on*, 44(4):1018–1021, 1996.
24. E. Rafajłowicz. Optimum choice of moving sensor trajectories for distributed-parameter system identification. *International Journal of Control*, 43(5):1441–1451, 1986.
25. A. Romanek, M. Patan, and D. Kowalów. Decentralized scheduling of sensor networks for parameter estimation of spatio-temporal processes. *Advanced and Intelligent Computations in Diagnosis and Control*, 386:145, 2015.
26. Z. Song, Y. Chen, C. R. Sastry, and N. C. Tas. *Optimal observation for cyber-physical systems: a fisher-information-matrix-based approach*. Springer Science & Business Media, 2009.
27. C. Tricaud and Y. Chen. *Optimal mobile sensing and actuation policies in cyber-physical systems*. Springer Science & Business Media, 2011.
28. C. Tricaud, M. P. Dariusz, U. Yang, and Q. Chen. D-optimal trajectory design of heterogeneous mobile sensors for parameter estimation of distributed systems. In *2008 American Control Conference*, pages 663–668. IEEE, 2008.
29. D. Uciński. Optimal selection of measurement locations for parameter estimation in distributed processes. *International Journal of Applied Mathematics and Computer Science*, 10(2):357–379, 2000.
30. D. Uciński. *Optimal measurement methods for distributed parameter system identification*. CRC Press, 2004.
31. D. Uciński. Sensor network scheduling for identification of spatially distributed processes. *International Journal of Applied Mathematics and Computer Science*, 22(1):25–40, 2012.
32. D. Uciński and M. Patan. Sensor network design for the estimation of spatially distributed processes. *International Journal of Applied Mathematics and Computer Science*, 20(3):459–481, 2010.
33. É. Walter and L. Pronzato. Qualitative and quantitative experiment design for phenomenological models survey. *Automatica*, 26(2):195–213, 1990.
34. L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

Application of data driven methods in diagnostic of selected process faults of nuclear power plant steam turbine

Karol Kulkowski*, Michał Grochowski*, Anna Kobylarz*, Kazimierz Duzinkiewicz*

*Gdańsk University of Technology

Abstract. Article presents a comparison of process anomaly detection in nuclear power plant steam turbine using combination of data driven methods. Three types of faults are considered: water hammering, fouling and thermocouple fault. As a virtual plant a nonlinear, dynamic, mathematical steam turbine model is used. Two approaches for fault detection using one class and two class classifiers are tested and compared.

Keywords: steam turbine, fault detection, data driven methods, K-means clustering, PCA, SVM.

1 Introduction

The safe operation of a steam turbine, especially when it is a part of critical infrastructure system such as Nuclear Power Plant (NPP) [1], is very important. It is also essential to prevent and decrease the hazard of faults occurring within it. Extremely important in this case is continuous monitoring of such element. It should allow for fast enough detection of potential fault and to prevent its occurrence [2]. It also enables to plan the renovation and proper exploitation, preventing increasing the occurred fault.

Detailed description of the steam turbine and its models was presented in [3]. Because of the high risk caused by the nature of NPP, the steam turbine is exposed to faults and abnormalities what can result in permanent fault or steam turbine stop. Exemplary and representative faults of steam turbine are described in Sect. 2. The paper proposes the two data driven approaches for the faults detection. In the first approach it is well known Principal Component Analysis (PCA) method preceded by data clusterization in order to approximate the nonlinear dependencies on data by the linear statistical tool. In the second approach the Support Vector Machines (SVM) method is used, however due to large number of state variables to analyze, SVM is applied after the data size reduction using PCA.

2 Problem characterization

Steam turbine serves to convert energy contained in heat of fresh steam generated in steam generator to mechanical energy translating into a turbine shaft. The conversion is possible thanks to the expansion process occurring along the steam turbine. The example of steam turbine used in research presented in this paper was shown in Fig. 1.

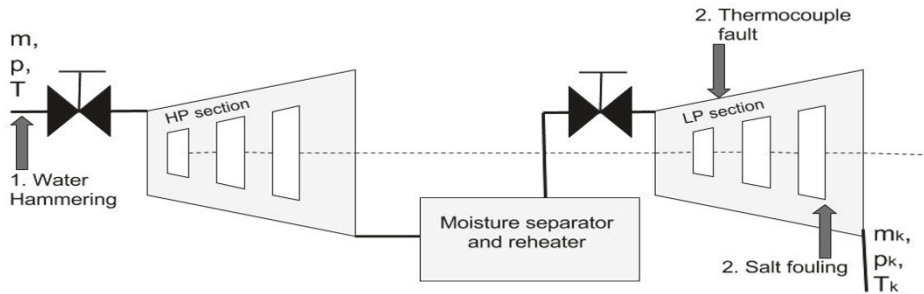


Fig. 1. NPP steam turbine scheme and faults occurrence points

Steam turbine operates on fresh steam defined by its properties such as: steam mass flow (m), steam pressure (p) and steam temperature (T). Steam with such properties passes along steam turbine by k series set of turbine stages. Each stage consists of blades on the rotor combined with turbine shaft. Steam passing through the stage causes the movement of the rotor by colliding with the blades. It simultaneously causes the drop of steam pressure and speed. Stages are arranged into sections based on the design parameters and conditions of steam it was designed for. Three main sections can be specified: High Pressure (HP) section, Intermediate Pressure (IP) section and Low Pressure (LP) section. In case of this paper a turbine consisting of HP and LP sections is used. In order to improve steam properties, what leads to improvement of turbine efficiency, a moisture separator and steam reheater can be used. Both of them can be placed between sections, e.g. in this case between HP and LP sections. During steam turbine work it is exposed to conditions which could lead to faults. Some of potential faults are presented in this paper and enlisted in Table 1., based on [4, 5].

In this paper we consider the three representatives of these faults: water hammering, salt deposition and finally the malfunction of measuring device - the thermocouple. Water hammering is a phenomenon occurring in steam turbine when properties of steam on the inlet of the turbine are disrupted in such a way that steam temperature significantly drops by at least 50 Kelvins. Such behavior can be caused by faulty work of a steam generator. In order to analyze possibilities of detecting such kind of fault it was simulated by accordingly progressive decreasing the steam turbine input temperature. The salt deposition

Table 1. List of considered faults

<i>Fault</i>	<i>Element</i>	<i>Symptom</i>	<i>Potential effect</i>
Water hammering	Inlet of section	Decrease of temperature - more than 50K	Decrease of turbine efficiency; may cause corrosion and turbine damage
Salt deposition	Turbine stage(s)	Reduction of turbine cross-section; increase of pressure in stages	Increase of tension inside stage; may cause mechanical damage
Thermocouple fault	Any temperature measurement	Measurement value exceeds the admissible error	Neglect of critical fault detection; errors in power calculations (if measurement used in calculations)
Physical damage	Metal elements of steam turbine	Increase of fresh and secondary steam temperature	Decreasing of durability metal may cause physical damage
Steam generator fault	Turbine inlet	Decrease of steam temperature on the inlet	Decrease of steam turbine efficiency
Valve damage	Valve	Decrease of pressure and mass flow rate after valve	Decrease of turbine efficiency
Blade damage	Rotor blade	Steam properties disturbance	Leads to physical damage of blade, rotor

can occur along whole steam turbine as well as along some of the stages. The most exposed for the salt deposition phenomenon are few last stages of the LP section. The salt deposition is caused by improperly balanced steam properties in the particular sections, namely it occurs when the steam passing through is very humid. Salt deposited in stage causes the decrease of the stage diameter what reduces turbine efficiency. With the smaller diameter of stage the pressure is increased what in a long term can affect the mechanical durability. This fault manifests by increasing the pressure and increasing the mass flow rate on the outlet of the stage and in such a way it is simulated within the last LP section stage, during the tests. The third fault analyzed in this paper is malfunction of the thermocouple. Obviously it does not affect the process in the steam turbine itself, but false measurements device readings can lead to improper steam turbine operating and control. The fault of thermocouple is considered when the

error of the measurement is exceeding the admissible error provided by manufacturer in specification. It was simulated by adding a measurement noise on the temperature values on the outlet of second LP section stage. The places within the steam turbine when the described faults are simulated are indicated in Fig. 1. Considered faults are enlisted with its symptoms potential, potential effects and elements it concern (see Table 1) as first three from the top.

For the purposes of the experiments carried out and presented in this paper a nonlinear mathematical model of steam turbine 4CK465 was used based on [6, 7]. The model includes high and low pressure parts of the steam turbine, that consist of 10 and 6 stages, respectively. Also the moisture separator and reheater placed between sections are included in the model. The model allows to gain 111 values of variables such as: pressure, mass flow rate, temperature, power, enthalpy drop. For the purpose of this paper it was used both for generating the data of normal steam turbine operating as well as for simulating the faults of the steam turbine. For here presented experiments the data from 37 state variables were utilized. The data contains temperature and pressure values at each of 16 stages with additional information considering temperature and pressure on steam turbine inlet, temperature and pressure on the outlet of the reheater and temperature on the LP section outlet.

In case of data driven methods, measuring noise can affect the results, hence it should be taken into consideration. In this paper simulation data were burdened with the noise (with maximal deviation of 0.35 percent of variable nominal value) and this was taken under account during defining the PCA confidence limit.

3 Data driven methods - fault detection

In case of steam turbine working as a part of the NPP it is especially important to avoid abnormal states which could cause the fault. Hence, it is very important to detect the process abnormalities as early as it is possible in order to give the plant operators time to avoid the serious damage, by changing the way of process operating or shutting down the process. Such systems need models of occurring processes: theory driven physical modeling or data driven. Theory driven modeling needs a priori very deep knowledge about the system. As opposed, in data driven approach the models are built using large amount of historical process data. This is especially useful for the systems where measurements can be easily taken, while the relationship between these measurements may only be described using complex, hard to identify and uncertain mathematical equations. The choice of measured variables is important and has to take into account the purpose of the modeling. However, it is not easy especially in cases when there are many often redundant measurements leading to so called data flood. Hence, the quantitative data driven tools from the group of multivariable statistic [9], [10] neural networks [11] or more general machine learning can be very useful. In this paper we analyse two approaches for detecting selected process anomalies. The first approach is to use the most popular one class classifier, the PCA, e.g. [12]. Due to the fact that the processes taking place within the steam turbine are

nonlinear, while the PCA is able to properly detect only the linear dependencies, the measurements are firstly preprocessed. The data are split into the clusters in such a way that they can be described by linear relations. In order to do that the K-Means Clustering (k-means) was used [13]. As a result data was assigned to one of two clusters. Next, for each of the obtained clusters, the PCA models are established. These models represent the data gathered from the steam turbine operating in wide range of varying inputs under no fault conditions. As the results of k-means and PCA methods, two models with 4 principal components (*PCs*) for cluster 1 and 5 PCs for cluster 2 were obtained. Squared Prediction Error (*SPE*) and Hotelling (T^2) measures are used for indicating the faults. The resulting measures are selected based on euclidean distances between tested data and centroid locations of each cluster. For tested sample closer to the centroid of one of two clusters the measure coming from PCA model built based on this cluster is selected.

In the second approach we use nonlinear two class classifier, Support Vector Machines (SVM). Because the fact that the number of the analyzed variables is quite large (37) and that information they carry is redundant, they are preprocessed by using PCA in order to reduce their size and to highlight the most important features. In order to detect described process anomalies (water hammering, salt deposition and finally the malfunction of measuring device), the 3 SVM models were build and trained to indicate one particular fault. As the training data was reduced by PCA, 5 most important features were used. The outputs from the SVM models should be '1' when the turbine is in its normal (healthy) state, and '1' if the particular SVM recognizes the fault it was learned. Simple logic operation on the SVM's outputs finishes detection process.

4 Results

In the first approach using PCA models, the data representing healthy state of the steam turbine operating over changing input conditions (Fig. 2.) are used to build the diagnostic models. In order to better fit the linear PCA model into the nonlinear data, they were divided into two clusters by k-means clustering method. Figure. 3. illustrates the PCA models quality with respect to the training data. Next, these models were used to detect process faults within the turbine. During the diagnostic phase, the data are compared against the Euclidean distances from the centers of gravities of the determined clusters. The diagnostic measures are calculated by the PCA model describing the cluster with the smaller distance. The results showing the effect of faults detection process are presented on Figs. 3.-6. *SPE* and T^2 measures in these figures were divided by their upper confidence limits (SPE_{Limit} and T^2_{Limit}), hence exceeding the '1' value means that such data do not belong to this model. Water hammering is easily and early detected by both *SPE* and T^2 measures (Fig. 4.), while the salt fouling and malfunction of the thermocouple is indicated only by *SPE*. In this experiment, the confidence limit is assumed to be 0.95 and there are 4 PCs for

the model describing data in cluster 1, and 5 PCs for the second cluster.

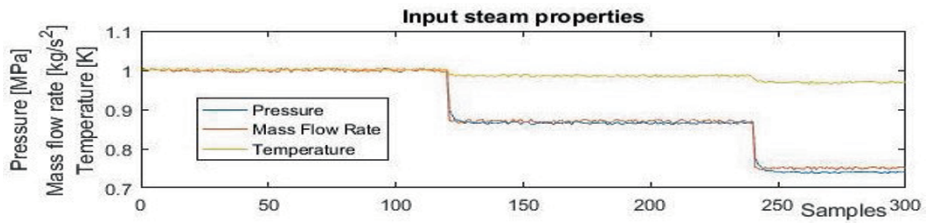


Fig. 2. Input steam properties

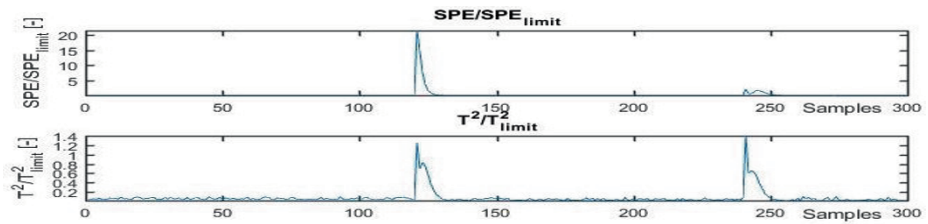


Fig. 3. Training data using k-means-PCA approach

In the second approach the fault detection was performed using SVM based on the 5 of 37 most important principal components generated by PCA method. Each of the SVM models was trained to recognize one particular fault. The results are presented in Figs. 7.-9. All the simulated faults are detected and recognized very clearly and almost immediately. Preprocessing the data by PCA made the symptoms of faults occurred more 'visible' what significantly improved the SVM classification. SVM models utilizes Gaussian Radial Basis Function as a kernel, with the kernel width 1.3096 and soft margin 100.

5 Summary

The paper compares two different approaches for fault detection in the steam turbine. PCA preprocessed by k-means clustering represents the one-class classification approach. In this case process data representing 'normal' behavior of the plant is described by the PCA model and any deviation from them is treated as a fault. In general, the more the process deviates from the 'normal' one the more the SPE and T^2 measures grow what might suggest the size of the fault, however it is not always the rule. On the other hand, PCA is not able to distinguish

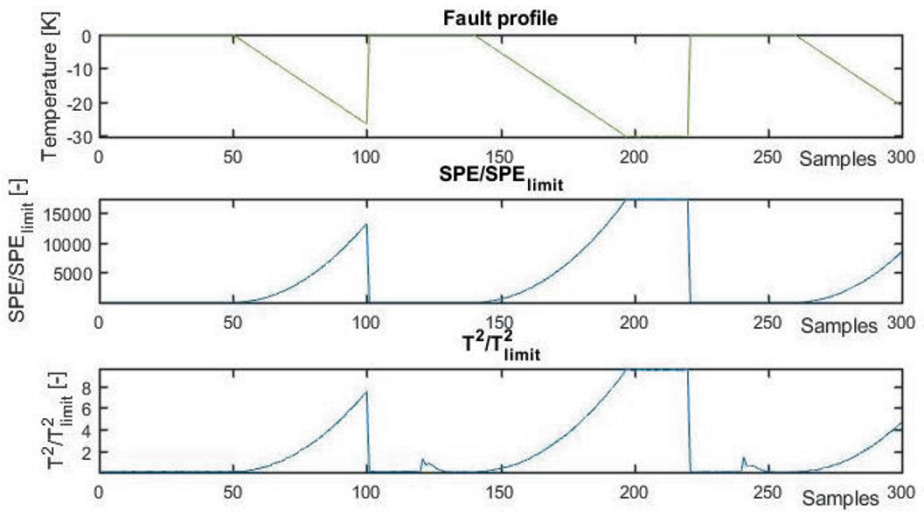


Fig. 4. Water Hammering detection using k-means-PCA approach

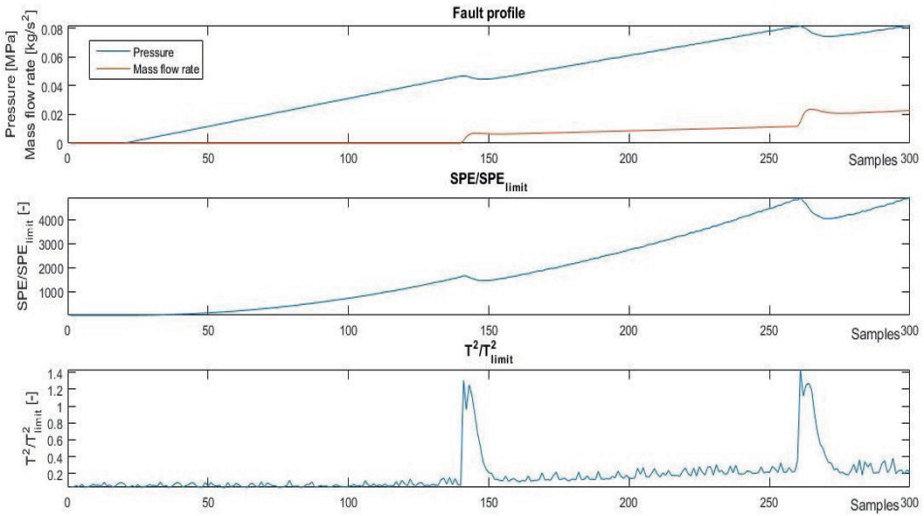


Fig. 5. Salt fouling detection using k-means-PCA approach

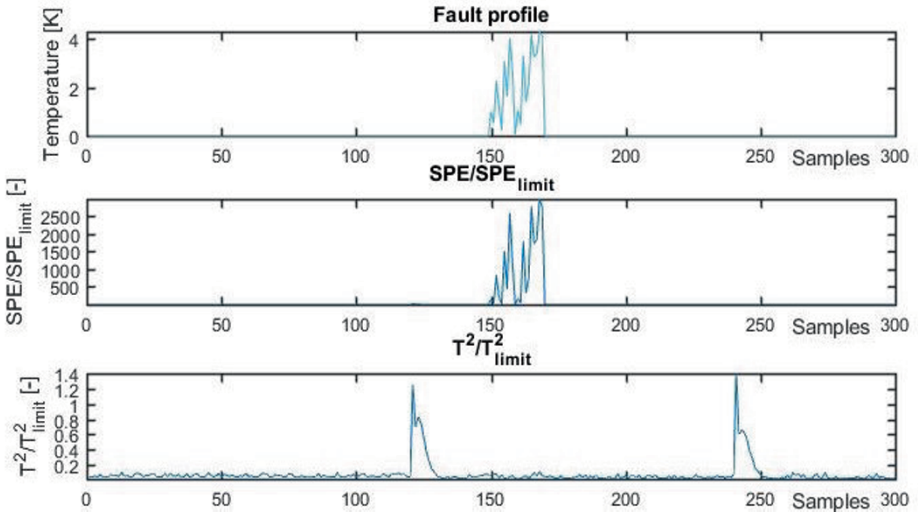


Fig. 6. Thermocouple fault detection using k-means-PCA approach

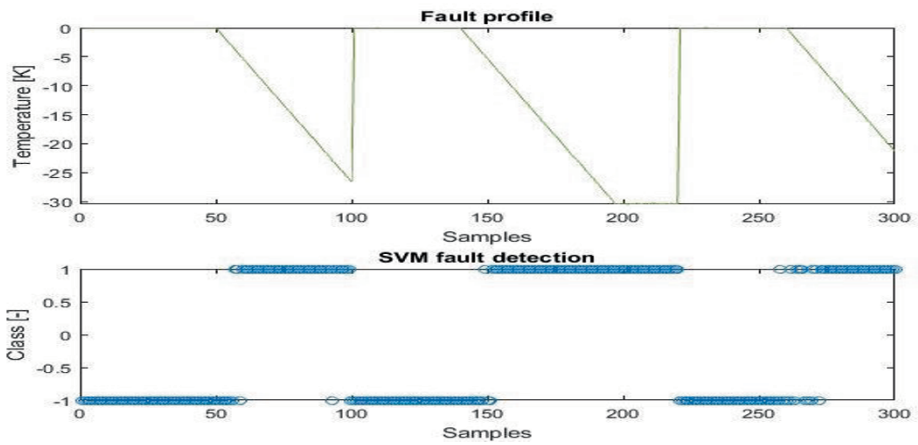


Fig. 7. Water Hammering detection using PCA-SVM approach

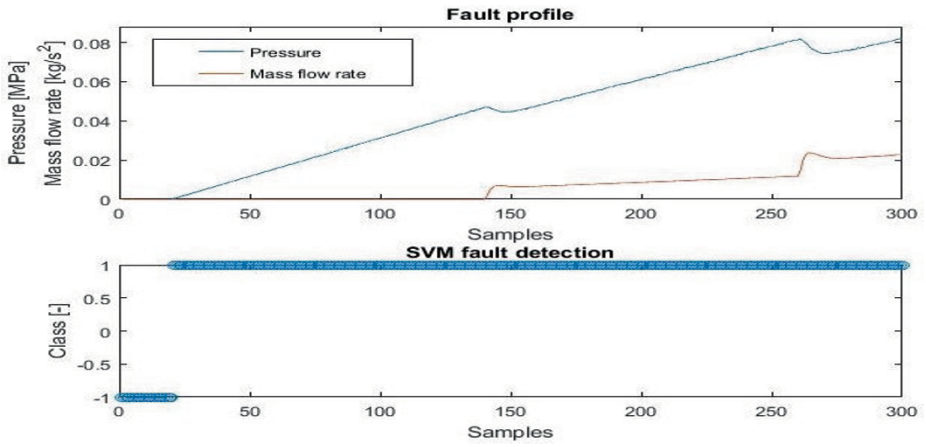


Fig. 8. Salt fouling detection using PCA-SVM approach

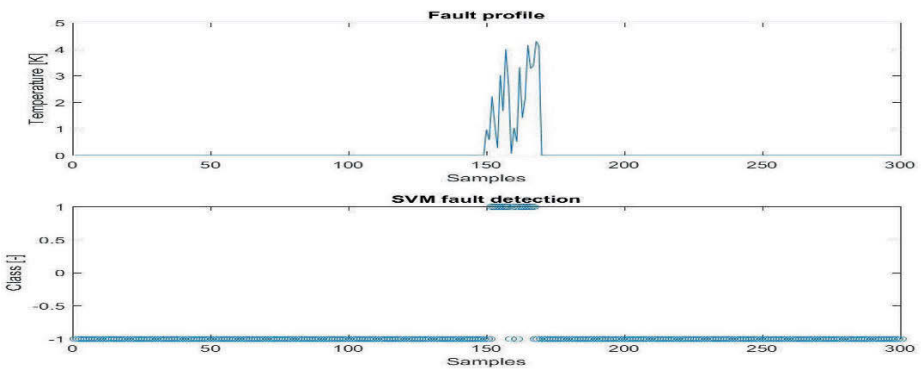


Fig. 9. Thermocouple fault detection using PCA-SVM approach

the kind of faults or anomalies. In order to enable the linear PCA to cope with nonlinear nature of the data, they were clustered before using k-means method. In contrast to the PCA, SVM is a representative of the two-class classification methods and enables work with the nonlinear data. To facilitate the process of classification the data was preprocessed by PCA for the size reduction and for extraction of the most important process features. Both approaches have proven themselves in detecting of the simulated anomalies. Authors current work is focused on utilization of the information from the diagnostic system into predictive control of the nuclear power plant steam turbine (FTC MPC).

References

1. Pawlik M., Strzelczyk F., *Elektrownie*, WNT, Warszawa, 2014.
2. Korbicz J., Kocielny J.M., Kowalczyk Z., Cholewa W., (Eds) *Fault Diagnosis. Models, Artificial Intelligence, Applications.*, Berlin, Heidelberg: Springer-Verlag, 2004.
3. Kulkowski K., Kobylarz A., Grochowski M., Duzinkiewicz K., *Dynamic model of nuclear power plant steam turbine*, Archives of Control Sciences, vol. 25 (LXI), 2015, pp. 65-86.
4. Janiczek R., *Eksploatacja elektrowni parowych*, WNT, 1980
5. www.plant-maintenance.com/articles/steam_turbine_analysis.shtml.(11.01.2017)
6. Perycz S., Próchnicki W., *Steam turbine mathematical model of nuclear power unit with WWER-440 reactor, allowing to analyze transient states with $\omega=var.$* , Institute of Electrical Power and Control Engineering, Gdańsk University of Technology, Gdańsk, Poland, Technical Report, 1989, in Polish.
7. Dobosz J., Duzinkiewicz K., Perycz S., Próchnicki W., *Steam turbine simulation model of transient states for nuclear power unit with WWER-440 reactor with $\omega=const.$* , Institute of Electrical Power and Control Engineering, Gdańsk University of Technology, Gdańsk, Poland, 1989, in Polish.
8. Sokółski P., Kulkowski K., Kobylarz A., Duzinkiewicz K., Rutkowski T.A., Grochowski M., *Wieloobszarowa regulacja systemu turbogenerators elektrowni jądrowej*, Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej, vol. 42, Gdańsk, 2015, pp. 129-132.
9. Grochowski M., Matczak M., Sokołowski M., *Optimising approach to designing kernel PCA model for diagnosis purposes with and without a priori known faulty data*, IEEE Proc. of the 20th International Conference on Methods and Models in Automation and Robotics MMAR 2015, Międzyzdroje, Poland, 2015.
10. Nowicki A., Grochowski M., *Kernel PCA in application to leakage detection in drinking water distribution system*, Book Series: Lecture Notes in Artificial Intelligence, Vol. 6922, 2011, pp. 497-506.
11. Swędrowski, L., Duzinkiewicz, K., Grochowski, M., Rutkowski T., *Use of neural networks in diagnostics of rolling-element bearing of the induction motor*, SMART DIAGNOSTICS V Book Series: Key Engineering Materials, Vol. 588, 2014, pp. 333-342.
12. Panek D., Skalski A., Gajda J., Tadeusiewicz R., *Acoustic analysis assessment in speech pathology detection*, Int. Journal of Applied Mathematics and Computer Science, No. 3, Vol. 25, 2015.
13. Jackson J.E., *A User's Guide to Principal Components*, John Wiley & Sons, INC., 1991.

Path planning algorithm for ship collisions avoidance in environment with changing strategy of dynamic obstacles

Łukasz Kuczkowski and Roman Śmierzchalski

Gdansk University of Technology, Faculty of Electrical and Control Engineering, str. Gabriela Narutowicza 11/12, 80-233 Gdansk, Poland
{lukasz.kuczkowski, roman.smierzchalski}@pg.gda.pl

Abstract. In this paper a path planning algorithm for the ship collision avoidance is presented. Tested algorithm is used to determine close to optimal ship paths taking into account changing strategy of dynamic obstacles. For this purpose a path planning problem is defined. A specific structure of the individual path and fitness function is presented. Principle of operation of evolutionary algorithm and based on it dedicated application vEP/N++ is described. Using presented algorithm the simulations on close-to-real sea environment is performed. Tested environment presents the problem of avoiding one static obstacle representing island and two dynamic objects representing strange ships. Obtained results proof that used approach allows to calculate efficient and close-to-optimal path for marine vessel in close-to-real time.

Keywords: Path planning·Evolutionary algorithms·Collision avoidance

1 Introduction

One of the tasks for controlling a moveable object (i.e. mobile robot, autonomous marine vehicle) is to get an object from a specified starting point to its destination or to the task (mission) area. To achieve this a path has to be plotted against a specific criterion i.e. the shortest time to reach the destination. The path has to avoid obstacles which are treated as static and dynamic limitations. Usually the dynamic obstacles are moveable objects, which move along certain trajectory with specific speed and static obstacles represent fixed elements of the environment. A briefly described problem is called path planning and can be divided into two basic tasks: an off-line task, in which we look for the path of the object in a steady environment, and an on-line task, in which the object moves in the environment that meets the variability and uncertainty restrictions. The on-line mode of the path planning relates to the control of the moving object in the non-stationary environment, in which parts of some obstacles reveal certain dynamics.

The problem of path planning occurs in numerous technical applications, such as, for instance planning crane heavy lifting in industrial plants [1, 2]. The cranes are frequently used in plant construction, maintenance shutdown and new equipment installation. The task is to find a safe and cost effective way of lifting taking into account plant environment, crane mechanical data, crane position, start and end lifting configurations.

Normally the group of workers solve the problem by site investigation, planning and evaluations based on their experience and available data. Another example can be control of marine vessel [3]. The task is to design a position and heading control system for surface vessel. A path plotted by path planning system in this case is used as a reference value for error measurement. One of the most common application of path planning is motion planning of robotic arm manipulators [4, 5]. Robotic manipulators have vast array of usage especially in assembly industrial systems such as in the automotive industry. The movement of robotic arm can be control based on end points or trajectories which can be difficult to calculate due to complexity of the optimization problem and dynamics of the obstacles.

Another group of problems are the issues related to path and trajectory planning for all kinds of vehicles. The problem is defined in the following way: having given a moving object and the description of the environment, plan the path or trajectory for an object motion between the beginning and end location which avoids all constraints and satisfies certain optimization criteria. Recently, in the center of attention are especially unmanned vehicles both autonomous and remotely operated: aerial [6, 7], ground [8, 9], surface [10, 11], underwater [12, 13]. Another important issue in vehicles path planning is weather routing problem [14]. It is crucial from the point of view of passage time, fuel consumption and safety of passage to consider e.g. high wind speed regions or customisable areas (e.g. due to piracy).

Most of existing methods of solving the path planning problem can be divided in two categories: traditional methods and intelligent methods. Traditional methods, in the majority, include graph search techniques [15, 16] (in which the search/configuration space is discretized and based on that changed into graph, then the shortest path algorithm, like Dijkstra [17] or A* [18], are used to determine close-to-optimal path through the graph) and artificial potential field method [19, 20], where the basic concept is to fill the search/configuration space with an artificial potential field in which the considered moving object is attracted to its target and is repulsed away from the obstacles. Intelligent methods include: evolutionary algorithms [21, 22, 23], ant colony optimization [24, 25], particle swarm optimization [26, 27], fuzzy logic [28, 29] and artificial neural network [30, 31].

In this paper path planning problem is described as task of ship collisions avoidance taking into account variable strategy of dynamic obstacles. In most cases it is assumed that dynamic obstacle (in maritime nomenclature strange ship or targets) move along a straight line defining by their course. The main goal of this paper is to test evolutionary path planning algorithm whether it is capable to properly determine the safe ship path in define environment.

This paper is organized in the following way: in chapter two the path planning problem is defined. Chapter three describes the evolutionary algorithm and vEPN++ application, while the following chapter presents the results. In chapter 5 a discussion and conclusions are presented.

2 Problem Definition

The problem of path planning in collision situation for ships consists of plotting a path P , as a part of a route the ship travels from current position (starting point $p_0(x_0, y_0)$) to the destination point $p_e(x_e, y_e)$. The path is composed of linear segment sequences p_i ($i = 1, \dots, n$), connected with turning nodes (x_i, y_i) . The start and destination points are chosen by the operator. Considering this, a path P is feasible (is a part of the save route set) if each of its segments p_i ($i = 1, \dots, n$) remains within the boundaries of the environment and does not cross with neither dynamic nor static obstacles. The paths which cross the restricted areas generated by the static and dynamic constrains are considered unsafe or dangerous paths (target 1, point PPK (x, y) , Fig. 1). It is assumed that the search/configuration space is defined in two-dimensional cartesian coordinate system and its size depends on current conditions on sea and is determined by operator. Own ship model is 2 DOF and operates on kinematics and due to calculation simplicity it is considered as a point in search space.

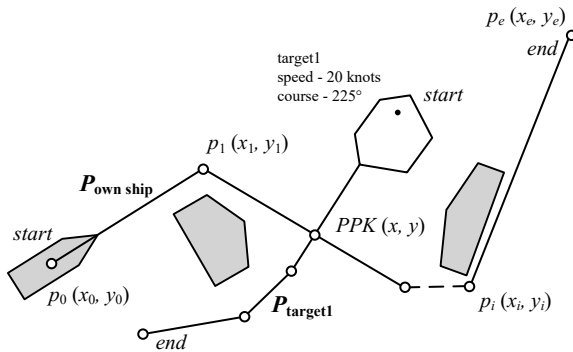


Fig. 1. Potential collision scenario

The task is to find a trajectory that compromises the cost of a necessary deviation from a given route, or from the optimum route leading to a destination point, and the safety of passing all static and dynamic obstacles. All trajectories which meet the safety conditions reducing the risk of collision to a satisfactory level constitute a set of feasible trajectories. The safety conditions are, as a rule, defined by the operator based on the speed ratio between the ships involved in the passing manoeuvre, the actual visibility, weather conditions, navigation area, manoeuvrability of the ship, etc.

Other constraints resulting from formal regulations (e.g., traffic restricted zones, fairways, etc) are assumed stationary and are defined by polygons – in a similar manner to that used in creating the electronic maps. When sailing in the stationary environment, the own ship meets other sailing strange ships/targets (some of which constitute a collision threat).

It is assumed that a dangerous target is a target that has appeared in the area of observation and can cross the estimated course of the own ship at a dangerous distance. The actual values of this distance depend on the assumed time horizon. Usually, the

distances of 5-8 nautical miles in front of the bow, and 2-4 nautical miles behind the stern of the ship are assumed as the limits for safe passing. In the research, the targets threatening with a collision are interpreted as the moving dangerous areas having shapes of domains and speeds parameter. Each target has its own path which determines its movement.

The problem presented is reduced to an optimization task with static and dynamic obstacles. The level of adaptation of the trajectory to the environment determines the total cost of the trajectory, which includes both the safety cost $Safe_Cost(P)$ and that connected with the economy $Econ_Cost(P)$ of the ship motion along the trajectory of concern. The total cost of the trajectory (fitness function) is defined as:

$$Total_Cost(P) = Safe_Cost(P) + Econ_Cost(P) \quad (1)$$

The safety conditions are met when the trajectory does not cross the fixed navigational constraints, nor the moving areas of danger. The actual value of the safety cost function $Safe_Cost(P)$ is evaluated as the maximum value defining the quality of the turning points p_i with respect to their distance from the constraints:

$$Safe_Cost(P) = w_c \cdot clear(P) \quad (2)$$

where: $clear(P) = \max_{i=1}^n c_i$, w_c is the weight coefficient, c_i is the length difference between the distance to the constraint (the closest turning point p_i) and the safe distance d defined by operator.

The trajectory cost connected with the economic conditions $Econ_Cost(P)$ includes: the total length of the trajectory P consisting of n line sections $p_i - dist(P)$, the function of the maximum turning angle between particular trajectory sections at turning points $p_i - smooth(P)$, the time needed for covering the trajectory $P - time(P)$. The total cost of the trajectory adaptation to the environment, resulting from the economic conditions, is equal to:

$$Econ_Cost(P) = w_d \cdot dist(P) + w_s \cdot smooth(P) + w_t \cdot time(P) \quad (3)$$

where: w_d, w_s, w_t are the weight coefficients.

Weight coefficients are determined by navigator preferences, concerning actual sailing conditions. One can set the algorithm to weighed the particular element of the path cost so that i.e. the safest paths will be evaluated as the ones with the highest fitness even though they will at the same time represent the longest route.

3 Evolutionary Path Planning Algorithm

Presented path planning problem is solved using an evolutionary method. In general, evolutionary algorithms process a set of solutions called population. The environment is defined based on the task pending (fitness function, constrains). Each individual (single population member) represents a different problem's solution. Based on the fitness function each individual is assigned a parameter called the fitness score. The fitness score determines the quality of the solution represented by each member. In the moment

of algorithm's initialization, the initial conditions are set. Each individual is being randomly generated. Afterwards the following steps are executed: selection of parents, genetic operations, evaluation and succession. In selection phase a temporary population is created to which random individuals from the base population are being copied. It is possible to introduce more than one copy of any individual. The greater the fitness score, the greater the chance of selecting particular member of the population. In the next step, the temporary population is processed by the genetic operations, which modifies individuals. The set of solutions calculated in this way is called the child population, which is evaluated. The succession phase creates a new base population. Those algorithm's phases are repeated in a loop until the termination condition is met.

The attractiveness of the use of the evolutionary techniques is connected with the fact that:

- random search is believed to be the most effective in dealing with NP-hard problems and in escaping from local minima,
- parallel search actions not only secure high speed but also provide opportunities for interactions between search actions, all this acting in favour of better efficiency of the optimization,
- intelligent behaviour can be treated as a composition of simple reactions to a complex world,
- a planner can be much more simplified, and still much more efficient and flexible, and increase the quality of the search if it is not confined to the action within a specific map structure,
- it is better to equip the planner with the flexibility to change the optimization goals than the ability to find the absolute optimum solution for a single particular goal.

Based on evolutionary computation technics the vEPN++ application was developed [21]. The vEP/N++ realizes all the above ideas by incorporating part of the maritime path planning problem knowledge into the evolutionary algorithm. What is evenly important and not quite obvious, due to the design of the chromosome structure (Table 1), which consist of set of turning points and genetic operators the vEP/N++ does not need a discrete map for search, which is usually required by other planners. Instead, the vEP/N++ "searches" the original and continuous environment by generating paths with the aid of various evolutionary operators. The objects in the environment can be defined as collections of straight-line "walls". This representation refers both to the known objects as well as to partial information of the unknown objects obtained from sensing. As a result, there is little difference for the vEP/N++ between the off-line planning and the on-line navigation. In fact, the vEP/N++ realizes the off-line planning and the on-line navigation using the same evolutionary algorithm and chromosome structure.

Table 1. The structure of the chromosome

PATH	
Turning points	Coordinates and speed
p_0	x_0, y_0, v_0
p_1	x_1, y_1, v_1
...	...
p_e	x_e, y_e, v_e

A crucial step in the development of the evolutionary trajectory planning systems was made by introducing the dynamic parameters: time and moving constraints. Chromosome consists of path nodes (turning points), that are described by own vessel course, speed and coordinates of actual position. In the evolutionary algorithm used for trajectory planning eight genetic operators are used: soft mutation, mutation, adding a gene, swapping gene locations, crossing, smoothing, deleting a gene, and individual repair [21]. The presented operators are variants of the mutation. When the mutation operator is selected to work, one of them is picked at random with a normal distribution.

Described application was tested and tuned against wide variety of testing environments including: weight coefficients tune, different methods of pre and post-selection, choose of termination function etc. [21, 22], [32, 33].

4 Results

The vEP/N++ application is used to solve the problem of maritime path planning in collision situation taking into account changing strategy of strange ships. A testing environment representing close to real maritime scenario. The following parameters are considered: ψ – course, v – speed. Environment (Fig. 2) presents the problem of avoiding one static obstacle representing island (Fig 2, Static obstacle) and two dynamic objects (target 1 and target 2). Figure 2 presents starting positions and destination of targets. Dotted lines represent paths on which targets move. Start and end are starting and destination points for own ship. Target 1 starts with $\psi = 90^\circ$, $v = 17$ knots and target 2 with $\psi = 315^\circ$, $v = 19$ knots. Speed of target ships do not change during the simulation.

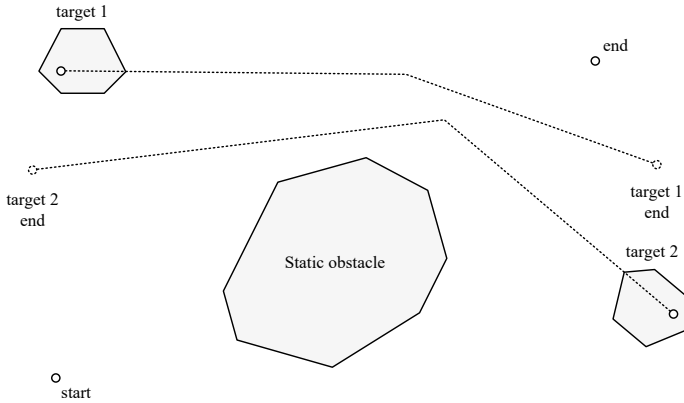


Fig. 2. Simulation environment

For presented environment the path planning process is performed by using the vEP/N++. The simulations are performed with parameters presented in Table 2. The termination criteria is set to 400 generations. Due to random nature of evolutionary algorithms the series of 50 simulations with the same parameters is perform. Achieved results are presented in graphical form (one figure with the best obtained result) and in table in which the values of fitness function and simulation time is compared.

Table 2. Simulation parameters

Parameter	Value
Population size	30
Crossover probability	0.8
Mutation probability	0.35
Selector	rank
Number of individuals replaced	6
Number of generations	400
Initial own ship's speed	21 knots

Figure 3 represent the best obtained solution. Individual with highest fitness score has his path bolded. The position of the dynamic objects is displayed for the best member of the population.

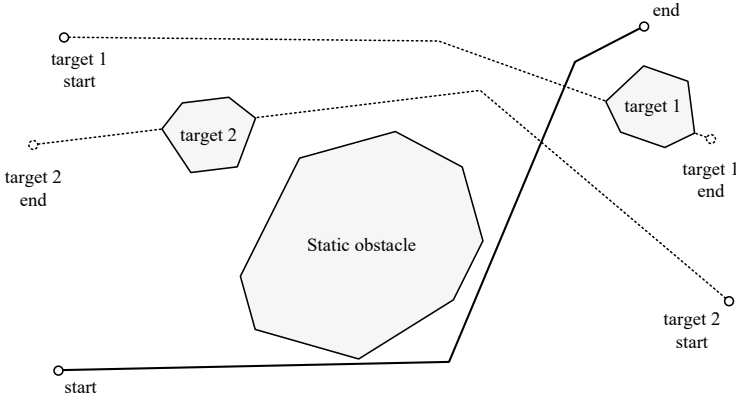


Fig. 3. Result after 400 generations

Table 3 presents summarized results from all simulations.

Table 3. Obtained results

Parameter	Value
The best fitness value	283,44
Average fitness value	307,23
The worst fitness value	451,83
The best calculation time	3,7s
Average calculation time	4,2s
The worst calculation time	4,5s

5 Discussion and conclusions

Presented algorithm was able to solve described problem in all simulations. Generated path (Figure 3) is correct in terms of collision avoidance and meets the requirements of possibly shortest and smoothest shape. In all performed simulations the algorithm returns correct solution. Most of the differences between solutions results from the static obstacle. In some solutions algorithm chose path above the static obstacle (not shown on figure) which results in slightly worse solution. Best and average fitness function value do not differ much. It means that the algorithm is recurrent. The base, average and the worst calculation times are similar and in all cases meet the requirements of close-to-real time operation within the meaning of maritime environment.

In this paper the path planning algorithm for collision avoidance at sea was presented. The algorithm was tested against dynamic environment in which the strategy of moving obstacle change. The simulation proofed that used approach allows to calculate efficient and close-to-optimal path for marine vessel in close-to-real time.

References

1. Cai, P.P., Cai, Y.Y., Chandrasekaran, I., Zheng, J.M.: Parallel genetic algorithm based automatic path planning for crane lifting in complex environments. *Automation in Construction* 62, 133-147 (2016)
2. Wu, Z., Xia, X.H.: Optimal motion planning for overhead cranes. *IET Control Theory and Applications* Vol. 8, Issue 17, pp. 1833-1842 (2014)
3. Witkowska, A.: Control Design for Slow Speed Positioning. In: Proc. 27th European Conference on Modelling and Simulation ECMS 2013, pp. 198-204 (2013)
4. Fei, Y.Q., Ding, F.Q., Zhao, X.F.: Collision-free motion planning of dual-arm reconfigurable robots. *Robotics and Computer-Integrated Manufacturing*, Vol. 20, Issue 4, pp. 351-357 (2004)
5. Korayem, M.H., Esfeden, R.A., Nekoo, S.R.: Path planning algorithm in wheeled mobile manipulators based on motion of arms. *Journal of Mechanical Science and Technology*, Vol. 29, Issue 4, pp. 1753-1763 (2015)
6. Nikolos, I.K., Valavanis, K.P., Tsurveloudis, N.C., Kostaras, A.N.: Evolutionary algorithm based offline/online path planner for UAV navigation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 33, No. 6, pp. 898-912 (2003)
7. Goerzen, C., Kong, Z., Mettler, B.: A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV Guidance. *Journal of Intelligent & Robotic Systems*, Vol. 57, Issue 1-4, pp. 65-100 (2010)
8. Vandapel, N., Donamukkala, R.R., Hebert, M.: Unmanned ground vehicle navigation using aerial ladar data. *International Journal of Robotics Research*, Vol. 25, Issue 1, pp. 31-51 (2006)
9. Hao, Y.X., Agrawal, S.K.: Planning and control of UGV formations in a dynamic environment: A practical framework with experiments. *Robotics and Autonomous Systems*, Vol. 51, Issue 2-3, pp. 101-110 (2005)
10. Campbell, S., Naeem, W., Irwin, G.W.: A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres. *Annual Reviews in Control*, Vol. 36, Issue 2, pp. 267-283 (2012)
11. Thakur, A., Svec, P., Gupta, S.K.: GPU based generation of state transition models using simulations for unmanned surface vehicle trajectory planning. *Robotics and Autonomous Systems*, Vol. 60, Issue 12, pp. 1457-1471 (2012)
12. Petres, C., Pailhas, Y., Patron, P., Petillot, Y., Evans, J., Lane, D.: Path planning for autonomous underwater vehicles. *IEEE Transactions on Robotics*, Vol. 23, Issue 2, pp. 331-341 (2007)
13. Repoullias, F., Papadopoulos, E.: Planar trajectory planning and tracking control design for underactuated AUVs. *Ocean Engineering*, Vol. 34, Issue: 11-12, pp. 1650-1667 (2007)
14. Szlapczynska, J.: Multi-objective Weather Routing with Customised Criteria and Constraints. *Journal of Navigation*, Vol. 68, Issue 2, pp. 338-354 (2015)
15. Maaref, H., Barret, C.: Sensor-based navigation of a mobile robot in an indoor environment. *Robotics and Autonomous Systems*, Vol. 38, Issue 1, pp. 1-18 (2002)
16. Jiang, K.C., Seneviratne, L.D., Earles, S.W.E.: A shortest path based path planning algorithm for nonholonomic mobile robots. *Journal of Intelligent & Robotic Systems*, Vol. 24, Issue 4, pp. 347-366 (1999)
17. Davoodia, M., Panahi, F., Mohadesc, A., Hashemic, S.N.: Multi-objective path planning in discrete space. *Applied Soft Computing* 13, 709-720 (2013)

18. Ari, I., Aksakalli, V., Aydogdu, V., Kum, S.: Optimal ship navigation with safety distance and realistic turn constraints. *European Journal of Operational Research* 229, 707-717 (2013)
19. Barraquand, J., Langlois, B., Latombe, J.C.: Numerical Potential-Field Techniques for Robot Path Planning. *IEEE Transactions on Systems Man and Cybernetics*, Vol. 22, Issue 2, pp. 224-241 (1992)
20. Ge, S.S., Cui, Y.J.: Dynamic motion planning for mobile robots using potential field method. *Autonomous Robots*, Vol. 13, Issue 3, pp. 207-222 (2002)
21. Smierzchalski, R., Michalewicz, Z.: Modeling of ship trajectory in collision situations by an evolutionary algorithm. *IEEE Transactions on Evolutionary Computation*, Vol. 4, Issue 3, pp. 227-241 (2000)
22. Kuczkowski, L., Smierzchalski, R.: Comparison of Single and Multi-population Evolutionary Algorithm for Path Planning in Navigation Situation. *Solid State Phenomena* 210, 166-177 (2014)
23. Alvarez, A., Caiti, A., Onken, R.: Evolutionary path planning for autonomous underwater vehicles in a variable ocean. *IEEE Journal of Oceanic Engineering*, Vol. 29, Issue 2, pp. 418-429 (2004)
24. Lazarowska, A.: Swarm Intelligence Approach to Safe Ship Control. *Polish Maritime Research*, Vol. 22, Issue 4, pp. 34-40 (2016)
25. Chen, X., Kong, Y.Y., Fang, X., Wu, Q.D.: A fast two-stage ACO algorithm for robotic path planning. *Neural Computing & Applications*, Vol. 22, Issue 2, pp. 313-319 (2013)
26. Roberge, V., Tarbouchi, M., Labonte, G.: Comparison of Parallel Genetic Algorithm and Particle Swarm Optimization for Real-Time UAV Path Planning. *IEEE Transactions on Industrial Informatics*, Vol. 9, Issue 1, pp. 132-141 (2013)
27. Fu, Y.G., Ding, M.Y., Zhou, C.P.: Phase Angle-Encoded and Quantum-Behaved Particle Swarm Optimization Applied to Three-Dimensional Route Planning for UAV. *IEEE Transactions On Systems Man And Cybernetics, Part A - Systems And Humans*, Vol. 42, Issue 2, pp. 511-526 (2012)
28. Wang, M., Liu, J.N.K.: Fuzzy logic-based real-time robot navigation in unknown environment. *Robotics and Autonomous Systems*, Vol. 56, Issue 7, pp. 625-643 (2008)
29. Yang, X.Y., Moallem, M., Patel, R.V.: A layered goal-oriented fuzzy motion planning strategy for mobile robot navigation. *IEEE Transactions on Systems Man and Cybernetics part B – Cybernetics*, Vol. 35, Issue 6, pp. 1214-1224 (2005)
30. Glasius, R., Komoda, A., Gielen, S.C.A.M.: Neural-Network Dynamics for Path Planning and Obstacle Avoidance. *Neural Networks*, Vol. 8, Issue 1, pp. 125-133 (1995)
31. Yang, S.X., Meng, M.Q.H.: Real-time collision-free motion planning of a mobile robot using a neural dynamics-based approach. *IEEE Transactions on Neural Networks*, Vol. 14, Issue 6, pp. 1541-1552 (2003)
32. Kuczkowski, L., Smierzchalski, R.: Selection Pressure in the Evolutionary Path Planning Problem. In: Korbicz, J., Kowal, M. (eds.) *DPS 2014. AISC*, Vol. 230, pp. 523-534. Springer (2014)
33. Kuczkowski, L., Smierzchalski, R.: Termination functions for evolutionary path planning algorithm. *Proc. of 19th International Conference on Methods and Models in Automation and Robotics (MMAR)*, pp. 636-640, Miedzyzdroje, Poland (2014)

Robust sensor fault-tolerant control for non-linear aero-dynamical MIMO system

Marcin Pazera¹ and Marcin Witczak¹

¹Institute of Control and Computation Engineering,
University of Zielona Góra,
65-254 Zielona Góra, Poland
{M.Pazera,M.Witczak}@issi.uz.zgora.pl

Abstract. The paper deals with the problem of sensor fault-tolerant control for non-linear MIMO systems. The proposed strategy is based on fault estimation strategy. The \mathcal{H}_∞ approach is used to design the observer. Subsequently, the control strategy for the system with faulty sensors is proposed. The fault-tolerant controller is designed in such a way as to achieve the \mathcal{H}_∞ performance and tolerate predefined sensor faults. The final part shows an illustrative example with an implementation to a twin-rotor system.

1 Introduction

The problem of satisfying qualitative factors is an important thing while controlling the system. However, if some faults occur the classical controller may be insufficient to guide the system correctly. Moreover, such a situation can be dangerous for environment or even for human's health and life. Therefore, in the recent decades, the problem of Fault Diagnosis (FD) [3, 16, 8, 11, 1] has grown its importance. The researchers are developing newer and newer approaches to detect (fault detection), identify (fault identification) and localize (fault isolation) the faults which allows to improve the control strategy. From the point of view of the system, three classes of faults may be taken under consideration [3, 6, 20]: actuator, sensor and process (or component) faults. There exist Fault Detection and Isolation (FDI) solutions such as: minimum variance input and state estimator [7], sliding mode high-gain observers [19, 2], Kalman filter [10], adaptive estimation [23] and also \mathcal{H}_∞ approach [20, 14, 9, 4]. The FD acts as not only informative but increasingly, is combined with the control system. The merger of the control and FD is so-called as Fault-Tolerant Control (FTC) [6, 15, 13, 20, 18].

The main objective of the paper is to propose the control strategy that allows to guide the system while the sensors faults occur. The proposed strategy is robust to process as well as measurement uncertainties and is based on \mathcal{H}_∞ approach. To compensate the fault influence its estimate is subtracted from the state.

The paper is organized as follows: section 2 describes the control problem in the case when any sensor is faulty, subsequently, section 3 proposes a designing

path of the observer that allows to compensate the occurred faults. Section 4 presents some results obtained with the proposed approach by its implementation to a twin-rotor MIMO system and finally, section 5 concludes the paper.

2 Problem statement

Let us consider a non-linear discrete-time system:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{W}_1\mathbf{w}_k, \quad (1)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{f}_{s,k} + \mathbf{W}_2\mathbf{w}_k, \quad (2)$$

where $\mathbf{x}_k \in \mathbb{X} \subset \mathbb{R}^n$, $\mathbf{u}_k \in \mathbb{R}^r$, $\mathbf{y}_k \in \mathbb{R}^m$, are the state, control input and output vectors, respectively. The non-linear function $\mathbf{g}(\mathbf{x}_k, \mathbf{u}_k)$ describes the behaviour of the system with respect to the state and input. Moreover $\mathbf{f}_{s,k} \in \mathbb{F}_s \subset \mathbb{R}^{n_f}$ is the sensor fault vector, where n_f is the number of sensor faults. Furthermore, \mathbf{W}_1 and \mathbf{W}_2 denote the noise distribution matrices and \mathbf{w}_k expresses the exogenous disturbance vector. It can be easily shown that \mathbf{w}_k can be split in such a way as $\mathbf{w}_k = [\mathbf{w}_{1,k}^T, \mathbf{w}_{2,k}^T]^T$ where $\mathbf{w}_{1,k}$ and $\mathbf{w}_{2,k}$ are process and measurement uncertainties, respectively.

The problem is to control the system irrespective to the fact of sensor fault occurrence. The integrated strategy of control and fault diagnosis should be able to tolerate the sensor faults which may occur in the system. The general idea that state behind this approach is to estimate the faults. Based on this knowledge the state estimate can be used to guide the system.

3 Control strategy

The main topic of this section is to design the observer which will make possible to estimate all the states as well as all the sensor faults, simultaneously.

In this paper the following state and sensor fault estimator is proposed:

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_k + \mathbf{g}(\hat{\mathbf{x}}_k, \mathbf{u}_k) + \mathbf{K}_x \left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_k - \hat{\mathbf{f}}_{s,k} \right), \quad (3)$$

$$\hat{\mathbf{f}}_{s,k+1} = \hat{\mathbf{f}}_{s,k} + \mathbf{K}_s \left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_k - \hat{\mathbf{f}}_{s,k} \right). \quad (4)$$

Note that, instead of using a set of the observers, one single observer is able to estimate the faults for all sensors in the system. The control actions can be calculated using the following form

$$\mathbf{u}_k = \mathbf{K}_c\hat{\mathbf{x}}_k + \mathbf{K}_r\mathbf{r}_k, \quad (5)$$

where \mathbf{K}_c , \mathbf{K}_r and \mathbf{r}_k are the control gain matrix, pre-filter matrix and reference vector, respectively. Thus, based on the fault free state estimate the classical state feedback controller is proposed. The gain matrix for the controller is not a

main topic of consideration and it could be obtained with well-known methodologies.

The problem is to find \mathbf{K}_x and \mathbf{K}_s which represent gain matrices for the state and fault estimate, respectively. To handle with this issue from (1), (2) and (3) a state estimation error can be given

$$\begin{aligned} \mathbf{e}_{k+1} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{W}_1\mathbf{w}_k - \mathbf{A}\hat{\mathbf{x}}_k - \mathbf{B}\mathbf{u}_k \\ &\quad - \mathbf{g}(\hat{\mathbf{x}}_k, \mathbf{u}_k) - \mathbf{K}_x\mathbf{y}_k + \mathbf{K}_x\mathbf{C}\hat{\mathbf{x}}_k + \mathbf{K}_x\hat{\mathbf{f}}_{s,k} = [\mathbf{A} - \mathbf{K}_x\mathbf{C}] \mathbf{e}_k \\ &\quad - \mathbf{K}_x\mathbf{e}_{s,k} + [\mathbf{W}_1 - \mathbf{K}_x\mathbf{W}_2] \mathbf{w}_k, \end{aligned} \tag{6}$$

where $\mathbf{e}_{s,k} = \mathbf{f}_{s,k} - \hat{\mathbf{f}}_{s,k}$. Subsequently, from (2) and (4) the fault estimation error can be presented as follows

$$\begin{aligned} \mathbf{e}_{s,k+1} &= \mathbf{f}_{s,k+1} - \hat{\mathbf{f}}_{s,k+1} = \mathbf{f}_{s,k+1} + \mathbf{f}_{s,k} - \mathbf{f}_{s,k} - \hat{\mathbf{f}}_{s,k} - \mathbf{K}_s\mathbf{y}_k \\ &\quad + \mathbf{K}_s\mathbf{C}\hat{\mathbf{x}}_k + \mathbf{K}_s\hat{\mathbf{f}}_{s,k} = \boldsymbol{\varepsilon}_k + [\mathbf{I} - \mathbf{K}_s] \mathbf{e}_{s,k} - \mathbf{K}_s\mathbf{C}\mathbf{e}_k - \mathbf{K}_s\mathbf{W}_2\mathbf{w}_k, \end{aligned} \tag{7}$$

with $\boldsymbol{\varepsilon}_k = \mathbf{f}_{s,k+1} - \mathbf{f}_{s,k}$ which means the error between consecutive samples of the fault.

Moreover, using Differential Mean Value Theorem (DMVT) [21] it can be shown that

$$\mathbf{g}(\mathbf{x}_k, \mathbf{u}_k) - \mathbf{g}(\hat{\mathbf{x}}_k, \mathbf{u}_k) = \mathbf{M}_k(\mathbf{x}_k - \hat{\mathbf{x}}_k), \tag{8}$$

with

$$\mathbf{M}_k = \begin{bmatrix} \frac{\partial g_1}{\partial x}(\mathbf{c}_1, \mathbf{u}_k) \\ \vdots \\ \frac{\partial g_n}{\partial x}(\mathbf{c}_n, \mathbf{u}_k) \end{bmatrix}, \tag{9}$$

where $\mathbf{c}_1, \dots, \mathbf{c}_n \in \text{Co}(\mathbf{x}_k, \hat{\mathbf{x}}_k)$, $\mathbf{c}_i \neq \mathbf{x}_k$, $\mathbf{c}_i \neq \hat{\mathbf{x}}_k$, $i = 1, \dots, n$. Having regard the fact that all states are bounded in real system, $\mathbf{x}_k \in \mathbb{X}$ let

$$\underline{x}_{i,j} \leq \frac{\partial g_i(\mathbf{x})}{\partial x_j} \leq \bar{x}_{i,j}, \quad i = 1, \dots, n, \quad j = 1, \dots, n, \tag{10}$$

it is clear that there exist $\mathbf{M}_k \in \mathbb{M}$ such that

$$\mathbb{M} = \{ \mathbf{M}_k \in \mathbb{R}^{n \times n} | \underline{x}_{i,j} \leq m_{k,i,j} \leq \bar{x}_{i,j}, i, j = 1, \dots, n, \}. \tag{11}$$

Thus, the state estimation error (6) can be rewritten into the following form

$$\begin{aligned} \mathbf{e}_{k+1} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1} = [\mathbf{A} + \mathbf{M}_k - \mathbf{K}_x\mathbf{C}] \mathbf{e}_k - \mathbf{K}_x\mathbf{e}_{s,k} \\ &\quad + [\mathbf{W}_1 - \mathbf{K}_x\mathbf{W}_2] \mathbf{w}_k. \end{aligned} \tag{12}$$

Furthermore, the following super-vectors can be constructed:

$$\bar{\mathbf{e}}_{k+1} = \begin{bmatrix} \mathbf{e}_{k+1} \\ \mathbf{e}_{s,k+1} \end{bmatrix}, \tag{13}$$

$$\mathbf{v}_k = \begin{bmatrix} \mathbf{w}_k \\ \boldsymbol{\varepsilon}_k \end{bmatrix}. \tag{14}$$

The estimation error of the state and fault can be presented in a compact form

$$\bar{\mathbf{e}}_{k+1} = \mathbf{X}_k \bar{\mathbf{e}}_k + \mathbf{Z} \mathbf{v}_k, \tag{15}$$

with:

$$\mathbf{X}_k = \bar{\mathbf{A}}_k - \bar{\mathbf{K}} \bar{\mathbf{C}}, \tag{16}$$

$$\mathbf{Z} = \bar{\mathbf{W}} - \bar{\mathbf{K}} \bar{\mathbf{V}}, \tag{17}$$

where:

$$\begin{aligned} \bar{\mathbf{A}}_k &= \begin{bmatrix} \mathbf{A} + \mathbf{M}_k \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad \bar{\mathbf{C}} = [\mathbf{C} \ \mathbf{I}], \quad \bar{\mathbf{K}} = \begin{bmatrix} \mathbf{K}_x \\ \mathbf{K}_s \end{bmatrix}, \\ \bar{\mathbf{W}} &= \begin{bmatrix} \mathbf{W}_1 \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad \bar{\mathbf{V}} = [\mathbf{W}_2 \ \mathbf{0}]. \end{aligned} \tag{18}$$

Taking into account the estimation error for both, state and fault, the following theorem can be defined:

Theorem 1. *There exist matrices $\mathbf{N}, \mathbf{U}, \mathbf{P} \succ 0$ and an attenuation level $\mu > 0$ for all $\mathbf{M}_k \in \mathbb{M}$ such that the following inequality is satisfied*

$$\begin{bmatrix} \mathbf{I} - \mathbf{P} & \mathbf{0} & \bar{\mathbf{A}}_k^T \mathbf{U} - \bar{\mathbf{C}}^T \mathbf{N}^T \\ \mathbf{0} & -\mu^2 \mathbf{I} & \bar{\mathbf{W}} \mathbf{U} - \bar{\mathbf{V}}^T \mathbf{N}^T \\ \mathbf{U} \bar{\mathbf{A}}_k - \mathbf{N} \bar{\mathbf{C}} & \mathbf{U} \bar{\mathbf{W}} - \mathbf{N} \bar{\mathbf{V}} & \mathbf{P} - \mathbf{U} - \mathbf{U}^T \end{bmatrix} \prec 0. \tag{19}$$

Proof. The problem of the designing the \mathcal{H}_∞ observer [12, 22] is to obtain matrices \mathbf{N}, \mathbf{U} and \mathbf{P} such that

$$\lim_{k \rightarrow \infty} \bar{\mathbf{e}}_k = \mathbf{0} \quad \text{for} \quad \mathbf{v}_k = \mathbf{0}, \tag{20}$$

$$\|\bar{\mathbf{e}}_k\|_{l_2} \leq \mu \|\mathbf{v}_k\|_{l_2} \quad \text{for} \quad \mathbf{v}_k \neq \mathbf{0}, \bar{\mathbf{e}}_0 = \mathbf{0}. \tag{21}$$

To solve the problem, it is satisfactory to find a Lyapunov function such that

$$\Delta V_k + \bar{\mathbf{e}}_k^T \bar{\mathbf{e}}_k - \mu^2 \mathbf{v}_k^T \mathbf{v}_k < 0, \tag{22}$$

where:

$$V_k = \bar{\mathbf{e}}_k^T \mathbf{P} \bar{\mathbf{e}}_k, \quad \mathbf{P} \succ 0, \tag{23}$$

$$\Delta V_k = V_{k+1} - V_k. \tag{24}$$

As a consequence by using (15) it is easy to show that

$$\begin{aligned} \Delta V_k + \bar{\mathbf{e}}_k^T \bar{\mathbf{e}}_k - \mu^2 \mathbf{v}_k^T \mathbf{v}_k &= \bar{\mathbf{e}}_k^T \left(\mathbf{X}_k^T \mathbf{P} \mathbf{X}_k + \mathbf{I} - \mathbf{P} \right) \bar{\mathbf{e}}_k + \bar{\mathbf{e}}_k^T \left(\mathbf{X}_k^T \mathbf{P} \mathbf{Z} \right) \mathbf{v}_k \\ &+ \mathbf{v}_k^T \left(\mathbf{Z}^T \mathbf{P} \mathbf{X}_k \right) \bar{\mathbf{e}}_k + \mathbf{v}_k^T \left(\mathbf{Z}^T \mathbf{P} \mathbf{Z} - \mu^2 \mathbf{I} \right) \mathbf{v}_k < 0, \end{aligned} \tag{25}$$

which is equivalent to

$$\bar{\mathbf{v}}_k^T \begin{bmatrix} \mathbf{X}_k^T \mathbf{P} \mathbf{X}_k + \mathbf{I} - \mathbf{P} & \mathbf{X}_k^T \mathbf{P} \mathbf{Z} \\ \mathbf{Z}^T \mathbf{P} \mathbf{X}_k & \mathbf{Z}^T \mathbf{P} \mathbf{Z} - \mu^2 \mathbf{I} \end{bmatrix} \bar{\mathbf{v}}_k \prec 0, \tag{26}$$

with $\bar{\mathbf{v}}_k = [\bar{\mathbf{e}}_k^T, \mathbf{v}_k^T]^T$. Now, let us recall the following lemma [5]:

Lemma 1. *The following statements are equivalent:*

1. *There exists $\mathbf{X}_k \succ 0$ such that*

$$\mathbf{V}^T \mathbf{X}_k \mathbf{V} - \mathbf{W} \prec 0. \quad (27)$$

2. *There exist $\mathbf{X}_k \succ 0$ such that*

$$\begin{bmatrix} -\mathbf{W} & \mathbf{V}^T \mathbf{U}^T \\ \mathbf{U} \mathbf{V} & \mathbf{X}_k - \mathbf{U} - \mathbf{U}^T \end{bmatrix} \prec 0. \quad (28)$$

Applying lemma 1 to (26) gives

$$\begin{bmatrix} \mathbf{I} - \mathbf{P} & \mathbf{0} & \mathbf{X}_k^T \mathbf{U}^T \\ \mathbf{0} & -\mu^2 \mathbf{I} & \mathbf{Z}^T \mathbf{U}^T \\ \mathbf{U} \mathbf{X}_k & \mathbf{U} \mathbf{Z} & \mathbf{P} - \mathbf{U} - \mathbf{U}^T \end{bmatrix} \prec 0, \quad (29)$$

and then substituting:

$$\mathbf{U} \mathbf{X}_k = \mathbf{U} \bar{\mathbf{A}}_k - \mathbf{U} \bar{\mathbf{K}} \bar{\mathbf{C}} = \mathbf{U} \bar{\mathbf{A}}_k - \mathbf{N} \bar{\mathbf{C}}, \quad (30)$$

$$\mathbf{U} \mathbf{Z} = \mathbf{U} \bar{\mathbf{W}} - \mathbf{U} \bar{\mathbf{K}} \bar{\mathbf{V}} = \mathbf{U} \bar{\mathbf{W}} - \mathbf{N} \bar{\mathbf{V}}, \quad (31)$$

completes the proof. \square

Note that, \mathbb{M} specified by (11) can be equivalently expressed by

$$\mathbb{M} = \left\{ \mathbf{M}(\alpha) : \mathbf{M}(\alpha) = \sum_{i=1}^N \alpha_i \mathbf{M}_i, \sum_{i=1}^N \alpha_i = 1, \alpha_i \geq 0 \right\}, \quad (32)$$

where $\mathbf{N} = 2^{n^2}$. Note that, this is a general description, which does not take into account that some elements of \mathbf{M} may be constant. In such cases, \mathbf{N} is given by $\mathbf{N} = 2^{(n-c)^2}$, where c stands for the number of constant elements of \mathbf{M} . Thus, the system can be described in a Linear Parameter Varying (LPV) form. Solving (19) is equivalent to solve (for $i = 1, \dots, N$)

$$\begin{bmatrix} \mathbf{I} - \mathbf{P} & \mathbf{0} & \bar{\mathbf{A}}_i^T \mathbf{U} - \bar{\mathbf{C}}^T \mathbf{N}^T \\ \mathbf{0} & -\mu^2 \mathbf{I} & \bar{\mathbf{W}} \mathbf{U} - \bar{\mathbf{V}}^T \mathbf{N}^T \\ \mathbf{U} \bar{\mathbf{A}}_i - \mathbf{N} \bar{\mathbf{C}} & \mathbf{U} \bar{\mathbf{W}} - \mathbf{N} \bar{\mathbf{V}} & \mathbf{P} - \mathbf{U} - \mathbf{U}^T \end{bmatrix} \preceq 0. \quad (33)$$

and then determine

$$\bar{\mathbf{K}} = \begin{bmatrix} \mathbf{K}_x \\ \mathbf{K}_s \end{bmatrix} = \mathbf{U}^{-1} \mathbf{N}. \quad (34)$$

4 Illustrative example

4.1 Results

To verify the proposed approach a twin-rotor aero-dynamical system (fig. 1) is employed. Such a system is designed to simulate the flight object in laboratory

conditions. The system can be described by highly nonlinear model with cross coupled axes by using following equations:

$$\frac{d\omega_h}{dt} = \frac{k_a k_1}{J_{tr} R_a} u_h - \left(\frac{B_{tr}}{J_{tr}} + \frac{k_a^2}{J_{tr} R_a} \right) \omega_h - \frac{f_1(\omega_h)}{J_{tr}}, \tag{35}$$

$$\begin{aligned} \frac{d\Omega_h}{dt} &= \frac{k_{oh} f_2(\omega_h) \cos(\theta_v) - k_{oh} \Omega_h - f_3(\theta_h) + f_6(\theta_v)}{K_D \cos^2(\theta_v) + K_E \sin^2(\theta_v) + K_F} \\ &+ \frac{k_m \omega_v \sin(\theta_v) \Omega_v (K_D \cos^2(\theta_v) - K_E \sin^2(\theta_v) - K_F - 2K_E \cos^2(\theta_v))}{(K_D \cos^2(\theta_v) + K_E \sin^2(\theta_v) + K_F)^2} \\ &+ \frac{k_m \sin(\theta_v) \left(\frac{k_b k_2}{R_b} u_v - \left(B_{mr} + \frac{k_b^2}{R_b} \right) \omega_v - f_4(\omega_v) \right)}{J_{mr} (K_D \cos^2(\theta_v) + K_E \sin^2(\theta_v) + K_F)}, \end{aligned} \tag{36}$$

$$\frac{d\theta_h}{dt} = \Omega_h, \tag{37}$$

$$\frac{d\omega_v}{dt} = \frac{k_b k_2}{J_{mr} R_b} u_v - \left(\frac{B_{mr}}{J_{mr}} + \frac{k_b^2}{J_{mr} R_b} \right) \omega_v - \frac{f_4(\omega_v)}{J_{mr}}, \tag{38}$$

$$\begin{aligned} \frac{d\Omega_v}{dt} &= \frac{l_m f_5(\omega_v) + k_g \Omega_h f_5(\omega_v) \cos(\theta_v) - k_{ov} \Omega_v}{J_v} \\ &+ \frac{g ((K_A - K_B) \cos(\theta_v) - K_C \sin(\theta_v)) - \Omega_h^2 K_H \sin(\theta_v) \cos(\theta_v)}{J_v} \\ &+ \frac{k_t \left(\frac{k_a k_1}{R_a} u_h - \left(B_{tr} + \frac{k_a^2}{R_a} \right) \omega_h - f_1(\omega_h) \right)}{J_v J_{tr}}, \end{aligned} \tag{39}$$

$$\frac{d\theta_v}{dt} = \Omega_v, \tag{40}$$

where ω_h , Ω_h , θ_h , ω_h , Ω_h and θ_h are the rotational velocity of the tail rotor, angular velocity around vertical axes, yaw angle of the beam, rotational velocity of the main rotor, angular velocity around horizontal axes and pitch angle of the beam, respectively. The system state vector is $\mathbf{x} = [\omega_h^T, \Omega_h^T, \theta_h^T, \omega_v^T, \Omega_v^T, \theta_v^T]^T$ and the sytem input vector is $\mathbf{u} = [u_h^T, u_v^T]^T$ where u_h and u_v are the voltages of the tail and main rotors. The rest of the parameters are inherited from [17].

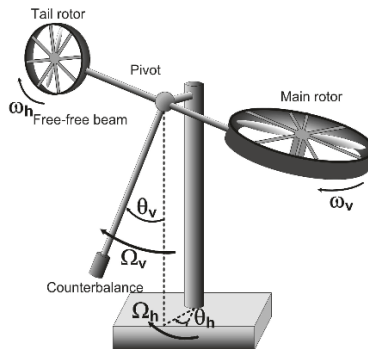


Fig. 1. Twin-rotor aero-dynamical system

The nonlinear model which describes the behaviour of the system has been discretized with sampling time $T_s = 0.01[s]$ which leads to the state-space representation (1)–(2). It is worth to emphasize that the angular velocity around both, vertical and horizontal axes, were not measured directly during the experiment, what implies the output equation featured by $\mathbf{C} = \text{diag}(1, 0, 1, 1, 0, 1)$. Moreover, let us consider a fault scenario $\mathbf{f}_{s,k} = [\mathbf{f}_{s,1,k}^T, \mathbf{f}_{s,3,k}^T, \mathbf{f}_{s,4,k}^T, \mathbf{f}_{s,6,k}^T]^T$ with:

$$\begin{aligned} \mathbf{f}_{s,1,k} &= \begin{cases} \mathbf{y}_{1,k} - 192, & 4000 \leq k \leq 6000, \\ 0, & \text{otherwise,} \end{cases} \\ \mathbf{f}_{s,3,k} &= \begin{cases} \mathbf{y}_{2,k} - 0.2, & 8000 \leq k \leq 10000, \\ 0, & \text{otherwise,} \end{cases} \\ \mathbf{f}_{s,4,k} &= 0, \\ \mathbf{f}_{s,6,k} &= \begin{cases} \mathbf{y}_{4,k} + 0.1, & 9000 \leq k \leq 12000, \\ 0, & \text{otherwise,} \end{cases} \end{aligned} \quad (41)$$

which means that temporary fault occurred in three of four sensors, and the two of them were partly at the same time.

Let us assume that the initial state for the system as well as for the observer are $\mathbf{x}_0 = [0, 0, 0.001, 0, 0, 0.001]$ and $\hat{\mathbf{x}}_0 = [0, 0, 0.01, 0, 0, 0.01]$, respectively, while the fault estimate is initialized by $\hat{\mathbf{f}}_{s,0} = [0, 0, 0, 0]$.

By solving a set of LMIs (33) described in section 3 the following gain matrices has been obtained:

$$\mathbf{K}_x = 10^{-4} \cdot \begin{bmatrix} -0.0360 & 0.0014 & 0.0004 & 0.0000 \\ 0.0053 & -0.0080 & 0.0014 & 0.0272 \\ -0.0020 & 0.1961 & -0.0003 & 0.0048 \\ 0.0016 & -0.0023 & 0.0802 & 0.0081 \\ -0.0031 & 0.0237 & -0.0525 & 0.5181 \\ 0.0026 & 0.0072 & 0.0186 & 0.3602 \end{bmatrix}, \quad (42)$$

$$\mathbf{K}_s = \begin{bmatrix} 0.9987 & 0.0000 & -0.0000 & -0.0000 \\ 0.0000 & 0.9995 & -0.0000 & 0.0000 \\ -0.0000 & 0.0000 & 0.9991 & -0.0000 \\ 0.0000 & 0.0000 & -0.0001 & 0.9996 \end{bmatrix}. \quad (43)$$

4.2 Discussion

Figure 2 presents the response of the system in the faulty case. The black solid line in these graphs represents the system state with FTC and the red dashed line represents the system state controlled by classical controller without fault tolerance. The controlled state variables θ_h and θ_v are presented in the lower left and right graph, respectively. It is easy to see that the system with FTC is following the reference signal (grey line) in contrast to the response without FTC. Figure 3 shows comparisons between real and estimated faults. It is easy to see that the estimated faults are pursuing the real one highly satisfactory. The obtained results confirm that the FTC allows to control the system correctly even in the case when the faults occurred, contrarily to the results obtained without fault tolerance.

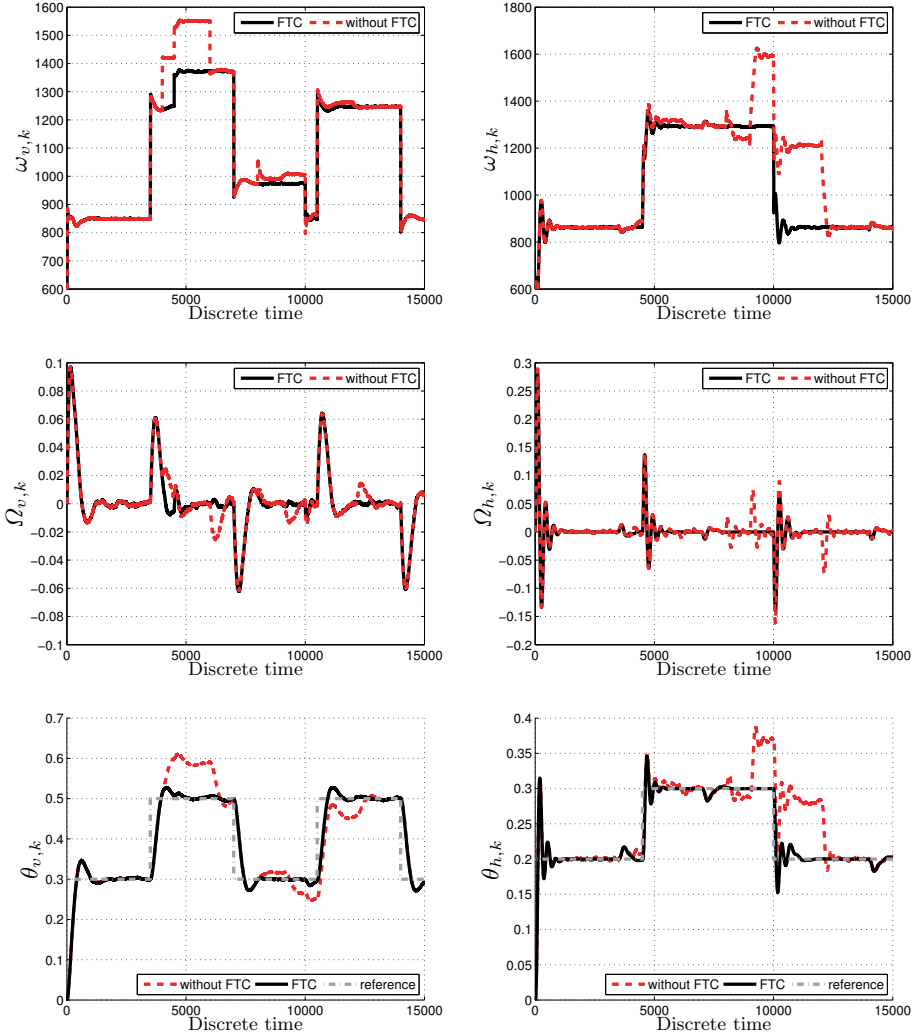


Fig. 2. The response of the controlled system

5 Conclusions

The main aim of the paper was to handle with the problem of control the system while the sensor faults occur. A strategy for simultaneous estimation of the state and the fault was proposed. The presented observer was able to estimate the fault for all of the sensors in the system simultaneously and thanks to that it is easy to compensate the fault influence. The strategy was based on \mathcal{H}_∞ approach which brings down to solve a set of LMIs. It can be easily used in fault diagnosis for linear as well as nonlinear LPV-like systems. The final part of the paper shows

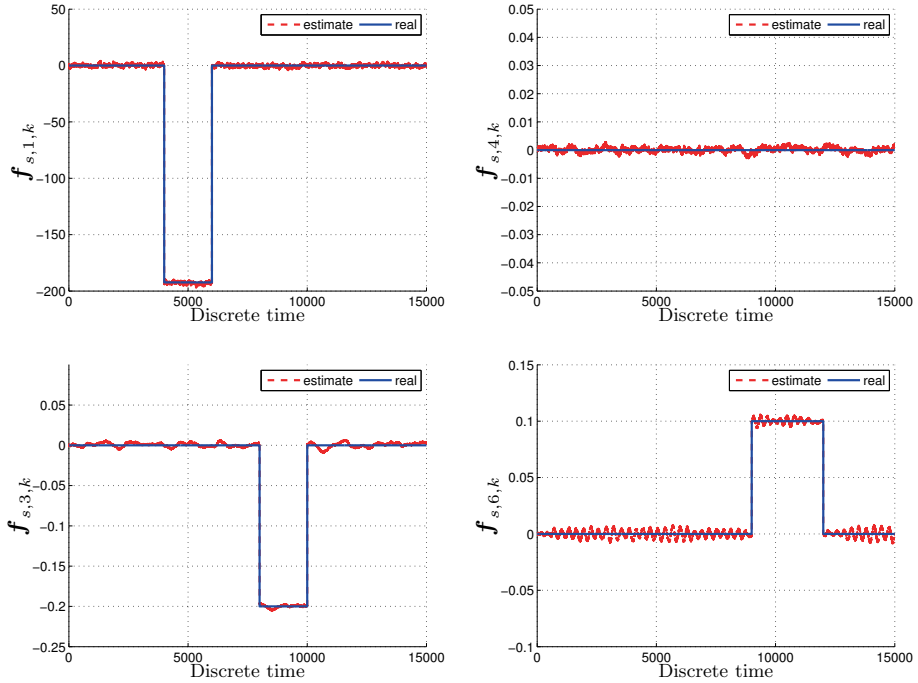


Fig. 3. The sensors faults and their estimates

an illustrative example with an application to the twin-rotor aero-dynamical system. The achieved results confirm the correctness of the proposed approach.

Acknowledgements

The work was supported by the National Science Centre of Poland under grant: 2013/11/B/ST7/01110.

References

1. Mogens Blanke, Michel Kinnaert, Jan Lunze, Marcel Staroswiecki, and J Schröder. *Diagnosis and fault-tolerant control*, volume 2. Springer, 2006.
2. A. Brahim, S. Dhahri, F. Hmida, and A. Sellami. An h_∞ sliding mode observer for Takagi–Sugeno nonlinear systems with simultaneous actuator and sensor faults. *International Journal of Applied Mathematics and Computer Science*, 25(3):547–559, 2015.
3. J. Chen and R.J. Patton. *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Kluwer Academic Publishers, 1999.
4. Lejun Chen, Ron Patton, and Philippe Goupil. Robust fault estimation using an lpv reference model: Addsafe benchmark case study. *Control Engineering Practice*, 49:194–203, 2016.

5. M. C. de Oliveira, J. Bernussou, and J. C. Geromel. A new discrete-time robust stability condition. *Systems & control letters*, 37(4):261–265, 1999.
6. Guillaume JJ Ducard. *Fault-tolerant flight control and guidance systems: Practical methods for small unmanned aerial vehicles*. Springer Science & Business Media, 2009.
7. Steven Gillijns and Bart De Moor. Unbiased minimum-variance input and state estimation for linear discrete-time systems. *Automatica*, 43(1):111–116, 2007.
8. Rolf Isermann. *Fault-diagnosis applications: model-based condition monitoring: actuators, drives, machinery, plants, sensors, and fault-tolerant systems*. Springer Science & Business Media, 2011.
9. Qingxian Jia, Huayi Li, Yingchun Zhang, and Xueqin Chen. Robust observer-based sensor fault reconstruction for discrete-time systems via a descriptor system approach. *International Journal of Control, Automation and Systems*, 13(2):274–283, 2015.
10. JY Keller and Mohamed Darouach. Two-stage kalman estimator with unknown exogenous inputs. *Automatica*, 35(2):339–342, 1999.
11. J Korbicz, JM Koscielny, Z Kowalczyk, and W Cholewa. Fault diagnosis. models, artificial intelligence, applications, 2004.
12. H. Li and M. Fu. A linear matrix inequality approach to robust H_∞ filtering. *IEEE Transactions on Signal Processing*, 45(9):2338–2350, 1997.
13. Mufeed Mahmoud, Jin Jiang, and Youmin Zhang. *Active fault tolerant control systems: stochastic analysis and synthesis*, volume 287. Springer Science & Business Media, 2003.
14. Euripedes G Nobrega, Musa O Abdalla, and Karolos M Grigoriadis. Robust fault estimation of uncertain systems using an lmi-based approach. *International Journal of Robust and Nonlinear Control*, 18(18):1657–1680, 2008.
15. Hassan Noura, Didier Theilliol, Jean-Christophe Ponsart, and Abbas Chamseddine. *Fault-tolerant control systems: Design and practical applications*. Springer Science & Business Media, 2009.
16. Ron J Patton, Paul M Frank, and Robert N Clark. *Issues of fault diagnosis for dynamic systems*. Springer Science & Business Media, 2013.
17. D. Rotondo, F. Nejjari, and V. Puig. Quasi-lpv modeling, identification and control of a twin rotor mimo system. *Control Engineering Practice*, 21(6):829–846, 2013.
18. Lothar Seybold, Marcin Witczak, Paweł Majdzik, and Ralf Stetter. Towards robust predictive fault-tolerant control for a battery assembly system. *International Journal of Applied Mathematics and Computer Science*, 25(4):849–862, 2015.
19. Kalyana C Veluvolu, MY Kim, and Dongik Lee. Nonlinear sliding mode high-gain observers for fault estimation. *International Journal of Systems Science*, 42(7):1065–1074, 2011.
20. M. Witczak. *Fault Diagnosis and Fault-Tolerant Control Strategies for Non-Linear Systems: Analytical and Soft Computing approaches*. Springer International Publishing, Heidelberg, Germany, 2014.
21. A. Zemouche and M. Boutayeb. Observer design for Lipschitz non-linear systems: the discrete time case. *IEEE Trans. Circuits and Systems - II: Express Briefs*, 53(8):777–781, 2006.
22. A. Zemouche, M. Boutayeb, and G. I. Bara. Observers for a class of lipschitz systems with extension to H_∞ performance analysis. *Systems & Control Letters*, 57(1):18–27, 2008.
23. Xiaodong Zhang, Marios M Polycarpou, and Thomas Parisini. Fault diagnosis of a class of nonlinear uncertain systems with lipschitz nonlinearities using adaptive estimation. *Automatica*, 46(2):290–299, 2010.

The workspace of industrial manipulator in case of fault of the drive

Krzysztof Jaroszewski

West Pomeranian University of Technology, Szczecin,
ul. 26 Kwietnia 10, 71-126 Szczecin, Poland
kjaroszewski@zut.edu.pl

Abstract. In the paper very preliminary idea of fault tolerant industrial robot is presented. It is noticed that, the most natural way of fault tolerance - redundancy, is not appropriate in case of the robotics area. Using on the production line redundant robots (treated overall as the actuators) is unprofitable from economic point of view. Moreover, redundancy of the actuators (seeing as the parts of the robot) is mostly impossible from construction point of view. Hence, the solution of fault tolerance is offered in the way of adequate control system. In the first chapter problem statement is done and the general idea of solving it is presented. Following, idea of control system is briefly mentioned. In the next unit simple graphs presents the robot's workspace in selected case of the fault. The wider idea of fault tolerance control in case of industrial manipulator is given in the summary.

Keywords: manipulator, fault, industrial manipulator workspace, diagnostics

1 Introduction

There is well known and understandable that Fault Tolerant Systems (FTS) are highly demanded in any case. Especially industry expects such solutions, regardless of the branch, due to the fact that such solutions let them to save a money and a time. This is why the FTS and Fault Tolerant Control (FTC) subjects are still under research. Of course, the most popular way of practising FTC is redundancy. It means the redundant (auxiliary) sensors and / or actuators are used. Not only sensors and actuators may be redundant but also control system, information and communication system, generally each part of the process could have its replacement. Inasmuch, FTC algorithms concerning sensors and actuators are less complicated in implementation than other components of the control system. Wherefore, redundancy based on actuators and especially on sensors is the most encountered. On the other hand, such approach, using redundant sensors and actuators, is expensive and require to be taken into consideration during object's design process. Hence, solution based on devices with FTC should be treated as a much more appropriate approach. It means that e.g. instead of using two redundant devices (seeing as an actuator) is used only one with implementing FTC. Using such device on production line also may result in saving a time

and a money. Nevertheless, it is obvious that efficiency of such device under fault may be lower in comparison with efficiency of the device in case of normal condition of operation. Taking into account production line with robots it is in many cases impossible and always irrational, from economical point of view, to design process with redundant robots. However, still in such case tolerance for faults is highly dared. Even higher demanding for fault tolerance is in case of robots use in outer space, were there are no chance to repair faulty device and the only chose is to do task with the lower efficiency or not to do it at all. Very similar issue is in case of medical robots. It is hard to imagine that the robot could be repaired during the surgery. It is rather the surgery is stopped. However sometimes it is impossible just to break the surgery process, it would rather be in some way finished. What can be done in such case is to use robots with FTC implemented into. Of course, such solution is not robust in case of each potential damage but for some selected it could works. What is the most important, it is assumed that mentioned above robot is regular one, taking into consideration its mechanical construction, however it only has control system dedicated for fault maintain.

It is well known that there are no FTC without proper diagnostics. For that reason the issue of adequate diagnostic should be taken into account at first, many publication deal with the diagnostic subject, e.g. [1–4] for case of the industrial area of implementation. As the diagnostics issue depends on the branch of the industry there are peculiar methods for specific processes and devices, it is easy to find publication corresponding to it, e.g. in the area of robotics: [5–7]. Diagnostic process may be leaded in the purpose of object stoppage [8]: instantly or in specific way or to predict faulty state [9] of the object as well as to start FTC procedure [10–13].

2 Control system concept

Classical implementation of robot control system boils down to definig series of point coordinates to be achieve by the robot grip and a type of robot arm motion between that points. However, all points may be achivable in some faulty states and in non-faulty state of the robot, the way of achieving points may be different due to the fuctionalty state of the robot. Hence, assuming, that there is possibility to switch between programs downloaded to the robots control system we may achieve tolerance for some actuators' faults by choosing appropriate program, prepared in advance, for the predicted faults. The idea of the fault tolerant control system architecture was presented e.g. in [14]. Of course, it is impossible to predict all faults with their parameters. It means, e.g. we may predict jam of the junction robots arms, however it could happen at different angle between the arms. Newetheless, assuming that we have properly working mesurement system we may, thoughtly, treat faulty robot (e.g. with fixed two arms) as a another one robot of known slightly different mechanical construction. This is why the best idea is not to send to the robot's control system compiled program with commends and coordinates of points to achieve but only points

coordinates and definition of obstacle coordinates. In such case the program should be calculated in the way it takes into account different construction of the robot seeing as the one of the junction is fixed. Till now we obtain the new one arm created from a fixed junction of two arms. The simply algorithm of the idea is presented on Fig. 1.

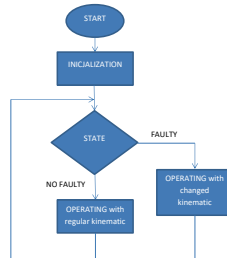


Fig. 1. The simply algorithm of the idea

3 Faulty type - fixed junction

Taking into account classical construction of the serial manipulator of 6 degree of freedom (DoF) it is rather obvious that fixed one of the first three junction is much more difficult to tolerate then fixed junction among the rest. In the most construction of such robot first three arms are dedicated for long distance motion and the rest three for rather smaller ones. Hence, the idea of solving fixed junction fault will be presented when junction no 5 will be fixed.

3.1 Graphical presentation of the solution idea

For the purpose of that paper the Staubli robot TX60 was taken into account. Its parameters are presented on Fig. 2. The simply presentation of the robot workspace, assuming that only junction no 2 and no 5 are taken into consideration is presented on Fig. 3, 4, 5, with angle change resolution 25 deg., 5 deg. and 1 deg. respectively. Let assume that the junction no 5 may be fixed with the angle (measurably accessible) equal -89 deg. An illustration of workspace such faulty robot is presented on Fig. 6, with 5 deg. change of no 2 junction.

It is well know that due to multiple of invers kinematic solutions dared position and orientation of the grip may be achieved for different joints arrangements. It gives a chance to achieve some grip position and orientation even in case of

CHARACTERISTICS	TX60	TX6K
Maximum load ¹⁾	9 kg	3 kg
Rated load	3.5 kg	2 kg
Reach between axis 1 and 2	400 mm	400 mm
Number of degrees of freedom	6	5
Responsibility ISO 9283	± 0.02 mm	± 0.03 mm
Serial USB interface controller	Yes	Yes
Weight	31.4 kg	17.3 kg

RANGE OF MOTION	
Axis 1 (X)	± 180°
Axis 2 (R)	± 180.25°
Axis 3 (Z)	± 140.25°
Axis 4 (W)	± 180°
Axis 5 (E)	± 180.25°
Axis 6 (F)	± 270°

WORK ENVELOPE	
Maximum reach between axis 1 and 2 (R, W)	400 mm
Minimum reach between axis 1 and 2 (R, W)	400 mm
Minimum reach between axis 1 and 3 (R, Z)	180 mm
Minimum reach between axis 2 and 3 (R, Z)	180 mm
Reach between axis 3 and 5 (R, E)	210 mm

MAXIMUM VELOCITY	
Axis 1	450°/s
Axis 2	450°/s
Axis 3	360°/s
Axis 4	900°/s
Axis 5	360°/s
Axis 6	1440°/s
Maximum speed of load gravity center	0.1 m/s
Maximum inertia Axis 3	0.265 kg·m ²
Maximum inertia Axis 5	0.023 kg·m ²
Stiction	All axes

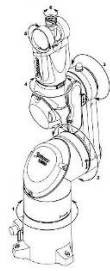


Fig. 2. Staubli TX 60 - parameters

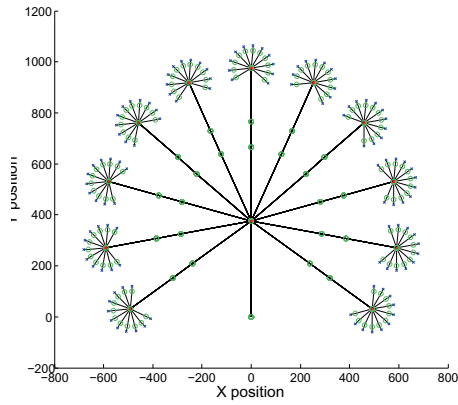


Fig. 3. Staubli TX 60 - workspace in case of 25 deg. change of no 2 and no 5 junction

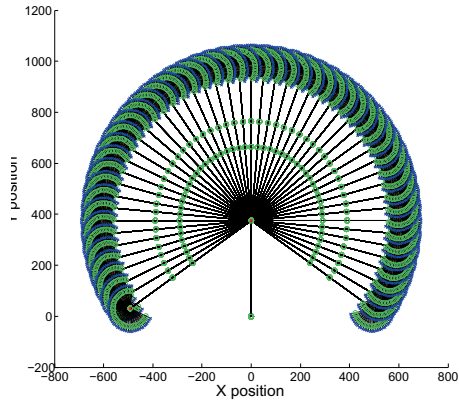


Fig. 4. Staubli TX 60 - workspace in case of 5 deg. change of no 2 and no 5 junction

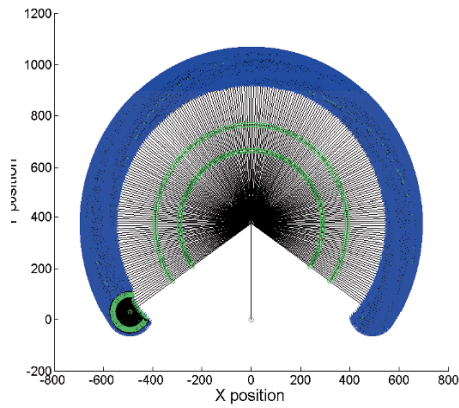


Fig. 5. Staubli TX 60 - workspace in case of 1 deg. change of no 2 and no 5 junction

fixed junction no 5. What is more, in the same plan it is possible to use junction no 3 to compensate dysfunction connected with fixed junction no 5. Let assume that junction no 5 fixed with 89 deg. angle. In case of no faulty state of manipulator the dared position (X:89.9, Y:97.7) may be achieved in manipulator's arm configuration as presented on Fig. 7. The same position and orientation of grip may be achieved with different junction configuration as presented e.g. on Fig. 8. Both cases, presented manipulator arm configuration in faulty and no-faulty junction no 5 is presented on Fig. 9.

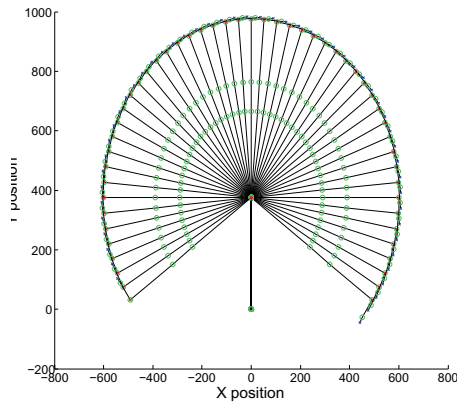


Fig. 6. Staubli TX 60 - workspace in case of 25 deg. change of no 2 junction and fixed on 89 deg. junction no 5

Of course, two cases should be discussed: 1. when in the workspace there are no obstacles and 2. when in the workspace is defined obstacle. However, in the paper the only first one was mentioned. Nevertheless, presented example shows that even in some faulty states manipulator still may be treated as a fit for operating.

4 Summary

The paper was prepared to show the idea of coping with the fault of one (or more) manipulator arms junction fixed. Due to the fact that there is multiple

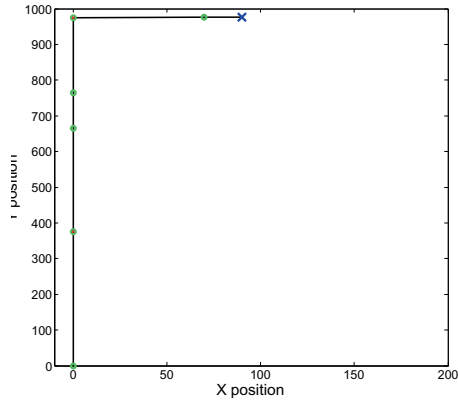


Fig. 7. Staubli TX 60 - the manipulator achieving assumed position X:89.9 Y:977 with all junction not faulty

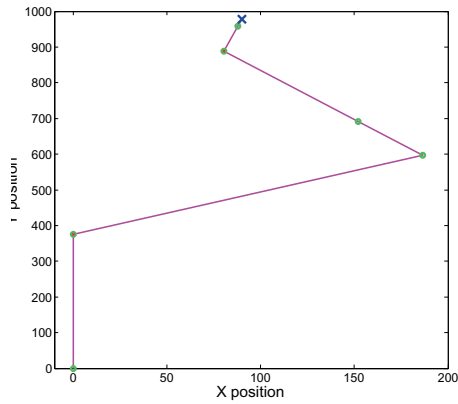


Fig. 8. Staubli TX 60 - the manipulator achieving assumed position X:89.9 Y:977 with junction no 5 fixed on -89 deg.

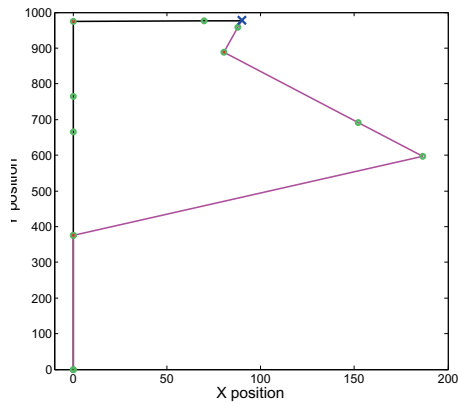


Fig. 9. Staubli TX 60 - the manipulator achieving assumed position X:89.9 Y:977 - comparison: faulty / not faulty junction no 5

of invers kinematic solutions it may be possible to continue the operation with faulty manipulator (junction fixed). Presented solution is based on the assumption that manipulator firmware is flexible and allow for switching between control algorithms. It means that in the control algorithm of the manipulator there is no one constant description of the manipulator kinematic but some parameters may be changed due to the current manipulator state (e.g. in case of fault). Suggested solution is possible to use only in case of possessing information about angle of fixed junction. The much more complicated task is in case of obstacles occurring in the workspace.

References

1. Korbicz J., Kościelny J.M., Kowalczyk Z., Cholewa W., Fault Diagnosis: Models, Artificial Intelligence, Applications, *Springer*, Berlin, (2004)
2. Korbicz J., Kościelny J.M., Modeling, Diagnostics and Process Control: Implementation in the DiaSter System, *Springer*, Berlin, (2010)
3. Kościelny J.M., Praktyczne problemy diagnostyki procesów przemysłowych, *Pomiary Automatyka Robotyka*, vol. 2/2010, p.p. 115-134, (2010)
4. Chiang L.H., Russel E.L., Braatz R.D., Fault detection and diagnosis in industrial systems, *Springer*, London, (2001)
5. Honghai Liu, George M. Coghill, A model-based approach to robot fault diagnosis, *Knowledge-Based Systems*, vol. 18, Issues 45, August 2005, Pages 225-233 AI-2004, Cambridge, England, 13th-15th December 2004, (2004)
6. Lipsett M.G., Robot Looseness Fault Diagnosis, *A thesis submitted to the Department of Mechanical Engineering in conformity with the requirements for the degree of Doctor of Philosophy Queen's University Kingston, Ontario, Canada*, July, (1995)
7. Xingyan Li, Lynne E. Parkerl, Sensor Analysis for Fault Detection in Tightly-Coupled Multi-Robot Team Tasks, *Proc. of IEEE International Conference on Robotics and Automation*, Rome, Italy, (2007)
8. Baerveldt A.J., Cooperation between man and robot: interface and safety, *Proceedings. IEEE International Workshop on Robot and Human Communication*, (Cat. No.92TH0469-7), IEEE Xplore Digital Library, (1992)
9. Warren E. Dixon, Ian D. Walker, Darren M. Dawson, John P. Hartranft, Fault Detection for Robot Manipulators with Parametric Uncertainty: A Prediction-Error-Based Approach *Elsevier*, IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 16, NO. 6, DECEMBER, (2000)
10. Didier Crestani, Karen Godary-Dejean, Fault Tolerance in Control Architectures for Mobile Robots: Fantasy or Reality?, <https://hal-lirmm.ccsd.cnrs.fr/lirmm-00804370>
11. Noore A., Real time fault tolerant control of robot manipulators, Mathematical and Computer Modelling,, *Elsevier*, vol. 38, Issues 12, July 2003, Pages 13-22, (2003)
12. Siqueira Adriano Almeida Goncalves, Terra Marco H., Bergerman Marcel, Robust Control of Robots. Fault Tolerant Approaches, *Springer-Verlag*, London (2011)
13. Claudio Bonivento, Luca Gentili and Andrea Paoli, Internal model based fault tolerant control of a robot manipulator, *Proc. of 43rd IEEE Conference on Decision and Control*, December 14-17, 2004, Atlantis, Paradise Island, Bahama, (2004)
14. Heejune Ahn, Dong-Su Lee, Ahn Sang Chul, A hierarchical fault tolerant architecture for component-based service robots, *Proc. of 8th IEEE International Conference on Industrial Informatics*, IEEE Xplore Digital Library, (2010)

Determining and verifying the safety integrity level with security aspects

Emilian Piesik and Marcin Śliwiński

Gdańsk University of Technology,
str. Gabriela Narutowicza 11/12, 80-233 Gdańsk, Poland,
emilian.piesik@pg.gda.pl, marcin.sliwinski@pg.gda.pl,
WWW home page: <http://eia.pg.edu.pl>

Abstract. Safety and security aspects consist of two different group of functional requirements for the control and protection systems. It is the reason why the analyses of safety and security shouldnt be integrated directly. The paper proposes extension of the currently used methods of functional safety analyses. It can be done with inclusion of the level of information security assigned to the technical system. The article addresses some important issues of the functional safety analysis, namely the safety integrity level (SIL) verification of distributed control and protection systems with regard to security aspects. A method based on quantitative and qualitative information is proposed for the SIL (IEC 61508, 61511) verification with regard of the evaluation assurance levels (EAL) (ISO/IEC 15408) and the security assurance levels (SAL) (IEC 62443).

Keywords: risk analysis, functional safety methodology, security

1 Introduction

The role of safety-related control and protection systems for the risk reduction is nowadays obvious, because are designed to reduce the risks of accident scenarios, especially those with major consequences many times, e.g. from ten times to thousand and more times depending on required risk mitigation. These systems belong to the category of industrial control systems (ICS).

They implement a set of safety functions and can be designed as the electrical / electronic / programmable electronic systems (E/E/PES) regarding generic standard IEC 61508 and/or the safety instrumented systems (SIS) with regard to requirements of IEC 61511 developed for the process industry. Some more important safety functions, reducing substantially relevant risks, require implementing of protection layers, according to a concept of defence in depth (DinD). Requirements concerning security related aspects will be considered regarding requirements of series of international standards IEC 62443 an ISO 27000.

An integrated risk analysis and assessment methodology proposed is compatible with some known methods used often in practice, such as HAZOP (hazard

and operability), LOPA (layer of protection analysis) and SVA (security vulnerability analysis).

Security related analyses of the ICS during its design and operation as distributed computer system (DCS) with relevant SCADA (supervisory control and data acquisition) functions are very important in hazardous plants especially when they are considered within critical infrastructure (CI).

2 Classification of the process control and protection systems

A conventional control and protection system consists of programmable logic controller (PLC), sensors, actuators, control station with supervisory control and data acquisition (SCADA) and the control station. Another important element of the control and protection system is the human operator who is supervising its operation.

The systems elements may be connected by different internal or external communication channels. The information sent between PLC and the control station can be transferred by standard series or parallel communication protocols or other methods of communication, e.g. wireless GSM/GPRS.

Three main categories of distributed control and protection systems were proposed, based on the presence of computer system or industrial network, its specification and type of data transfer methods [4, 18, 3, 21, 22, 24]:

I. Systems installed in concentrated critical objects using only the internal communication channels (e.g. local network LAN),

II. Systems installed in concentrated or distributed critical plants, where the protection and monitoring system data are sent by internal communication channels and can be sent using external channels,

III. Systems installed in distributed critical instalations, where data are sent mainly by external communication channels.

A IEC 61508:2010 (new IEC 61511:2015) introduces some additional requirements concerning the data communication channels and security aspects in functional safety solutions. It describes two main communication channel types white or black one. A white channel means that the entire communications channel is designed, implemented and validated according to IEC 61508 requirements. The black one means that some parts of communication channel are not designed, implemented and validated according to IEC 61508. In that case, communication interfaces should be implemented according to the railway applications communication, signalling and processing systems IEC 62280 standard (Safety-related communication in closed transmission systems) [1, 2, 19, 23, 10].

3 Determining required safety integrity level security with security aspects

One of the main purpose of the functional safety analysis is the determination of safety integrity level (SIL) for a given safety-related function, which is to be

implemented by the control and/or protection systems that are usually based on programmable electronic systems. They are playing an important role in many applications, including the control and protection of hazardous installations. However, a failure or incorrect operation of such critical elements, controlling and/or protecting an industrial system could lead to serious injury or even the death of one or more people. In some cases it can lead to a significant environmental damage or property loss too. That's the reason why the risk analysis of the E/E/PE systems is so important.

In the new extended approach was introduced in, based on modifiable risk graphs, which allows building any risk graph schemes with given number of the risk parameters and their ranges expressed qualitatively or preferably semi-quantitatively [10]. An example of functional safety analysis will be presented below. It is based on a control system (shown in Fig. 1), which consists of some basic components like sensors, transmitters, programmable logic controllers and valves. It is a part of oil fluid receiving system from the wellhead. The well fluid is heated in the preheater and then, after pressure reduction process, to the main heater and a separator. The additional bypass is provided to allow temperature control and maintain constant temperature of fluid.

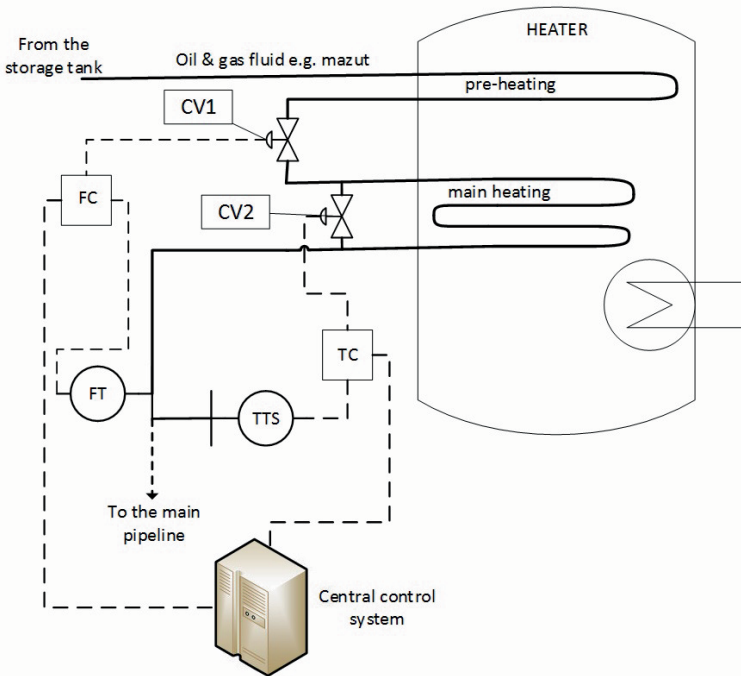


Fig. 1. An example of the control system

The functional safety analysis relies on information taken from a process of hazard identification as well as further risk assessment for designed or existing basic process control system. Some factors influence the frequency and some are responsible for consequences. The frequency parameter is basically associated with reliability of the control system equipment and human factors. The security aspects, which are associated with e.g. communication between equipment or restrictions in access to the system and associated assets, are usually omitted during this stage of analysis. However, they can significantly influence the final results. So, there should be a simple but effective method that allows quickly append those aspects into typical functional safety analysis. It is very important especially in analyses of complex, distributed control systems. Proposed method is based on extended risk graph method. A conventional set of four risk parameters in the risk graph method is: the consequence (C1), the frequency and exposure time (F1), the possibility of failing to avoid hazard (F2) and the probability of unwanted occurrence of potential events that demand the operation of given E/E/PE safety-related system (F3). A required SIL for a chosen safety function will be a result of the functional safety analysis performed.

Taking into consideration the extended version of the risk graph method the modifiable risk graph and a knowledge-based system associated with it [1, 5, 7, 25], it can be possible to include some security analysis results to the risk graph, in case of using industrial internal and/or external network in the control system. In almost each technical system some vulnerabilities can exist and can create additional risk to human, environment and/or assets. The security analysis helps finding the weak points in communication system and designing appropriate countermeasures. Taking into account the functional safety aspects, identified vulnerabilities and implemented countermeasures can change the level of assessed security and affect the required value of SIL (e.g. SIL1). Having the result of security analysis for given control system, it can be divided into some general categories, for example a qualitative descriptive ranges like: low level of security, medium level of security or high level of security.

The security analysis concept is proposed in the standard ISO/IEC 15408 [16]. Security is considered with the protection from threats, where threats are categorized as the potential for abuse of assets. All categories of threats should be considered, but in the domain of security usually greater attention is given to those threats that are related to malicious or other human intentional activities. The Evaluation Assurance Level (EAL) is a package of assurance requirements, which covers the complete development of a product with a given level of strictness. Common Criteria lists seven levels, with EAL1 being the most basic (cheapest to evaluate and implement) and EAL7 being the most strict (most expensive). Higher EAL levels do not necessarily imply better security, they only mean that the claimed security assurance of the TOE (target of evaluation) has been more extensively validated.

The evaluation process establishes a level of confidence that the security functions of such products and systems and the assurance measures applied to them meet these requirements. The evaluation results may help the developers

and users to determine whether the product or system is secure enough for their intended application and whether the security risks implicit in its use are tolerable [8, 9, 16, 27, 10]. If the security analysis is performed on the basis of [16], the corresponding EAL (evaluation assurance level) should be determined. In this case this EAL can be taken into account in functional safety analysis too (see Table 1).

Table 1. Levels of security and corresponding EALs

Evaluation assurance level	Level of security	Risk parameter range
EAL1	Low level	F_1^3
EAL2	Low level	F_1^3
EAL3	Medium level	F_2^3
EAL4	Medium level	F_2^3
EAL5	High level	F_3^3
EAL6	High level	F_3^3
EAL7	High level	F_3^3

The modifiable risk graph built with additional risk parameter F3 corresponding to the level of security is presented in Fig. 2. The proper calibration of such risk graph give an opportunity to increase the requirements to the E/E/PE safety-related system, which will implement safety function, in case of too low security level of analyzed system. That means, the less secure system is, the frequency of unwanted dangerous accident increases. In that case the frequency is dependent not only on reliability of equipment but also on malicious actions taken by potential attacker, which can lead to a dangerous situation and potentially to an accident. In response to this problem the safety-related system should be more reliable, so it has to fulfill more restrictive requirements (higher SIL). It means that in given system (Fig. 1), which has the level of security determined, the basic safety integrity level requirement (SIL1) should be increased in case of low security analysis results as follows:

- SIL1 for high level of control system security,
- SIL2 for medium level of security,
- SIL3 for low level of security.

Taking into account the situation when security of control system is low, the necessary risk reduction to the tolerable level have to be higher, so the E/E/PE safety-related system implementing safety function, should fulfill more rigorous requirements (e.g. SIL2 or SIL3). Another approach for security evaluation for industrial automation and control systems is IEC 62443 [13]. A concept of Security Assurance Level (SAL) has been introduced in this normative document. There are four security levels (SAL1 to 4) and they are assessed for given security zone using the set of 7 functional requirements. The SAL is a relatively new security measure concerning the control and protection systems. It is evaluated

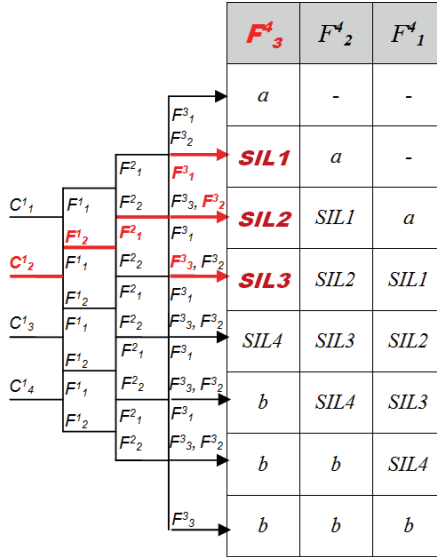


Fig. 2. Example of extended risk graph overflow

based on a defined vector of seven requirements for relevant security zone:

$$SAL = \{AC \ UC \ DI \ DC \ RDF \ TRE \ RA\} \tag{1}$$

where: AC - identification and authentication control, UC - use control, DI - data integrity DC - data confidentiality, RDF - restricted data flow, TRE - timely response to event, RA - resource availability.

Another method of the security analysis can be proposed on the basis of the SeSa (SecureSafety) approach, which was designed by the Norwegian research institution SINTEF [11]. It is dedicated to the control systems and automatic protection devices used in the offshore installations, monitored and managed remotely from the mainland by generally available means of communication [6]. The Safety Instrumented Systems (SIS) according to the series of standards IEC 61508 and IEC 61511 are very important not only for the safety, but also security aspects should be also taken into account [14, 15].

Using the SeSa rings related to security protection is another approach useful for the integration of functional safety and security aspects (Fig. 3). An important task of integrated functional safety and security analysis of such systems is the verification of required SIL taking into account the potential influence of described above security levels, described the EAL, SAL or SeSa protection rings.

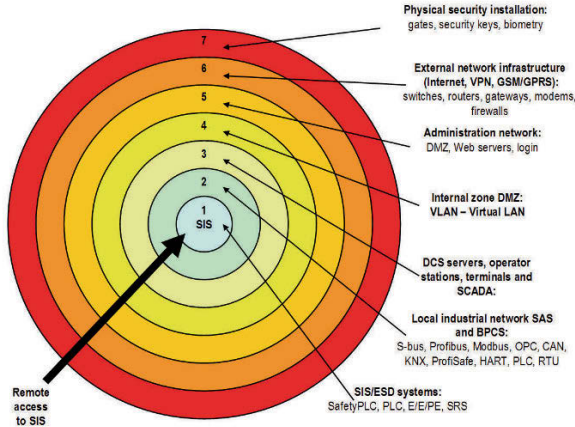


Fig. 3. Rings of the protection in SIS system overflow

4 Verification SIL with security aspects

Taking into account the exemplary control system (see Fig. 1) and the possible accident scenarios associated with him, the safety-related function can be introduced. In this particular case a SIF related to the pressure increase hazardous scenario will be taken into further consideration.

Having a required SIL for this safety-related function, a proper architecture of SIS should be designed. After this step, the proposed architecture have to be verified, i.e. checked if it fulfills requirements. The process of SIL verification, similarly like SIL determination, usually does not include security aspects. An important task of integrated functional safety and security analysis of such systems is the verification of required SIL taking into account the potential influence of described above security levels, described the EAL, SAL or SeSa protection rings. The SIL is associated with safety aspects while the EAL, SAL and SeSa is concerned with level of information security of entire system performing monitoring, control and/or protection functions (see Table 3).

It is possible that undesirable external events or malicious acts may influence the system by threatening to perform the safety-related functions in case of low security level. Thereby the low level of security might reduce the safety integrity level (SIL) when the SIL is to be verified. Thus, it is important to include security aspects in designing and verifying the programmable control and protection systems operating in an industrial network.

From the risk assessment the safety integrity level for given safety function overpressure protection pipeline was determined as SIL3. In industrial practice such level requires usually to be designed using a more sophisticated configuration. Safety function (overpressure protection) is implemented in distributed safety instrumented system (see Figure 5).

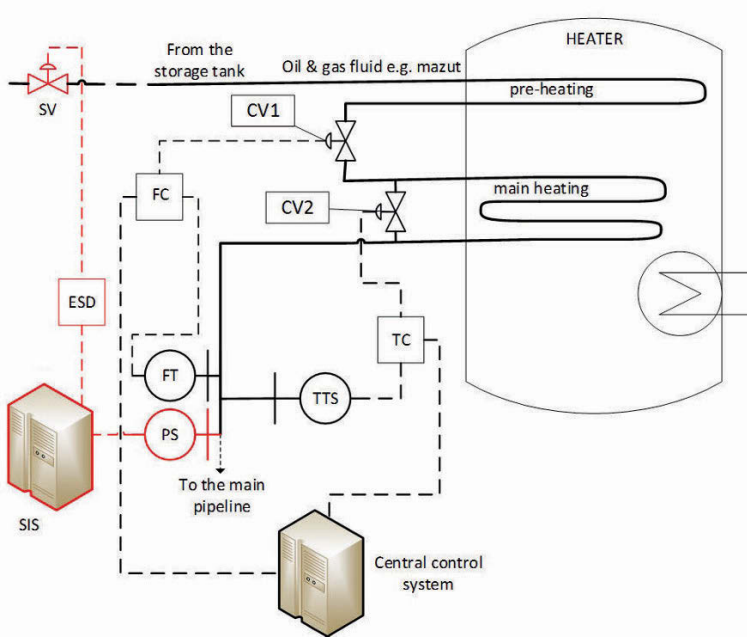


Fig. 4. An example of the control and protection system with safety instrumented system

Table 2. Reliability data for elements SIS system

	PS	CM	PLC	SV
DC [%]	54	90	66	24
λ_{DU} [1/h]	$3 \cdot 10^{-7}$	$1 \cdot 10^{-7}$	$5 \cdot 10^{-6}$	$8 \cdot 10^{-7}$
T_I [h]	8760	8760	8760	8760
β	0.02	0.01	0.01	0.02

It has to be verified in the process of probabilistic modeling, taking into account its subsystems including networks. Reliability data for SIS elements are presented in Table 2. For given system a proper architecture is considered to meet the SIL requirement for entire system. The communication channel is created by serial link of relevant subsystems. Therefore, its reliable operation is dependent on correct functioning of each subsystem. Assessment of the result obtained shows that for the SIS structure on Figure 5 is:

$$PFD_{avgSIS} \cong PFD_{PS(2oo3)} + PFD_{avgCM} + PFD_{PLC(1oo2)} + PFD_{SV(1oo2)} \quad (2)$$

where: PFD_{avgSIS} - average probability of failure on demand for the SIS system, $PFD_{PS(2oo3)}$ - for the pressure sensor, PFD_{avgCM} - average probability of failure on demand for the communication module, $PFD_{PLC(1oo2)}$ - for the PLC, $PFD_{SV(1oo2)}$ - for the safety valve.

Thus, the PFD_{avg} is equal $9,215 \cdot 10^{-4}$ fulfilling formally requirements for random failures on level of SIL3. The omission of some subsystems or communication network can lead to too optimistic results, particularly in case of distributed control and protection systems of category II and III [20, 26].

The SIL is associated with safety aspects while the EAL, SAL and SeSa is concerned with level of information security of entire system performing monitoring, control and/or protection functions (see Table 3).

It is possible that undesirable external events or malicious acts may influence the system by threatening to perform the safety-related functions in case of low security level. Thereby the low level of security might reduce the safety integrity level (SIL) when the SIL is to be verified. Thus, it is important to include security aspects in designing and verifying the programmable control and protection systems operating in an industrial network.

Table 3. SIL that can be claimed for given EAL, SAL or SeSa protection rings for systems of category II and (III)

Determined security				Verified SIL for II cat. (III cat.) functional safety			
EAL	SAL	Protection rings	Level of security	1	2	3	4
1	1	1	LOW	-(-)	SIL1 (-)	SIL2 (1)	SIL3 (2)
2	1	1		-(-)	SIL1 (-)	SIL2 (1)	SIL3 (2)
3	2	2	MEDIUM	SIL1 (-)	SIL2 (1)	SIL3 (2)	SIL4 (3)
4	2	4		SIL1 (-)	SIL2 (1)	SIL3 (2)	SIL4 (3)
5	3	5	HIGH	SIL1 (1)	SIL2 (2)	SIL3 (3)	SIL4 (4)
6	4	6		SIL1 (1)	SIL2 (2)	SIL3 (3)	SIL4 (4)
7	4	7		SIL1 (1)	SIL2 (2)	SIL3 (3)	SIL4 (4)

Example of the control and protection system in critical infrastructure is shown in Figure 1. In this example a SIF was defined related to control and reduce potential overpressure for hazardous scenario considered. Having a required SIL for this safety-related function, a proper architecture of SIS can be designed. After this analysis, the proposed architecture has to be verified, i.e. checked if it fulfills specified requirements. The process of SIL verification, similarly like SIL determination, usually doesnt include in industrial practice the security aspects.

But when SIS uses some communication channels this problem should be taken into account. Such SIS system is presented in Figure 4. In such case there is a challenge to include security aspects in designing and verifying SIL of the programmable control and protection system operating in a network that implements given safety function.

An integrated approach is proposed, in which determining and verifying safety integrity level (SIL) with levels of security (EAL, SAL and SeSa) is related to the system category (I, II or III). It is possible that undesirable external events and malicious acts may impair the system by threatening to perform the safety-related functions in case of low security level. Such integrated approach is

necessary, because not including security aspects in designing safety-related control and/or protection systems operating in network may result in deteriorating safety (lower SIL than required). In such cases the SIL verification, integrated with security aspects, is necessary as shown in Figure 6.

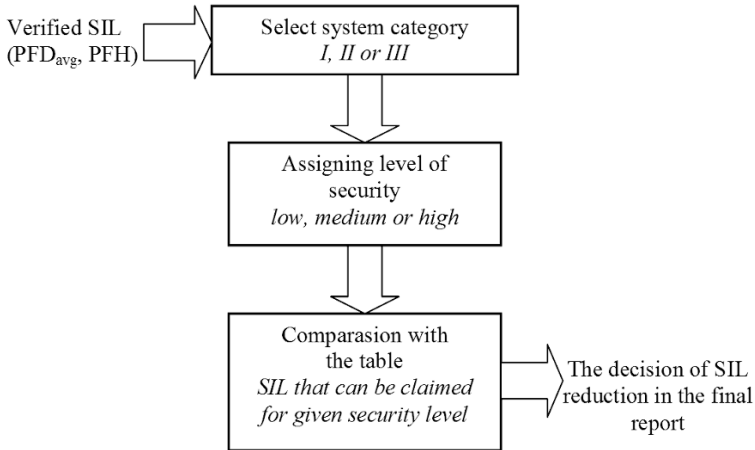


Fig. 5. Procedure of the safety integrity level verification including the security aspects

The security measures which may be taken into account during the functional safety analyses are also of a prime importance. In this article only some of them have been presented. A well-known concept of EAL, SAL and SeSa is the basis for presented methodology. But there are also limitations of in applying the common criteria [16] and for some solutions of programmable systems the EAL related measures may be insufficient. Usually EAL is related only to single hardware or software element. That is the reason why other security models or descriptions should be taken into account. One of them may be proposed lately the SAL [13] based approach, indented to describe in an integrated way the system security in relation to functional safety concept [14, 15].

5 Summary

Functional safety, which is a part of overall safety, is aimed at reducing the risk of a hazardous system operating to an acceptable or tolerable level by introducing a set of safety-related functions (SRFs). They are to be implemented by the control and/or protection systems which are usually operating in a computer network using the wire and/or wireless communication technologies. In functional safety analyses these aspects are sometimes neglected. The standard IEC 61511 doesn't indicate directly how to consider the safety of communication channels in the

functional safety analysis. There is no doubt that it is a substantial problem, therefore in a new version of IEC 61511:2015 standard some additional requirements concerning the data communication channels in functional safety solutions are introduced [13, 15, 27]. One of the main objectives of functional safety analysis is determining of required safety integrity level (SIL) for the safety-related functions to be realized by safety-related systems. According to IEC 61508 to each SIL (1–4) the interval probabilistic quantitative criterion is defined. Functional safety analysis procedure usually doesn't include security aspects. But in case of distributed control and protection system it can have a practical significance [6, 3–5]. It may affect the results of determining as well as verifying of SIL, taking into account functional safety analysis.

References

1. Aaro R., Hansen G.K.: Reliability quantification of computer-based safety systems. An introduction to PDS. SINTEF Industrial Management. Report No. STF37 A97434, Trondheim (1997)
2. ANSI/ISA99.00.012007. Security for Industrial Automation and Control Systems. Part 1: Terminology, Concepts, and Models, (2007)
3. Barnert T., Piesik E., Śliwiński M.: Real-time simulator of agricultural biogas plant, *Computers and Electronics in Agriculture* 108, 1-11 (2014)
4. Barnert T., Kosmowski K.T., Śliwiński M.: Security aspects in verification of the safety integrity level of distributed control and protection systems. *Journal of KONBIN*, Air Force Institute of Technology, KONBIN 2008, Wrocaw. Warsaw. 150-176, (2008)
5. Barnert, T., Kosmowski, K.T., Śliwiński M. 2009. A knowledge-based approach for functional safety management. Taylor & Francis Group, European Safety & Reliability Conference, ESREL 2009, Prague. London, (2009)
6. Barnert, T., Śliwiński M. Functional safety and information security in the critical infrastructure objects and systems (in Polish), *Modern communication and data transfer systems for safety and security*. Wolters Kluwer, 476-507 (2013)
7. CSS PNCSD Control Systems Security Program National Cyber Security Division. Configuring Managing Remote Access for Industrial Control Systems. Centre for the Protection of National Infrastructure CPNI, US Homeland Security, (2010)
8. CSS PNCSD Control Systems Security Program National Cyber Security Division. Cyber Security Assessments of Industrial Control Systems. Centre for the Protection of National Infrastructure CPNI, US Homeland Security, (2010)
9. CSS PNCSD Control Systems Security Program National Cyber Security Division. Recommended Practice: Improving Industrial Control Systems Cyber security with Defense-In-Depth Strategies. Centre for the Protection of National Infrastructure CPNI, US Homeland Security, (2010)
10. Piesik E., Śliwiński M., Barnert T.: Determining and verifying the safety integrity level of the safety instrumented systems with the uncertainty and security aspects, *Reliability Engineering & System Safety* 152, 259-272, (2016)
11. Grtan, T.O., Jaatun, M.G., ien, K., Onshus, T. The SeSa Method for Assessing Secure Remote Access to Safety Instrumented Systems (SINTEF A1626). Trondheim, Norway (2007)
12. Hoyland A., Rausand M.: *System Reliability Theory. Models and Statistical Methods*. Second Edition, New York: John Wiley & Sons, Inc. (2004)

13. IEC 62443. Security for industrial automation and control systems. Parts 1-13 (undergoing development). International Electrotechnical Commission, Geneva (2013)
14. IEC 61508. Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems, Parts 1-7. International Electrotechnical Commission, Geneva (2010)
15. IEC 61511, 2015. Functional safety: Safety instrumented systems for the process industry sector. Parts 1-3. International Electrotechnical Commission (IEC) (2015)
16. ISO/IEC 15408:1999: Information technology Security techniques Evaluation criteria for IT security Part 13 (1999)
17. Kosmowski, K.T. Functional safety and reliability analysis methodology for hazardous industrial plants. Gdańsk University of Technology Publishers (2013)
18. Kosmowski K.T., Śliwiński, Barnert T.: Functional safety and security assessment of the control and protection systems. Taylor & Francis Group, European Safety & Reliability Conference, ESREL 2006, Estoril. London (2006)
19. Kosmowski K.T., Barnert T., Śliwiński M., Porzeziński, M.: Functional Safety Assessment within the Risk Informed Decision Making Process. Proceedings of Joint American and European Conference PSAM 11 / ESREL 2012. Helsinki (2012)
20. Mahan, R.E. (et al.). Secure Data Transfer Guidance for Industrial Control and SCADA Systems. PNNL—20776, Pacific Northwest National Laboratory, Richland (2011)
21. OECD IFP: Project on Future Global Shocks. Reducing Systemic Cybersecurity Risk. IFP/ WKP/ FGS (2011)
22. OECD PCI: Protection of Critical Infrastructure and the Role of Investment Policies Relating to National Security. Paris: Organisation for Economic Co-operation and Development (2008)
23. Piwowar J., Chatelet E., Laclemece P. : An Efficient Process to Reduce Infrastructure Vulnerabilities Facing Malevolence. Reliability Engineering & System Safety 94 (11): 18691877, (2009)
24. Porzeziński M., Redlarski G., Śliwiński M.: Industrial computer networks functional safety. In: Functional safety management in critical systems, 271288. Gdańsk: Fundacja Rozwoju Uniwersytetu Gdańskiego, (2007)
25. Tixier J., Dusserre G., Salvi O., Gaston D.: Review of 62 risk analysis methodologies of industrial plants. Journal of Loss Prevention in the Process Industries. Vol. 15. Elsevier, (2006)
26. Śliwiński, M., Kosmowski, K.T., Piesik, E. Verification of the safety integrity levels with regard of information security issues (in Polish), In: Advanced Systems for Automation and Diagnostics, PWNT, Gdańsk (2015)
27. US-CERT: Control Systems - Overview of Cyber Vulnerabilities. *http* : *//www.us - cert.gov/control_systems/csvuls.html*, Access: (2015)

A data granulation model for searching knowledge about diagnosed objects

Anna Bryniarska¹

Opole University of Technology, Institute of Computer Science,
ul. Proszkowska 76, 45-758 Opole, Poland

Abstract. Diagnostic knowledge about chosen technical classes of objects can be effectively gained by analyzing Internet webpages. In this paper for analyzing these data is proposed the data granulation method. Information granules are mathematical models describing data aggregates. Data aggregates are connected with each other and described by the Fuzzy Description Logic. It is presented that this data granulation model can be used to sharpen the diagnostic knowledge.

1 Introduction

A data (information) granulation is an important paradigm of modeling and processing data with uncertainty. Information granules are the main mathematical constructions in the context of granular computing [16]. If the information granules [15] refer to diagnosed technical objects, then they are mathematical models which describe aggregated diagnostic data [2, 3].

Obtaining diagnostic data in the Internet network, whether searching information, is understood as searching, reliable for experts, subsets: $X \cup X \times X$, where X is some set of available addresses of Internet resources, i.e. such subsets which refer to chosen field of diagnostic knowledge. For a specific query about this knowledge, these subsets indicate, reliable for experts, Internet addresses, where can be found searched diagnostic data. For searching this data we use *the Semantic Web* theory.

Searching information in the Semantic Web is to find a copy of data which are:

- an one-argument attribute value i.e. data representing knowledge about some features or types of the object;
- a two-argument attribute value i.e. data representing knowledge about object properties or relations between two objects.

In first case, data are called *concepts*, and in the second one, they are called *roles*. To describe concepts and roles is used *the Fuzzy Description Logic* (fuzzyDL) language [1, 5, 9, 13]. The fuzzyDL language can be expanded to use some formulas of the first order logic. Then, in this language, we can create *a thesaurus* which describes reference concepts and roles. Otherwise, *a ontology* describes concepts and roles which are searched data. If the searched data from ontology

and the experts criteria (based on their knowledge) are compatible with thesaurus data with some compliance degree, then this relation is called *a residuum* [6].

Searching data in the Internet resources (in the set of Internet addresses) is identified with some interpretations, which describe *a similarity degree* of this data to thesaurus data. The similarity degree is a number in the range of $[0, 1]$ and is a measure of membership of search data to the set of data available in the Internet. Moreover, the similarity degree takes into account the semantic structure of data. Concepts and roles from the ontology are interpreted as fuzzy sets described in the space of the knowledge resources addresses in the Web and pairs of these addresses, respectively. In this way is done the fuzzification of the knowledge representation. The defuzzification of knowledge is interpreted as establishing *the residuum* [6]. It means that for a specific query about knowledge, we find reliable for experts the set of addresses in the Web which represents this knowledge. The sets of interpretation which create residuum are further regarded as searching information.

During the search of information, we use the search rule – *the residuum rule*. We use this rule and the fuzzyDL logic in *the information retrieval logic* (IRL) [6].

The process of searching information is a measurement process, in which we measure data availability in the space of Internet addresses. When we use the residuum rule, we also diagnose whether data, about the diagnosed object, are available in the Internet. In this sense, searching of diagnostic information is also a diagnostic process [2, 3]. We can extend this process to search data in the Semantic Web.

A confidence range for this data is an operation, which determines acceptance (confidence) to the resources, in which we can obtain diagnostic data closest in meaning to thesaurus data. In this way searched information are sharpened. Moreover, it is also using some granulation operation of these resources. We can present the procedure of the information granularity in searching information about the diagnosed technical objects:

- use the diagnostic information retrieval logic language [6, 7],
- describe system of diagnostic information granules, in which this language is interpreted,
- establish the confidence range for data by mathematical morphology methods like erosion and dilation operations. It allows to sharpen diagnostic information granules, in such a way that it eliminates the diagnosis errors i.e. interference of the diagnosed object and information which do not effect on the identification of diagnosed states.

2 Concept of granular computing

In the literature [16] granules are described by:

- Granules elements:

- Related by indistinguishability, similarity, proximity or functionality,
 - Semantic interpretation of placing objects in one granule and its relations,
 - Granules are disjoint or uniquely determined by representatives.
- Granules attributes:
- Internal features of the elements set.
 - External features of granules as whole.
 - Context properties of granules existence.
- Granules structures:
- Internal structure of granule – its elements,
 - Joint structure of granules in one level of abstraction (the granules family – internal connected with the relation network),
 - Hierarchical structure of granules from different levels (the granules network).

Many perspectives on the set granulation are presented in [16]. The granulation method is used to sharpen available information in following granulations levels:

- Language describing a fragment of reality - ontology, thesaurus,
- Objectivity - probability theory, Bayesian theory of probability,
- Subjectivity - Dempster-Shafer theory, fuzzy sets theory,
- Communication - information system, rough sets theory.

2.1 Granules system

A *granules system* is defined in the paper [15] and also in papers [11, 12]. We precise that definition of the granules system:

$$GrS = \langle G, O, El, v, cl \rangle \quad (1)$$

where G is a set of granules indicated from a finite set of elementary granules El by operations from a set O . Furthermore, v, cl are functions:

$$v : G \times G \rightarrow [0, 1] \quad cl : G \times G \rightarrow [0, 1] \quad (2)$$

such that for $g, g_1, g_2 \in G, p \in [0, 1]$:

$$v(g, g) = 1 \quad cl(g, g) = 1 \quad (3)$$

$$cl(g_1, g_2) \geq p \Leftrightarrow v(g_1, g_2) \geq p \wedge v(g_2, g_1) \geq p \quad (4)$$

The function v is called a *granules inclusion*, and cl is a *closeness of granules*. The expressions $cl(g_1, g_2) = p$ and $v(g_1, g_2) = p$ we read respectively: granules g_1, g_2 are closed in the p degree and the granule g_1 is included in the granule g_2 in the p degree.

2.2 Morphological processing of granules

Let two granules systems be:

$$GrS_1 = \langle G^1, O^1, El^1, v^1, cl^1 \rangle \quad GrS_2 = \langle G^2, O^2, El^2, v^2, cl^2 \rangle \quad (5)$$

In the set G^1 are granules $\Delta g \in G^1$ interpreted as granules with error. Then the operation $\varepsilon : G^1 \rightarrow G^2$ decrease the degree of the granules inclusion in relation to the system GrS_1 :

$$\forall g_1 v_2(\varepsilon(\Delta g_1), \varepsilon(g_2)) \leq v_1(\Delta g_1, g_2) \quad (6)$$

and increase the closeness degree:

$$\forall g_1 cl_1(\Delta g_1, g_2) \leq cl_2(\varepsilon(\Delta g_1), \varepsilon(g_2)) \quad (7)$$

The operation ε is called the *erosion*.

Whereas, the operation $\delta : G^1 \rightarrow G^2$ decrease the closeness degree of granules in relation to the system GrS_1 :

$$\forall g_2 cl_2(\delta(g_1), \delta(\Delta g_2)) \leq cl_1(g_1, \Delta g_2) \quad (8)$$

and increase the granules inclusion degree:

$$\forall g_2 v_1(g_1, \Delta g_2) \leq v_2(\delta(g_1), \delta(\Delta g_2)) \quad (9)$$

The operation δ is called the *dilation*.

These definitions of erosion and dilation operation are generalization of definition from papers [4, 10, 14].

2.3 Granulation

Let $ReS = \langle U, R, U_0 \rangle$ be any relational system, where U is a non-empty set, which contains relations fields from the set $R, U_0 \subseteq U$. Moreover, let ReS represent some fragment of reality, for example a physical object, technical object, live organism, measurement system, diagnostic system etc. When T is a theory induced by the model ReS , and $\mathbf{Gr} = \langle G, Gr \rangle$ is an interpretation of the theory T in the granules system GrS , then the interpretation \mathbf{Gr} is called a *system granulation* ReS . The interpretation ϕ^{Gr} is called a *granular computing* of properties ϕ in the GrS system (it is a generalization of the granular computing from papers [12, 15]).

3 System of diagnostic granules

To search diagnostic information in the Web, we firstly prepare a *constructional-technical classifier* for the technical object. In this classifier are all diagnosed parts of objects based on their construction and technology.

Let objects O_1, O_2, \dots, O_k be the pattern diagnosed objects which represent typical symptom-state schemes of errors for some technical object. The model $Model_CT(O_1, O_2, \dots, O_k)$ of constructional - technical classifier is a constructional and technical classes hierarchy. This hierarchy is identified with a semantic network in which classes are concepts, and their connections are roles. We below describe a procedure of determination of the thesaurus and ontology for this semantic network [8].

3.1 The procedure of determination of the thesaurus

The procedure of determination of the thesaurus for diagnostic information:

1. The subject classes with constructional patterns of objects are O_1, O_2, \dots, O_n .
2. The subclasses $T_1^i, T_2^i, \dots, T_l^i$ of the i -th class, respond to different technologies.
3. The constructional groups are $K_1^{i,j}, K_2^{i,j}, \dots, K_t^{i,j}$ for the j -th technology.
4. The base parts are $B_1^{i,j,k}, B_2^{i,j,k}, \dots, B_m^{i,j,k}$ for the k -th constructional group; they satisfy conditions: they have uniquely defined symbol or they are defined by dimension range (interval). If these conditions are not satisfied by diagnosed object, then it is defect, damage or failure of the object.
5. Determine conclude relations between classes: $T_s^i \subseteq O_i, K_s^{i,j} \subseteq T_j^i, B_s^{i,j,k} \subseteq K_k^{i,j}$.
6. Determine the modifiers and synonyms for these classes and relations names.
7. Create schemas of expressions of the fuzzyDL language.
8. Distinguish axioms of the fuzzyDL.

The class hierarchy defined in points 1. – 5. is the constructional-technological classifier of objects with patterns: O_1, O_2, \dots, O_k , what is denoted as $Model_CT(O_1, O_2, \dots, O_k)$. Moreover, in points 1. – 8. is defined the thesaurus for objects: O_1, O_2, \dots, O_k , what is denoted as $Thesaurus(O_1, O_2, \dots, O_k)$.

The morphological table for $Model_CT(O_1, O_2, \dots, O_k)$ can be described as a set of all class systems of the form $[O_i, T_j^i, K_k^{i,j}, B_s^{i,j,k}]$.

Furthermore, we can establish the semantic network for model $Model_CT(O_1, O_2, \dots, O_k)$ by defining edges and nodes of this network as follow. A set of nodes:

$$\Omega = \{\omega : \omega = i \wedge O_i \in Class\} \cup \{\omega : \omega = (i, j) \wedge T_j^i \in Class\} \cup \{\omega : \omega = (i, j, k) \wedge K_k^{i,j} \in Class\} \cup \{\omega : \omega = (i, j, k, s) \wedge B_s^{i,j,k} \in Class\} \quad (10)$$

The classes names $O_i, T_j^i, K_k^{i,j}, B_s^{i,j,k}$ are descriptions of the nodes: $i, (i, j), (i, j, k), (i, j, k, s)$.

A set of edges is a set of all pairs of the nodes, which descriptions satisfy one of these relations: $T_s^i \subseteq O_i, K_s^{i,j} \subseteq T_j^i, B_s^{i,j,k} \subseteq K_k^{i,j}$. The names of these relations are descriptions of the edges.

3.2 The procedure of determination of the ontology

An ontology $Diag_CT(x, O_0, O)$ defines similarity of a diagnosed object O to a pattern object $O_0 \in O_1, O_2, \dots, O_k$ in the Internet resource x .

A diagnose of the object O in the semantic network, based on the *Thesaurus* (O_1, O_2, \dots, O_k), is defined as follow. We define, for example in the Web Ontology Language (OWL), the space of the Internet resources X and then we:

1. Establish the class of the diagnosed object O .
2. Establish the pattern object $O_0 \in \{O_1, O_2, \dots, O_k\}$, to which will be compare the object O .
3. Narrow the classifier $Model_CT(O_1, O_2, \dots, O_k)$ only to classes which describe the pattern object O_0 . For this classifier we use notation: $Model(O_0)$.
4. Establish thesaurus classes which describe the object O_0 . These classes are in the thesaurus relations with the diagnosed object class O .
5. Establish thesaurus relations which corresponds to the $Model(O_0)$ classifier.
6. Define and distinguish new constructional-technical classes, which are in the thesaurus relations to the object class O .
7. Make a list of the Internet resources $L(X)$. There is also the resource in which are searched constructional-technical classes, called the diagnostic symptoms. The symptoms are in relations to the thesaurus classes for the classifier $Model(O_0)$. Other thesaurus classes for the classifier $Model(O_0)$ are called the diagnostic states. The diagnostic states indicate some knowledge gaps, significant deviations, failure or damage of the diagnosed object.
8. For every resource $x \in L(X)$, define a set of diagnostic symptoms $Symptom(x, O_0, O)$ and a set of diagnostic states $State(x, O_0, O)$.
9. Establish the constructional-technological classifier which define the similarity between the diagnosed object O and the pattern object $O_0 \in \{O_1, O_2, \dots, O_k\}$ in the Internet resource $x \in L(X)$: $Diag_CT(x, O_0, O)$, as a set of diagnostic symptoms and states with thesaurus relations between them.
10. The ontology defined for $Diag_CT(x, O_0, O)$, is called the ontology of the object O diagnosis, based on the similarity of the object O to the pattern object O_0 in the resource $x \in L(X)$: $DiagOnt(x, O_0, O)$.
11. For any Internet resource $x \in L(X)$ define, by the method given in [8], value of the similarity indicator $\mu(x, O_0, O) = [\omega_0]$ of the diagnosed object O to the pattern one O_0 in the resource $x \in L(X)$, where ω_0 is a node of the semantic network of the classifier $Model(O_0)$.

3.3 Diagnostic information granules

We define a set of fuzzy interpretation of concepts and roles in the semantic network to determine the similarity between diagnosed and pattern object. Then the fuzzified space is a set $L(X)$ of all Internet resources (the addresses of these resources). The instances of the concept names are copies of these names, which are in the resource $x \in L(X)$, and the instances of the concepts are copies of the names designates of these concepts which are in the resource x . The fuzzy

degree of these instances in the nodes ω , described by these instances names, are calculated for the resource x as *the similarity indicator* $[\omega]$. Furthermore, the roles instances are calculated by the weight function.

Retrieval of the diagnostic data in the semantic network (specific in the Semantic Web), in other words searching information, is to search reliable for experts subsets $X \cup X \times X$, where X is some set of available Internet resources addresses. These subsets refer to chosen diagnostic knowledge field. We can by these subsets reliably interpret the ontology expressions. In this purpose, similar to the statistic, we use *a confidence range* V for searching of diagnostic knowledge. The most important thing is that all experts, based on this confidence range V , accept some set of the membership degrees for any diagnostic object (resource) O to the pattern one $O_0 \in \{O_1, O_2, \dots, O_k\}$ in the resource $x \in L(X)$. This set of the membership degrees is defined by a granule i.e. a fuzzy set:

$$A(O_0, O) = \{(x, \mu_A(x)) : \mu_A(x) = \mu(x, O_0, O), \text{ for } x \in X \cup X \times X\} \quad (11)$$

In this way we get a granules system in the fuzzy set algebra. The weight function of the semantic network nodes describes *a conclusion of granules* and the similarity indicator of these nodes describes *a closeness of the granules*.

3.4 Morphological diagnostic processing of the fuzzy confidence range

The similarity indicator of the importance of searching diagnostic information, for two diagnosed objects, is expressed by the formula:

$$c(x, y) = \begin{cases} \frac{1-|x-y|}{x}, & \text{for } x > y \\ \frac{1-|x-y|}{y}, & \text{for } x \leq y \end{cases} \quad (12)$$

where: x – the importance degree of searching diagnostic information about the first object, y – the importance degree of searching diagnostic information about the second object. Of course: $c(x, y) = c(y, x)$. The importance degree of searching information in the Semantic Web, may be for example expressed by experts as a degree of feeling of the diagnosed state: fault, defect, error or abnormal operation of the diagnosed object O . The less is this degree for a specific state, the less is importance of searching this information and the less is this information.

We assume that the pattern objects O_1, O_2, \dots, O_k correspond respectively to the importance of searching diagnostic information: c_1, c_2, \dots, c_k . We consider only the similarity indicators, for the importance degrees x, y , for which $c(x, y) > \alpha_k$ (eg. $\alpha_k = 0,9$), where α_k is value consider by experts as the maximum-minimum value of the similarity degrees of the importance of information. For comparison we consider only such pairs of the pattern objects (O_i, O_j) , for which is satisfied the condition:

$$c(c_i, c_j) > 0,9 \text{ and } c_i \leq c_j \quad (13)$$

We consider function, which transforms granules of the pattern object pairs into granules of such pairs (O_i, O_j) , for which $c(c_i, c_j) > \alpha_k$. This function has properties of *the dilatation* operation of the granules system.

We can have confidence only in such similarity indicator $A(O_i, O_j) = [\omega(O_i, O_j)]$ for objects O_i, O_j , in which is used the weight function $v(\omega)$. Then for all pattern pairs (O_i, O_j) , the similarity indicator $[\omega(O_i, O_j)]$ is slightly different from the similarity indicator for the importance $c(c_i, c_j)$. The experts may consider as reliable for example the criterion:

$$\frac{|[\omega(O_i, O_j)] - c(c_i, c_j)|}{c(c_i, c_j)} < 1 - \alpha_k = 0, 1 \quad (14)$$

This criterion means that the experts will accept difference between the object similarity and importance similarity within 10%.

We consider function, which transforms granules of the pattern object pairs into granules of such pairs (O_i, O_j) , for which the similarity indicator $[\omega(O_i, O_j)]$ is slightly different from the importance indicator $c(c_i, c_j)$. This function has properties of *the erosion* operation of the granules system.

4 Granulation of the searching diagnostic information

Let ordered algebra of the fuzzy sets $\mathbf{F} = \langle F, \{\vee^F, \wedge^F, \neg^F\}, \{0^F, 1^F\}, v^F, cl^F \rangle$ [6] be a *granules system* (1) for the model $Model_CT(O_1, O_2, \dots, O_k)$. This algebra consists of the family of the fuzzy sets F with following operations:

- \wedge^F - intersection operation of the fuzzy sets;
- \vee^F - union operation of the fuzzy sets;
- \neg^F - complement operation of the fuzzy sets;

sets:

- 0^F - a fuzzy set which is a function having only one number value 0 (an empty set);
- 1^F - a fuzzy set which is a function having only one number value 1;

and functions:

- v^F - function called the conclusion of the fuzzy sets;
- cl^F - function called the closeness of the fuzzy sets.

The sum and product operations of the fuzzy sets are defined by the t-norms and s-norms [6].

Let X be a set of all addresses of the knowledge resources (data copies). This knowledge is represented by the Semantic Web. $X \times X$ is a set of all ordered pairs of elements from the set X . The granulation $\mathbf{Gr} = \langle F, Gr \rangle$ for the model $Model_CT(O_1, O_2, \dots, O_k)$ satisfied the following conditions [6, 9, 13]:

F1. For the concept instances t assigns some values $t^{Gr} \in X$, and for pairs of the concept instances (t_1, t_2) assigns pairs $(t_1^{Gr}, t_2^{Gr}) \in X \times X$.

Mostly, the concept instances are identified with the data copies. These data copies are regarded by IT specialist as objects. The space $X \cup X \times X$ is a set of the addresses of the Semantic Web resources and pairs of these addresses, which contains data copies. The data copies are also the specific connections between data, indicated by pairs of addresses in the Semantic Web.

F2. For the concepts names C assigns fuzzy sets $C^{Gr} : X \cup X \times X \rightarrow [0, 1]$, such that for any $x, y \in X$, $C^{Gr}(x, y) = C^{Gr}((x, y)) = C^{Gr}(y)$.

F3. For role names R assigns fuzzy sets $R^{Gr} : X \cup X \times X \rightarrow [0, 1]$, equal 0 for arguments from the set X .

For any $x \in X$, and concept names C, D :

F4. $\top^{Gr}(x) = 1$ - the granulation of the full concept;

F5. $\perp^{Gr}(x) = 0$ - the granulation of the empty concept;

F6. $(\neg C)^{Gr}(x) = (\neg^F C^{Gr})(x)$ - the granulation of the negation of the concept;

F7. $(C \sqcap D)^{Gr}(x) = (C^{Gr} \wedge^F D^{Gr})(x)$ - the granulation of the conjunction of the concept;

F8. $(C \sqcup D)^{Gr}(x) = (C^{Gr} \vee^F D^{Gr})(x)$ - the granulation of the alternative of the concept;

For any concept names C, D :

F9. $(C \sqsubseteq D)^{Gr} = c^F(C^{Gr}, D^{Gr})$ - the granulation of the conclusion of the concept;

F10. $(C = D)^{Gr} = v^F(C^{Gr}, D^{Gr})$ - the granulation of the equality of the concept.

When the granulation Gr satisfies the conditions **F1** – **F10**, then is called a concepts fuzzification. If as a result of fuzzification we get only characteristic function in the space $X \cup X \times X$, as a membership function for all concepts and roles, then such granulation is *precise*.

5 Conclusion

In this paper the granulation procedure which establishes information granules for description of the diagnosed technical objects is presented. These granules are the fuzzy sets representing aggregated data about objects retrieved from the Internet webpages. Data are connected with each other and these connections are described by the Description Fuzzy Logic for searching information [6] (the connection is between the ontology and thesaurus expressions). Granulation of the ontology expressions leads to the ordered algebra of the fuzzy sets. By using dilation and erosion operation we get granules which are most similar in meaning to the thesaurus expressions (which describe the pattern diagnosed objects).

If diagnostic data are written in the information tables, then the granulation procedure will change. Then in this procedure we have to establish the best possible data [7] written in these tables. Algebra of the possible data written in these tables will be, for this granulation procedure, a granule system. It will allow us to check if for available attributes can be establish the general information system in the Pawlak sense.

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): *The Description Logic. Handbook Theory, Implementation and Application*. Cambridge University Press, Cambridge (2003)
2. Baumeister J.: *Agile Development of Diagnostic Knowledge Systems*. infix, Akademische Verlagsgesellschaft Aka GmbH, Berlin (2004)
3. Belard N., Pencol'e Y., Combacau M.: A theory of meta-diagnosis: Reasoning about diagnostic systems. In *Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI11)*, pp. 731-737, (2011)
4. Bloch I.: *Mathematical Morphology*. In: *Handbook of Spatial Logics*, M. Aiello, I. Pratt-Hartmann and J. van Benthem (eds.), pp. 857-944, Springer (2007)
5. Bobillo, F., Straccia, U.: fuzzyDL: An expressive fuzzy description logic reasoner. In: *Proc. IEEE Int. Conference on Fuzzy Systems FUZZ-IEEE 2008 (IEEEWorld Congress on Computational Intelligence)*, pp. 923-930, (2008)
6. Bryniarska, A.: The Paradox of the Fuzzy Disambiguation in the Information Retrieval. (*IJARAI*) *International Journal of Advanced Research in Artificial Intelligence*, pp. 55-58, Volume 2 No 9, (2013)
7. Bryniarska, A.: The Model of Possible Web Data Retrieval. *Proceedings of 2nd IEEE International Conference on Cybernetics CYBCONF 2015*, pp. 348-353, (2015)
8. Bryniarska, A.: An Uncertain Diagnostic System of the Constructional and Technological Preferences. *Proc. Of The 21st International Conference on Methods and Models in Automation and Robotics MMAR 2016*, pp. 256-260, (2016)
9. Fanizzi, N., d'Amato, C., Esposito, F., Lukasiewicz, T.: Representing uncertain concepts in rough description logics via contextual indiscernibility relations. In: Bobillo, F., da Costa, P.C.G., d'Amato, C., et al. (eds.) *Proc. 4th Int. Workshop on Uncertainty Reasoning for the Semantic Web*, (2008)
10. Kosko B.: *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*. Prentice Hall, Englewood Cliffs, N.J. (1992)
11. Pedrycz, W.: Allocation of information granularity in optimization and decision-making models: towards building the foundations of Granular Computing, pp. 137-145, (2014)
12. Pedrycz, W.: *Granular computing: analysis and design of intelligent systems*. Taylor & Francis Group, Abingdon (2013)
13. Simou, N., Mailis, T., Stoilos, G., Stamou, S.: Optimization techniques for fuzzy description logics. In: *Description Logics. Proc. 23rd Int. Workshop on Description Logics (DL 2010)*. CEUR-WS, vol. 573, (2010)
14. Serra J.: *Image Analysis and Mathematical Morphology*. Academic Press (1982)
15. Skowron A., Swiniarski R., Synak P.: Approximation Spaces and Information Granulation. In: J.F. Peters and A. Skowron (Eds.): *Transactions on Rough Sets III*, LNCS 3400, pp. 175-189, Springer-Verlag Berlin Heidelberg (2005)
16. Yao, Y.Y. : The art of granular computing. In: *Proceeding of the International Conference on Rough Sets and Emerging Intelligent Systems Paradigms LNAI 4585*, pp. 101-112, (2007)

Part X
Intelligent systems

Cybernetics 2.0: Modern Challenges and Perspectives

Dmitry A. Novikov

V.A. Trapeznikov Institute of Control Sciences
Russian Academy of Sciences
Moscow, Russia

Moscow Institute of Physics and Technology
Moscow, Russia
Email: novikov@tushino.com

A new development stage of cybernetics (the so-called cybernetics 2.0) - as a science on general regularities of systems organization and control – is discussed. A fruitful combination of organization and control within cybernetics 2.0 would give a substantiated and efficient answer to the primary question of activity systems engineering: how should control systems for them be constructed? Actually, this is a “reflexive” question related to second-order and even higher-order cybernetics. Mankind has to learn to design and implement control systems for complex systems (high-technology manufacturing, product life cycle, organizations, regions, etc.), similarly to the existing achievements in technical systems engineering!

1. Cybernetics of N. Wiener [21]. CYBERNETICS (from the Greek κυβερνητική “governance,” κυβερνώ “to steer, navigate or govern,” κυβερνη “an administrative unit; an object of governance containing people”) is the science of *control* and *information transmission* processes in different systems, whether *machines*, *animals* or *society* (let us call it “*cybernetics 1.0*”).

Cybernetics studies the concepts of control and communication in living organisms, machines and organizations including self-organization. It focuses on how a (digital, mechanical or biological) system processes information, responds to it and changes or being changed for better functioning (including control and communication).

Cybernetics is an *interdisciplinary science*. It originated “at the junction” of mathematics, logic, semiotics, physiology, biology and sociology. Among its inherent features, we mention analysis and revelation of general principles and approaches in scientific cognition. Control theo-

ry, communication theory, operations research and others represent most weighty theories within cybernetics 1.0.

An alternative is the definition of Cybernetics (with capital C, to distinguish it from cybernetics whenever confuse may occur) as “THE SCIENCE OF *GENERAL REGULARITIES OF CONTROL AND DATA PROCESSING IN ANIMALS, MACHINES AND SOCIETY.*” The second definition differs from its first counterpart in the words “general regularities”. In the former case, the matter concerns “the umbrella brand,” i.e., the “integrated” results of all sciences dealing with problems of control and data processing in animals, machines and society. The latter case covers partial “intersection” of these results, i.e., usage of common results for all component sciences.

In its components, cybernetics intersects considerably with many other sciences, in the first place, with such metasciences as general systems theory and systems analysis and *informatics*.

Alongside with general cybernetics, there exist *special* (“sectoral”) *types of cybernetics*. A most natural approach (which follows from Wiener’s extended definition) is to separate out *technical* cybernetics, *biological* cybernetics and *socioeconomic* cybernetics besides *theoretical cybernetics* (i.e., Cybernetics). It is possible to compile a more complete list of special types of cybernetics (see references in [15]): *physical cybernetics* (to be more precise, “*cybernetical physics*”), *social cybernetics*, *educational cybernetics*, *quantum cybernetics* (quantum systems control, quantum computing), etc.

2. Cybernetics of Cybernetics and other Types of Cybernetics. In addition to Wiener’s classical cybernetics, the last 50+ years yielded other types of cybernetics declaring their connection with the former and endeavoring to develop it further.

No doubt, the most striking phenomenon was the appearance of *second-order cybernetics* (cybernetics of cybernetics, metacybernetics, new cybernetics; here “order” corresponds to “reflexion rank”). Cybernetics of cybernetic systems is associated with the names of M. Mead, G. Bateson and H. Foerster and puts its emphasis on the role of subject/observer performing control [1, 4, 10]. The central concept of second-order cybernetics is an *observer* as a subject refining the subject from the object (indeed, any system is a “model” generated from reality for a certain cognitive purpose and from some point of view/abstraction).

In contrast to Wiener's cybernetics, second-order cybernetics possesses *the conceptual-philosophical character* (for a mathematician or engineer, it is demonstrative that all publications on second-order cybernetics contain no formal models, algorithms, etc.). In fact, this type of cybernetics "transmits" the complementarity principle (with insufficient grounds) from physics to all other sciences, phenomena and processes.

The "biological" stage in second-order cybernetics is associated with the names of H. Maturana and F. Varela [8, 9, 20] and their notion of *autopoiesis* (self-generation and self-development of systems). F. Varela underlined that "first-order cybernetics is the cybernetics of observed systems; second-order cybernetics is the cybernetics of observing systems." The latter focuses on feedback of a controlled system and an observer.

However, the historical picture has appeared much more colorful and diverse, not confining to the second order.

Some authors adopt the terms "*third-order cybernetics*" (social autopoiesis; second-order cybernetics considering autoreflexion) and "*fourth-order cybernetics*" (third-order cybernetics considering observer's system of values), but they are conceptual and still have no generally accepted meanings (e.g., see a discussion in [6, 7, 11, 18, 19] and more references in [15]).

3. Cybernetics 2.0: Definition. The history of cybernetics and its state-of-the-art, as well as the development trends and prospects of several components of cybernetics (mainly, control theory – see also [5]) is briefly considered in [15]. What are the prospects of cybernetics? To answer this question, let us address the primary source—the initial definition of cybernetics as the science of CONTROL and COMMUNICATION.

Its interrelation with control seems more or less clear. At the first glance, this is also the case for communication: by the joint effort of scientists (including N. Wiener), the mathematical theory of communication and information appeared in the 1940's (quantitative models of information and communication channels capacity, coding theory, etc.).

But take a broader view of communication. Both in the paper [17] and in the original book [21], N. Wiener explicitly or implicitly mentioned interrelation or intercommunication or interaction—*reasonability* and *causality* (*cause-effect relations*). Really, in *feedback control sys-*

tems, control-effect is defined by its cause, i.e., the state of a controlled system (plant); conversely, control supplied to the input of a plant is induced by its cause, i.e., the state of a controller, and so on. No doubt, the channels and methods of communication are important but secondary whenever the matter concerns universal regularities for animals, machines and society.

A much broader view of communication implies interpreting communication as INTERCOMMUNICATION, e.g., between elements of a plant, between a controller and a plant, etc. including different types of impacts and interactions (material, informational and other ones). “Intercommunication” is a more general category than “communication.”

In the general systems context, intercommunication corresponds to the category of ORGANIZATION (see its definition and discussion below). Therefore, a simple correction (replacing “communication” with “organization” in Wiener’s definition of cybernetics) yields a more general and modern definition of cybernetics: “the science of systems organization and their control.” We call it *cybernetics 2.0*.

4. Organization and Organization Theory. According to the definition provided by Merriam-Webster dictionary, *an organization* is:

1. The condition or manner of being organized;
2. The act or process of organizing or of being organized;
3. An administrative and functional structure (as a business or a political party); also, the personnel of such a structure.

We’ll use the notion “organization” mostly in its second and first meanings, i.e., as a process and a result of this process. The third meaning (an organizational system) as a class of controlled objects appears in theory of control in *organizational systems* [16].

At descriptive (phenomenological) and explanatory levels [12], “system organization” reflects HOW and WHY EXACTLY SO, respectively, a system is organized (organization as a *property*). At normative level, “system organization” reflects how it MUST be organized (requirements to the *property* of organization) and how it SHOULD be organized (requirements to the *process* of organization).

Note that nowadays also exists “theory of organizations” (“organizational theory”) - a branch of management science, both in its subject (organizational systems) and methods used. Unfortunately, numerous textbooks (and just a few monographs!) give only descriptive generaliza-

tions on the property and process of organization in their Introductions, with most attention then switched to organizational systems, viz., management of organizations (for instance, see the classical textbook [2]).

A scientific branch responsible for the posed questions (Organization theory, or O3 (organization as a property, process and system, by analogy to C3 – Control, Computation, Communication [5, 15]) has almost not been developed to-date. Yet, this branch, originally founded by A. Bogdanov [2] as «The General Organizational Science», obviously has a close connection and partial intersection with general systems theory and systems analysis (mostly focused on descriptive level problems and a little bit dealing with normative level ones), as well as with methodology (as the general science of activity organization [12, 14]). Creating a full-fledged Organization theory is a topical problem of cybernetics!

Following the complication of systems created by mankind, the process and property of organization will attract more and more attention. Indeed, control of standard objects (e.g., controller design for technical and/or production systems) gradually becomes a handicraft rather than a science; modern challenges highlight standardization of activity organization technologies, creation of new activity technologies, etc. (*activity systems engineering*).

5. Cybernetics 2.0: Structure. We have defined cybernetics 2.0 as the science of (general regularities in) systems organization and their control.

A close connection between cybernetics and general systems theory and systems analysis [15], as well as the growing role of technologies leads to a worthy hypothesis. Cybernetics 2.0 includes *cybernetics* (Wiener’s cybernetics and higher-order cybernetics), *Cybernetics*, and *general systems theory* and *systems analysis* with results in the following forms:

- general laws, regularities and principles studied within metasciences—*Cybernetics* and *Systems analysis*;
- a set of results obtained by sciences-components (“umbrella brands”—*cybernetics* and *systems studies* uniting appropriate sciences);
- design principles of corresponding technologies.

Keywords for cybernetics 2.0 are *control*, *organization* and *system*.

Similarly to cybernetics in its common sense, cybernetics 2.0 has a *conceptual core* (Cybernetics 2.0 with capital C). At conceptual level, Cybernetics 2.0 is composed of control philosophy (including general

laws, regularities and principles of control), control methodology, Organization theory (including general laws, regularities and principles of (a) complex systems functioning and (b) development and choice of general technologies).

Basic sciences for cybernetics 2.0 are control theory, general systems theory and systems analysis, as well as systems engineering—see Fig. 1. *Complementary sciences* for cybernetics 2.0 are informatics, optimization, operations research and artificial intelligence—see Fig. 1. The *general architecture of cybernetics 2.0* (see Fig. 1) admits projection to different application domains and branches of subject-oriented sciences depending on a class of posed problems (technical, biological, social, etc.).

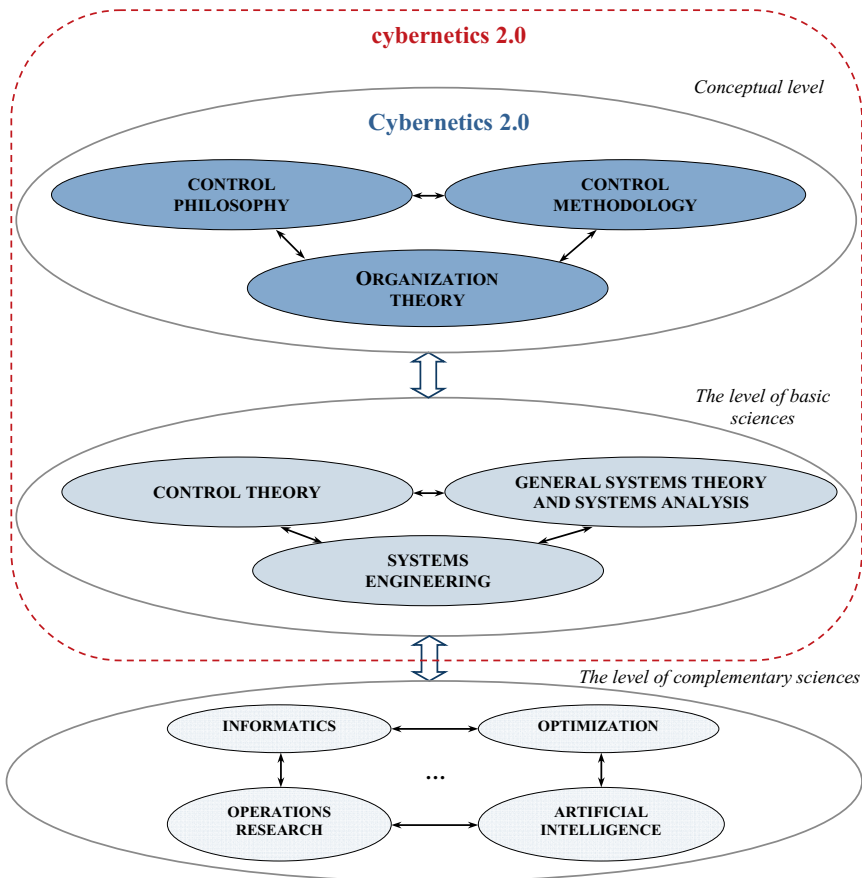


Fig. 1. The composition and structure of cybernetics 2.0

6. The Prospects of Cybernetics 2.0. Further development of cybernetics has several alternative scenarios as follows:

- *the negativistic scenario* (the prevailing opinion is that “cybernetics does not exist” and it gradually falls into oblivion);

- *the “umbrella” scenario* (owing to past endeavors, cybernetics is considered as a “mechanistic” (non-emergent) union, and its further development is forecasted using the aggregate of trends displayed by the basic and complementary sciences under the “umbrella brand” of cybernetics);

- *the “philosophical” scenario* (the framework of new results in cybernetics 2.0 includes conceptual considerations only—the development of conceptual level);

- *the subject-oriented (sectoral) scenario* (the basic results of cybernetics are obtained at the junction of sectoral applications);

- *the constructive-optimistic (desired) scenario* (the balanced development of the basic, complementary and “conceptual” sciences is the case, accompanied by the *convergence and interdisciplinary translation of their common results*, with subsequent generation of conceptual level generalizations (realization of Wiener’s dream “*to understand the region as a whole*”).

The development of cybernetics 2.0 in the conditions of intensified sciences differentiation provides the following:

- for scientists specialized in cybernetics proper and the representatives of adjacent sciences: the general picture of a wide subject domain (and a common language of its description), the positioning of their results and promotion in new theoretical and applied fields;

- for potential users of applied results (authorities, business structures): (1) confidence in the uniform positions of researchers; (2) more efficient solution of control problems for different objects based on new fundamental results and associated applied results.

Main challenges are control in social and living systems. Several classes of *control* problems seem topical, namely:

- network-centric systems (including military applications, networked and cloud production);

- informational control and cybersafety;

- life cycle control of complex organization-technical systems;

- activity systems engineering.

Among promising *application domains*, we mention living systems, social systems, microsystems, energetics and transport.

There exists a series of global *challenges* to cybernetics 2.0 (i.e., observed phenomena going beyond cybernetics 1.0), see [15]:

1) the scientific Tower of Babel (interdisciplinarity, differentiation of sciences; in the first place, in the context of cybernetics–sciences of control and adjacent sciences);

2) centralization collapse (decentralization and networkism, including systems of systems, distributed optimization, emergent intelligence, multi-agent systems, and so on);

3) strategic behavior (in all manifestations, including interests inconsistency, goal-setting, reflexion and so on);

4) complexity damnation (including all aspects of complexity and nonlinearity (Figuratively, in this sense cybernetics 2.0 has to include nonlinear automatic control theory studying nonlinear decentralized objects with nonlinear observers, etc.) of modern systems, as well as dimensionality damnation–big data and big control [13]).

Thus, the main *tasks of cybernetics 2.0* are developing the basic and complementary sciences, responding to the stated global challenges, as well as advancing in appropriate application domains.

And here are the main *Tasks of Cybernetics 2.0*:

1) ensuring the Interdisciplinarity of investigations (with respect to the basic and complementary sciences, as illustrated by Fig. 1);

2) revealing, systematizing and analyzing the general laws, regularities and principles of control for different-nature systems within control philosophy; this would require new and new generalizations;

3) elaborating and refining Organization theory (O^3).

We have described the phylogenesis of a new stage of cybernetics–cybernetics 2.0 and some modern challenges. Further development of cybernetics would call for considerable joint effort of mathematicians, philosophers, experts in control theory, systems engineering and many others involved.

References

- 1 Bateson G. Steps to an Ecology of Mind. – San Francisco: Chandler Pub. Co., 1972. – 542 p.
- 2 Bogdanov A. The General Organizational Science. – Moscow: Ekonomika, 1913-17. Vol. 1-2., 1925-29. Vol. 3. (in Russian) / Bogdanov A. Allgemeine Organisationslehre (Tektologie). – Berlin: Hirzel, 1926. I; 1928. II / Bogdanov A. Essays in Tektology. – Seaside: Intersystems Publications, 1980. – 291 p.

- 3 Daft R. *Organization Theory and Design*. 11th ed. – New York: Cengage Learning, 2012. – 688 p.
- 4 Foerster H. *The Cybernetics of Cybernetics*. 2nd edition. Minneapolis: Future Systems, 1995. – 228 p.
- 5 Forrest J., Novikov D. *Modern Trends in Control Theory: Networks, Hierarchies and Interdisciplinarity // Advances in Systems Science and Application*. 2012. Vol.12. No. 3. P. 1–13.
- 6 Mancilla R. *Introduction to Sociocybernetics (Part 1): Third-Order Cybernetics and a Basic Framework for Society // Journal of Sociocybernetics*. 2011. Vol. 42. No. 9. P. 35–56.
- 7 Mancilla R. *Introduction to Sociocybernetics (Part 3): Fourth-Order Cybernetics // Journal of Sociocybernetics*. 2013. Vol. 44. No. 11. P. 47–73.
- 8 Maturana H., Varela F. *Autopoiesis and Cognition*. – Dordrecht: D. Reidel Publishing Company, 1980. – 143 p.
- 9 Maturana H., Varela F. *The Tree of Knowledge*. – Boston: Shambhala Publications, 1987. – 231 p.
- 10 Mead M. *The Cybernetics of Cybernetics / Purposive Systems*. Ed. by H. von Foerster et al. – New York: Spartan Books, 1968. P. 1–11.
- 11 Müller K. *The New Science of Cybernetics: A Primer // Journal of Systemics, Cybernetics and Informatics*. 2013. Vol. 11. No. 9. P. 32–46.
- 12 Novikov A., Novikov D. *Research Methodology: From Philosophy of Science to Research Design*. – Amsterdam, CRC Press, 2013. – 130 p.
- 13 Novikov D. *Big Data and Big Control // Advances in Systems Studies and Applications*. 2015. Vol. 15. No. 1. P. 21–36.
- 14 Novikov D. *Control Methodology*. – New York: Nova Science Publishers, 2013. – 76 p.
- 15 Novikov D. *Cybernetics: from Past to Future*. – Berlin, Springer, 2016. – 107 p.
- 16 Novikov D. *Theory of Control in Organizations*. – New York: Nova Science Publishers, 2013. – 341 p.
- 17 Rosenblueth A., Wiener N., Bigelow J. *Behavior, Purpose and Teleology // Philosophy of Science*. 1943. No. 10. P. 18–24.
- 18 Umpleby S. *A Brief History of Cybernetics in the United States // Austrian Journal of Contemporary History*. 2008. Vol. 19. No. 4. P. 28–40.

- 19 Umpleby S. The Science of Cybernetics and the Cybernetics of Science // *Cybernetics and Systems*. 1990. Vol. 21. No. 1. P. 109–121.
- 20 Varela F. A Calculus for Self-reference // *International Journal of General Systems*. 1975. Vol. 2. P. 5–24.
- 21 Wiener N. *Cybernetics: or the Control and Communication in the Animal and the Machine*. – Cambridge: The Technology, 1948. – 194 p.

Blocks for the flow shop scheduling problem with uncertain parameters

Wojciech Bożejko¹, Łukasz Gniewkowski¹, and Mieczysław Wodecki²

¹ Department of Automatics, Mechatronics and Control Systems,
Faculty of Electronics, Wrocław University of Science and Technology,
Janiszewskiego 11-17, 50-372 Wrocław, Poland

{wojciech.bozejko, lukasz.gniewkowski}@pwr.edu.pl

² Institute of Computer Science, University of Wrocław
Joliot-Curie 15, 50-383 Wrocław, Poland
mwd@ii.uni.wroc.pl

Abstract. We consider a fuzzy variant of the permutational flow shop scheduling problem in which uncertain values of operations durations are represented by fuzzy numbers. Since even deterministic problem is strongly NP-hard, we propose a tabu search based metaheuristics with the new fuzzy blocks properties for neighborhood searching acceleration.

1 Introduction

Flow shop scheduling problem with makespan goal function, despite the simplicity of its formulation, belongs to the most difficult class of (strongly NP-hard) combinatorial optimization problems. There are many such problems in the literature considered with different parameters of tasks, constrains for machines and jobs and different goal functions – mainly makespan C_{\max} , sum of jobs finishing times C_{sum} and cycle time T . They are important both from theoretical and practical point of view, because they are considered as special, 'benchmark' cases of more general problems or elements of them – there is a number of test instances in the literature, e.g. prepared by Taillard [11].

Garey et al. [7] proved that the problem with makespan criterion C_{\max} is strongly NP-hard for the number of machines $m \geq 3$. Tabu search algorithms were proposed by Nowicki and Smutnicki [10] as well as by Grabowski and Wodecki [8]. Bożejko and Wodecki [3] applied this method in the parallel path-relinking method used to solve the flow shop scheduling problem. Bożejko [2] and Bożejko and Wodecki [4] also proposed parallel algorithms for the problem solving of: determination of the goal function in parallel [2] and scatter search metaheuristics [4], both for the considered problem with deterministic parameters. The problem with uncertain parameters was considered by Bożejko, Hejducki and Wodecki [5] in the genetic algorithm. Bożejko, Rajba and Wodecki [6] used probabilistic approach to model uncertain parameters of the scheduling problem.

2 Problem definition

Let us consider a set of n jobs $\mathcal{J}=\{1,2,\dots,n\}$ and a set of m machines $M = \{1, 2, \dots, m\}$. A job $j \in \mathcal{J}$ is a sequence of m operations $O_{1j}, O_{2j}, \dots, O_{mj}$. The operation O_{ij} should be executed, without interruption, on the machine i in the time p_{ij} . Execution of a job on the machine i (for $i = 2, 3, \dots, m$) can be started after finishing of the execution of this job on the previous machine $i - 1$ (see [1]).

The solution is the schedule of machine work which is represented by vectors of starting times $S = (S_1, S_2, \dots, S_n)$, where $S_j = (S_{1j}, S_{2j}, \dots, S_{mj})$ and finishing times of jobs $C = (C_1, C_2, \dots, C_n)$, where $C_j = (C_{1j}, C_{2j}, \dots, C_{mj})$. In practice, because $C_{ij} = S_{ij} + p_{ij}$, therefore the solution is fully characterized by any of these vectors. For the considered here regular goal function (makespan, C_{\max}) the schedule is maximally moved to the left on the time axis, so we can look for this solution in the set of jobs order on a machine i , and this order is represented be a permutation $\pi_i = (\pi_i(1), \pi_i(2), \dots, \pi_i(n))$ of elements from the set \mathcal{J} .

Permutational flow shop problem. In this paper we consider permutational flow shop problem with makespan criterion. So, let $\pi = (\pi(1), \pi(2), \dots, \pi(n))$ be a permutation of jobs $\{1, 2, \dots, n\}$, and Π a set of all such permutations. Each permutation $\pi \in \Pi$ defines the sequence of execution of jobs on machines (on each machine the same). For the job finishing time C_{ij} the following recurrent equation can be used:

$$C_{i\pi(j)} = \max\{C_{i-1,\pi(j)}, C_{i,\pi(j-1)}\} + p_{i\pi(j)},$$

$$i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n, \tag{1}$$

with an initial constraint

$$C_{i\pi(0)} = 0, \quad i = 1, 2, \dots, m, \quad C_{0\pi(j)} = 0, \quad j = 1, 2, \dots, n.$$

The value of makespan criterion $C_{\max} = C_{m\pi(n)}$ can be determine from the recurrent equation (1), or using one of non-recurrent equations:

$$C_{\max} = \max_{1 \leq j_1 \leq j_2 \leq \dots \leq j_{m-1} \leq n} \left[\sum_{j=1}^{j_1} p_{1\pi(j)} + \sum_{j=j_1}^{j_2} p_{2\pi(j)} + \dots + \sum_{j=j_{m-1}}^n p_{m\pi(j)} \right] \tag{2}$$

and symmetric

$$C_{\max} = \max_{1 \leq i_1 \leq i_2 \leq \dots \leq i_{n-1} \leq m} \left[\sum_{i=1}^{i_1} p_{i\pi(1)} + \sum_{i=i_1}^{i_2} p_{i\pi(2)} + \dots + \sum_{i=i_{n-1}}^m p_{i\pi(m)} \right]. \tag{3}$$

As we can see, for the C_{\max} determination we need two operators: of sum and maximum. It will be important during fuzzy number consideration in the further part of the paper.

Graph model. Values of C_{ij} from equations (1), (2) and (3) can be also determined by using graph model of the problem. For the given sequence of jobs execution $\pi \in \Pi$ we create a graph $G(\pi) = (M \times N, F^0 \cup F^*)$, where $M = \{1, 2, \dots, m\}$, $N = \{1, 2, \dots, n\}$.

$$F^0 = \bigcup_{s=1}^{m-1} \bigcup_{t=1}^n \{((s, t), (s + 1, t))\} \tag{4}$$

is the set of technological arcs (vertical) and

$$F^* = \bigcup_{s=1}^m \bigcup_{t=1}^{n-1} \{((s, t), (s, t + 1))\} \tag{5}$$

is a set of machine arc (horizontal). Arcs of the graph $G(\pi)$ have no weights, but the weight of each vertex (s, t) is $p_{s, \pi(t)}$. The time C_{ij} of finishing of execution of the job $\pi(j)$, $j = 1, 2, \dots, n$, on the machine i , $i = 1, 2, \dots, m$, corresponds to the length of the longest path from the vertex $(1, 1)$ to (i, j) within the weight of this last vertex. For the permutational flow shop problem, described as $F^* || C_{max}$ in the literature, the value of the goal function $C_{max}(\pi)$ for the fixed π equals to the length of the longest (critical) path in the graph $G(\pi)$.

The critical path is decomposed into subsequences $B = [B^1, B^2, \dots, B^m]$ called *blocks* in π , such that:

1. $B^i = [\pi(a^i), \pi(a^i + 1), \dots, \pi(b^i - 1), \pi(b^i)]$, $a^i \leq b^i$, $i = 1, 2, \dots, m$, where $a^1 = 1$, $b^m = n$ and $\pi(b^i) = \pi(a^{i+1})$, $i = 1, 2, \dots, m - 1$,
2. B^i contains operations processed on the same machine, $i = 1, 2, \dots, m$,
3. two consecutive blocks contain operations processed on different machines.

The block is a maximal subsequence of the critical path, which contains operations processed on the same machine. Operations $\pi(a^i)$ and $\pi(b^i)$ in B^i are called the *first* and *last* ones, respectively.

Theorem 1 (Grabowski [8]). *Let $D(\pi)$ be a graph with blocks B^i , $i = 1, 2, \dots, m$. If graph $D(\omega)$ has been obtained from $D(\pi)$ by an interchange of jobs and if $C_{max}(\omega) < C_{max}(\pi)$, then in $D(\omega)$:*

1. at least one job $j \in B^i$ precedes job $\pi(a^i)$, for some $i = 2, \dots, m$, or
2. at least one operation $j \in B^i$ succeeds job $\pi(b^i)$, for some $i = 1, 2, \dots, m - 1$.

Theorem 1 gives the necessary condition to obtain a permutation ω such that $C_{max}(\omega) < C_{max}(\pi)$. In the next part of the paper we will show, that it is possible to consider *fuzzy blocks* which are an extension of the Theorem 1 onto fuzzy processing times.

3 Operations for triangular fuzzy numbers

Definition 1 *Triangular fuzzy number is defined as a fuzzy set $u : \Re \rightarrow I = [0, 1]$ which fulfills the following conditions:*

1. u is continuous,
2. $u(x) = 0$ for $x \notin [a, c]$,
3. it exists a real number b such that $a \leq b \leq c$ and:
 - (a) $u(x)$ is strictly increasing on $[a, b]$,
 - (b) $u(x)$ is strictly decreasing on $[b, c]$,
 - (c) $u(b) = 1$,
 - (d) $u(x) = 0$ on $(-\infty, a) \cup (b, \infty)$.

Triangular fuzzy number A is defined as $A = [A^a, A^b, A^c]$:

The sum of two triangular fuzzy numbers $A = [A^a, A^b, A^c]$ and $B = [B^a, B^b, B^c]$ is defined by:

$$SUM(A, B)(z) = [(A^a + B^a), (A^b + B^b), (A^c + B^c)]. \tag{6}$$

The approximate maximum of two triangular fuzzy numbers $A = [A^a, A^b, A^c]$ and $B = [B^a, B^b, B^c]$ is defined by:

$$\widetilde{MAX}(A, B) = [max(A^a, B^a), max(A^b, B^b), max(A^c, B^c)]. \tag{7}$$

Remark 1 Based on (6) values SUM^a , SUM^b , SUM^c of sum of two triangular fuzzy numbers A i B depends only on their respective values in the arguments of operation.

Remark 2 Based on (6) values MAX^a , MAX^b , MAX^c of approximate maximum of two triangular fuzzy numbers A i B depends only on their respective values in the arguments of operation.

4 Blocks in fuzzy flow shop

According to Remarks 1, 2 and equation (2) it's possible to notice that C_{max}^a , C_{max}^b i C_{max}^c are independent from each other. In result, critical paths can be different. This paths will be denoted by CP^a , CP^b and CP^c .

Theorem 2 Changing order of operations within a block from the path CP^a does not improve the value of C_{max}^a .

Proof. According to (2) value C_{max-k}^a operation executed on machine k is defined by:

$$C_{max-k}^a(\pi) = p_{k\pi(u_k^\pi)}^a + p_{k\pi(u_{k+1}^\pi)}^a + \dots + p_{k\pi(u_{k+1}^\pi)}^a. \tag{8}$$

Let π^* be a permutation obtained by changing order on positions $u_k + i$ and $u_k + j$ where $0 < i < j < u_{k+1} - u_k$. Based on sum commutative, true is:

$$\begin{aligned} & p_{k\pi(u_k^\pi)}^a + \dots + p_{k\pi(u_k^\pi+i)}^a + \dots + p_{k\pi(u_k^\pi+j)}^a + \dots + p_{k\pi(u_{k+1}^\pi)}^a = \\ & p_{k\pi(u_k^\pi)}^a + \dots + p_{k\pi(u_k^\pi+j)}^a + \dots + p_{k\pi(u_k^\pi+i)}^a + \dots + p_{k\pi(u_{k+1}^\pi)}^a = \\ & p_{k\pi^*(u_k^{\pi^*})}^a + \dots + p_{k\pi^*(u_k^{\pi^*}+i)}^a + \dots + p_{k\pi^*(u_k^{\pi^*}+j)}^a + \dots + p_{k\pi^*(u_{k+1}^{\pi^*})}^a. \end{aligned} \tag{9}$$

In result:

$$C_{max,k}^a(\pi^*) = C_{max,k}^a(\pi), \tag{10}$$

seeing that

$$\pi^*(i) = \pi(i) \text{ for } i < u_k \vee i > u_k + 1 \tag{11}$$

than for π^* still exists path: $\langle \tilde{u}_1^\pi, \tilde{u}_2^\pi, \dots, \tilde{u}_{m-1}^\pi \rangle$ which weight is equal to $C_{max}^a(\pi)$. In result true is inequality $C_{max}^a(\pi^*) \geq C_{max}^a(\pi)$ which ends proof.

On the basis of analogical proofs can be shown that:

Theorem 3 *Changing order of operations within a block from path CP^b does not improve the value of C_{max}^b .*

Theorem 4 *Changing order of operations within a block from path CP^c does not improve the value of C_{max}^c .*

5 Tabu search

In general, the method of tabu search is modification of the local search method. In each iteration a neighborhood is generated. The best solution from this neighborhood is taken as starting solution for next iteration. To prevent backwards to last considered solutions, they are stored in so-called tabu list. All solutions from the tabu list are excluded from generated neighborhood. Tabu list also allows to increase the value of the objective function, so as to leave local minimum, at the same time prevents from quick return to previously obtained local minimum.

5.1 Neighborhood.

Based on Theorems 2, 3 and 4 it is possible to determine which movements will not improve solution fitness (called unattractive). In this paper we consider movements which move job from block's inside to outside. According to possibility of differences in CP^a , CP^b i CP^c there is no guarantee that mentioned move is attractive for all paths. To determine moves, which are attractive to all paths we define *common blocks*.

Definition 2 *Common block for k blocks u_1, u_2, \dots, u_k defined as $u_i = \{u_{ib}, u_{ie}\}$ is defined as $com_i = \{com_{iob}, com_{iib}, com_{iie}, com_{ioe}\}$ which fullfill:*

$$com_{iob} < com_{iib} < com_{iie} < com_{ioe}, \tag{12}$$

$$com_{iob} = \min_{i=1..k} \{u_{ib}\}, \tag{13}$$

$$com_{iib} = \max_{i=1..k} \{u_{ib}\}, \tag{14}$$

$$com_{iie} = \min_{i=1..k} \{u_{ie}\}, \tag{15}$$

$$com_{ioe} = \max_{i=1..k} \{u_{ie}\}. \tag{16}$$

Common blocks are determined for all combinations of blocks from all critical paths. For common block com_i every move from $s \in [com_{iib}, com_{iie}]$ to $e \in [0, com_{iob}] \cup [com_{ioe}, n]$ is attractive for all paths. By $TFSB_a$ we donate tabu search algorithm, which moves jobs to at most a position before beginning and after ending of a block.

6 Computational experiments

Test data have been obtained by *fuzzyfication* of Taillard benchmarks [11]. Pseudocode of *fuzzyfication* is shown in Fig. 1. The algorithm is adjustable by four parameters:

- minRange - the minimum width (distance between a and c) in fuzzy number,
- maxRange - the maximum width of fuzzy number,
- minOffset - the minimum distance between a and b in fuzzy number,
- maxOffset - the maximum distance between a and b in fuzzy number.

Parameters minRange and maxRange are expressed as a percentage of deterministic operation duration. Parameters minOffset and maxOffset are expressed as a percentage of range.

```

P[m][n] – Taillard model parameters;
Pfuzzy[m][n] – Fuzzy model parameters;

for i ← 1, m do
  for j ← 1, n do
    range = rand() * (maxRange - minRange) + minRange
    offset = rand() * (maxOffset - minOffset) + minOffset
    Pfuzzy[i][j].a = P[i][j] * (1 - offset * range)
    Pfuzzy[i][j].b = P[i][j]
    Pfuzzy[i][j].c = P[i][j] * (1 + (1 - offset) * range)
  end for
end for

```

Fig. 1: *Fuzzyfication* pseudocode

Experiments were run on a computer equipped with:

- CPU - Intel Core i7 CPU X 980 @ 3.33GHz (6 cores, 12 threads),
- RAM - 24GB,
- OS Linux Ubuntu 12.04.5 LTS, 64-bit.

Algorithm TSFB was compared with tabu search algorithm which determines neighborhood by every possible movement. Comparing stop criterion was time of running. Running time increases with the increase of size of the problem. We consider three values of a parameter: 0, 3 and 10. Criterion of solutions stability (see also [9]) for comparison was an average deviation (\bar{D}):

$$= \sum_{i=1}^R \{ (C_{max}(\pi^*) - C_{max}(\pi^h)) * \mu(i) \} / R \quad (17)$$

where R is a number of experiments, π^* is a permutation with lowest C_{max} value for experiments and π_h is permutation given by heuristic. The $\mu(i)$ was minimal membership function from all drawn operation durations.

time	<i>All</i>	<i>TSFB₀</i>	<i>TSFB₃</i>	<i>TSFB₁₀</i>
2	9.05	1.64	1.32	2.25
4	4.14	1.05	0.89	1.14
6	1.97	0.83	0.78	0.81
8	1.74	0.75	0.80	0.85
10	1.79	0.82	0.85	0.85
12	1.58	0.82	0.76	0.68
14	1.64	0.77	0.80	0.81
16	1.34	0.80	0.77	0.88
18	1.29	0.75	0.74	0.79
20	1.37	0.72	0.67	0.71

Table 1: \bar{D} experiment results for problem size 5x20

time	<i>All</i>	<i>TSFB₀</i>	<i>TSFB₃</i>	<i>TSFB₁₀</i>
4	16.75	7.69	5.43	6.63
8	6.80	4.85	3.44	3.30
12	3.89	4.00	2.53	2.44
16	2.97	3.36	2.08	2.31
20	2.46	3.25	1.56	1.83
24	2.28	3.03	1.97	1.98
28	2.26	2.83	1.97	2.08
32	2.04	2.65	1.33	1.72
36	2.10	2.57	1.50	1.96
40	1.78	2.49	1.34	1.73

Table 2: \bar{D} experiment results for problem size 10x20

The obtained results for TSFB show that using fuzzy blocks provides better solutions comparing to the algorithm which search all the neighborhood, especially for bigger problems. From the other hand, when the number of machines is similar to the number of jobs, algorithms can provide worse solution. This is result of shorten blocks of jobs on machines, which results in less number of possible movements. It finally led to being stuck in a local minimum. Results from Table 3 confirms that. TSFB algorithms with bigger a provide better solutions, because there is a wider neighborhood which prevents to stuck in local minimum. From the other hand, for bigger problems like in Table 5, algorithms with smaller a provide better solutions, especially in shorten algorithm's working time.

time	<i>All</i>	<i>TSFB₀</i>	<i>TSFB₃</i>	<i>TSFB₁₀</i>
6	25.11	17.05	14.69	16.64
12	12.27	12.80	10.37	7.79
18	7.13	11.96	7.75	6.84
24	3.77	9.32	7.02	5.55
30	3.91	10.08	6.20	4.57
36	3.64	8.41	5.47	4.69
42	3.41	9.15	4.30	4.05
48	3.37	8.15	5.04	3.91
54	2.14	7.45	4.28	3.53
60	2.49	8.51	4.93	3.29

Table 3: \bar{D} experiment results for problem size 20x20

time	<i>All</i>	<i>TSFB₀</i>	<i>TSFB₃</i>	<i>TSFB₁₀</i>
10	34.24	2.65	7.13	17.57
20	28.26	1.52	1.35	5.91
30	22.16	0.57	0.68	1.63
40	15.76	1.20	0.66	0.64
50	11.64	1.00	0.48	0.58
60	9.45	1.27	0.51	0.71
70	7.97	0.59	0.44	0.64
80	5.60	0.73	0.42	0.60
90	4.56	1.01	0.33	0.64
100	3.32	0.36	0.42	0.38

Table 4: \bar{D} experiment results for problem size 5x50

time	<i>All</i>	<i>TSFB₀</i>	<i>TSFB₃</i>	<i>TSFB₁₀</i>
20	54.23	13.73	24.95	39.37
40	49.29	5.56	11.97	25.05
60	44.41	4.78	7.08	15.69
80	39.14	3.44	5.47	11.08
100	33.22	4.43	4.19	7.94
120	29.01	2.94	3.81	6.40
140	26.18	2.90	3.37	5.49
160	23.48	2.55	3.04	4.52
180	20.92	3.00	2.59	4.51
200	17.30	2.76	2.42	3.63

Table 5: \bar{D} experiment results for problem size 10x50

7 Conclusions

In the paper we propose the new method of acceleration of the neighborhood determination in local search metaheuristics which operate on uncertain data. Proposed so-called fuzzy block approach is designed to use with uncertain jobs processing times which are represented by fuzzy numbers. Computational ex-

periments show that solutions generated by tabu search algorithm with using proposed methodology are much better than solutions obtained without this mechanism, in the same time of computations.

References

1. Bożejko W., Pempera J., Smutnicki C.: Parallel simulated annealing for the job shop scheduling problem. In: Allen G et al. (Eds.) ICCS 2009, Part I, Lecture Notes in Computer Science No. 5544, Springer (2009), 631–640.
2. Bożejko W.: Solving the flow shop problem by parallel programming. *Journal of Parallel and Distributed Computing* 69 (2009) 470–481.
3. Bożejko W., Wodecki M.: Parallel path-relinking method for the flow shop scheduling problem. *Lecture Notes in Computer Science* No. 5101, Springer (2008), 264–273.
4. Bożejko W., Wodecki M.: Parallel scatter search algorithm for the flow shop sequencing problem. *Lecture Notes in Computer Science* No. 4967, Springer (2008), 180–188.
5. Bożejko W., Hejducki Z., Wodecki M.: Fuzzy blocks in genetic algorithm for the flow shop problem. In *Proceedings of the Conference on Human System Interaction HSI'08*, IEEE Computer Society, 1-4244-1543-8/08/ IEEE (2008).
6. Bożejko W., Rajba P., Wodecki M.: Scheduling problem with uncertain parameters in Just in Time system. *Lecture Notes in Artificial Intelligence* No. 8468, Springer (2014), 456–467.
7. Garey M.R., Johnson D.S., Seti R.: The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research* 1 (1976), 117–129.
8. Grabowski J., Wodecki M.: A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion. *Computers & Operations Research* 31 (2004), 1891–1909.
9. Herroelen W., Leus E.: Project Scheduling under Uncertainty: Survey and Research Potentials. *European Journal of Operational Research* 165(2) (2005), 289–306.
10. Nowicki E., Smutnicki C.: A fast tabu search algorithm for the permutation flow shop problem. *European Journal of Operational Research* 91 (1996), 160–175.
11. Taillard E.: Benchmarks for basic scheduling problems. *European Journal of Operational Research* 64 (1993), 278–285.

Industrial Platform for Rapid Prototyping of Intelligent Diagnostic Systems

Tomasz Żabiński, Tomasz Mączka, and Jacek Kluska

Faculty of Electrical and Computer Engineering, Rzeszów University of Technology,
35-959 Rzeszów, Powstańców Warszawy 12, Poland,
{tomz,tmaczka,jacklu}@prz.edu.pl

Abstract. In this paper the industrial platform for rapid prototyping of intelligent real-time monitoring and diagnostic system was proposed. Its architecture is ready to utilize advanced computational intelligence methods, especially devoted to novelty detection such as autoassociative neural network, local outlier factor, one-class support vector machines, or to solve multiclass classification problems. The rapid prototyping tool set based on Matlab/Simulink and industrial automation equipment was described in details. As an example of the use of the proposed platform, CNC milling tool head mechanical imbalance online prediction system was described.

Keywords: diagnostic systems, computational intelligence, rapid prototyping, mechanical imbalance prediction, Intelligent Manufacturing System, Industry 4.0

1 Introduction

Complicated manufacturing processes include different machining operations and involve many process variables, which complex interactions determine machines performance and components quality. Current challenge is to develop a new production monitoring and diagnostic system structure that exhibit intelligence, robustness and adaptation to environment changes and disturbances [1,2], and simultaneously satisfy industrial requirements and standards. Modern industrial system structure constitutes a hardware and software platform for practical implementation of Intelligent Manufacturing System (IMS) and Industry 4.0 concepts in metal processing industry, e.g. aerospace manufacturing. This concepts require an intensive use of Information and Communication Technologies (ICT) to support reliable management of production processes and utilize Artificial Intelligence (AI) and Computational Intelligence (CI) techniques [3,4] to: monitor, control and diagnose machines and production processes; support a human in manufacturing activities; automatically arrange materials, tools and production compositions; recommend and perform actions to prevent faulty production, performance reduction and machines breakdowns; automatically discover and provide knowledge about manufacturing process, equipment efficiency

and condition; provide knowledge and tools for reliable management decisions; support techniques for production process optimization.

The main role of the intelligent real-time monitoring and diagnostic platform is to provide human operators and maintenance personnel with information, alarms and early warning signals to prevent production of out-of-specification components and to avoid machines breakdowns. The platform should also deliver advanced Human-System Interface (HSI) for efficient interaction between operators and computer systems, which can significantly improve overall production effectiveness [5]. Moreover, the intelligent platform should support maintenance management system and enable practical implementation of Predictive Maintenance (PdM) strategy and Failure Mode Avoidance paradigm to avoid potential failures in high precision machining facilities [6,7]

Due to the complexity of CNC machines, different working conditions in individual factory floors, diversity of machines history and technical condition as well as CI methods specificity, process of intelligent diagnostic system implementation for each particular machine should be treated individually. The measurement signals types and features as well as CI methods, and parameters should be adjusted to the particular machine or technological process. To fulfil this requirement, data used to develop and test intelligent diagnostic methods should be registered on particular machines in their destination location and in typical industrial working conditions. To have the ability to conduct experiments in industrial environment and shorten time of solutions development, the appropriate tool set must be used.

This paper is composed of the following Sections. In Section 2, new architecture concept and implementation details of the intelligent real-time monitoring and diagnostic industrial platform are presented. In Section 3, rapid prototyping tool set for intelligent diagnostic systems development is described. In Section 4, an exemplary application of the tool set for CNC milling tool head mechanical imbalance diagnostic is demonstrated. In Section 5, the conclusions are formulated.

2 Architecture of Industrial Platform for Intelligent Diagnostic Systems

The architecture of the intelligent diagnostic industrial platform proposed in this paper, consists of the three major modules: monitoring and feature extraction (MFE), real-time anomaly detection (RTAD) and fault diagnosis (FD) (Fig. 1). In the MFE module, signal processing (noise reduction, filtering, signal transformations, etc.) and feature extraction methods are used in real-time to receive operating parameters of the machine on the basis of sensors signals acquisition and data acquired from machines control systems. Operators observations are input to the system via HSI which is a part of monitoring module. Different methods for data processing and feature extraction can be used in MFE [14,15] The selection of appropriate sensors and signals features adjusted to the problem

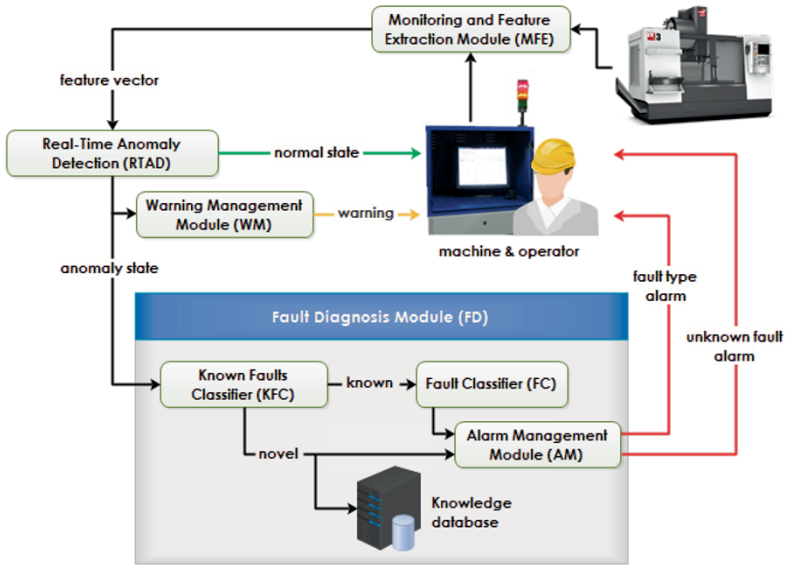


Fig. 1. Architecture of real-time monitoring and diagnostic system.

specificity is the crucial element of monitoring and diagnostic system development and is supported by rapid prototyping platform described in Section 3.

The signals features calculated by MFE are provided to RTAD module which is devoted to detect in real-time any forms of deviations (novelty/anomaly) in normal machine operation. In this module different novelty detection methods can be used, i.e. probabilistic, distance-based, etc. [16]. To develop novelty detection algorithm, only data for normal machine operation is required. When novelty is detected, RTAD sends the warning signal to the operator. In this work like in [16], normal condition is treated as positive example and novelty condition is treated as negative example. This convention is opposite to the one used in medical papers and in work [18], but is more common in the technical diagnostics field. Classifiers used in RTAD module should be characterized by as high as possible value of specificity defined as $Spe = TN / (TN + FP) \cdot 100\%$ parameter and as low as possible false alarm rate value defined as $FAR = FN / (TP + FN) \cdot 100\%$ (TP – true positive, TN – true negative, FP – false positive, FN – false negative).

In the FD module, the anomaly detected by RTAD is examined and the appropriate fault type alarm or unknown fault alarm is provided to the operator. The FD module consists of the two major subsystems, i.e. known faults classifier (KFC) and fault classifier (FC). KFC is used to examine anomaly state and determine if FC module is able to perform its correct classification on the basis of the knowledge gathered by the system. In this module, novelty detection methods can also be used, but in the opposite manner than in RTAD. To develop known fault type detection algorithm only data for known faults (positive exam-

ples) is used. If the particular fault type is known by the system, the appropriate fault type is identified by FC module and the alarm signal is provided to the operator. If the fault type is unknown, then data is stored in the system knowledge database and the unknown fault alarm is sent to the operator. Classifiers used in KFC module should be characterized by as high as possible value of sensitivity (Sen) parameter, defined as $Sen = TP / (TP + FN) \cdot 100\%$. In the FC module, either classical multiclass classifiers [9] or hierarchical structure of one-class classifiers [17,18] can be used.

Additional modules, i.e. warning management (WM) and alarm management (AM) are used to limit the number of faulty warnings and alarms on the basis of information provided by the operator via HSI. If the short-term factor of faulty warnings exceeds configured value, then basic parameters (e.g. threshold) of classifiers used in RTAD can be changed temporarily or permanently. If the global system factor of faulty warnings or alarms exceeds configured value, then the system reconfiguration is needed, e.g. classifiers advanced parameters or structure modification. In nowadays industrial practice, human experts are employed to reconfigure the system in such case.

Main elements of the architecture described above, was implemented on the basis of production process monitoring system described in [8]. The system consists of modern industrial automation equipment and custom-made software modules for data acquisition, monitoring and intelligent diagnosis.

For data acquisition and processing as well as for communication purposes, programmable automation controller (PAC) or industrial personal computer (IPC) equipped with real-time subsystem (e.g. TwinCAT 3) and general operating system (e.g. Windows, Linux) can be used. Dedicated, custom-made software, that works on IPC, performs diverse tasks simultaneously, both in real-time, and in general operating system layer. The real-time software automatically acquires data concerning machine state on the basis of communication with machine control system and by the use of electrical signals provided by additional sensors. The application for general operating system, written in C# language, provides HSI for machine operators as well as performs diagnostic operations (FD module) which are not time critical. The application also communicates with real-time software modules, peripheral devices (e.g. barcode reader) and with the server layer. Ethernet is used for communication between PAC/IPC and the server. Data is stored in PostgreSQL database and web services are used for communication between PAC/IPC and the server. In the real-time layer software, a separate programmable logic controller (PLC) task created using ST language (structured text - norm IEC 61131-3) is used to read data from CNC machine control system and from digital and analog input terminals. Another real-time task created using Matlab/Simulink software and automatic code generation tools (Matlab Coder and Simulink Coder) is used to perform data and signals processing (MFE module) and diagnostic operations (RTAD module) which are time critical. Communication between C# application and PLC real-time module is performed by using ADS (automation device specification) protocol.

3 Rapid Prototyping Tool Set for Intelligent Diagnostic Systems Development

The idea of the rapid prototyping tool set for intelligent diagnostic systems was developed as the extension of the rapid control prototyping (RCP) concept and authors experience in the RCP field [19,20,21]. RCP gives tools for quick and convenient control strategy verification and iterative controller development. RCP involves a controller simulated in real-time (on PC equipped with computer-aided control system design software, e.g. Matlab/Simulink, Scilab/Xcos) coupled with a real plant via hardware input/output devices [19,20,22]. A typical RCP structure can be modified in order to use the same PAC or IPC controllers during experiments and a development process as well as for industrial implementation of the final solution [21]. Nowadays, such scenario can be applied for industrial purposes by the use of commercial TwinCAT 3 platform from Beckhoff integrated with Matlab and Simulink software. As it was mentioned in the introduction, the development process of the intelligent diagnostic system should be performed individually for each particular machine. It can be seen that the development of a control system is analogous to development of intelligent diagnostic system and needs similar approach and tools. On the basis of this observation, rapid prototyping tool set for intelligent diagnostic system was developed. Four main phases of the intelligent diagnostic system development process can be distinguished: (1) collecting data from real object (dedicated experiments or normal operation) and creating data base (real-time); (2) analyzing and processing registered data/signals, choosing and computing signal features (offline); (3) choosing and testing diagnostic algorithm on the basis of collected data base (offline); (4) testing chosen algorithm on real object (real-time).

It is desirable to perform all the operations mentioned above on integrated hardware and software platform. A procedure for intelligent diagnostic system rapid prototyping process is shown in Fig. 2.

The rapid prototyping tool set consists of: (a) slx Simulink framework project for collecting data during real-time dedicated experiments performed on a real object, External Mode of Simulink is used in this case - supports phase 1; (b) PLC program framework and communication software for collecting data of normal object operation, data is directly stored in PostgreSQL database - supports phase 1; (c) set of m-files which use standard Matlab functions as well as custom made functions for iterative realization of phases 2-3; (d) slx Simulink framework project for MFE and RTAD implementation and TwinCAT 3 framework project - supports phase 4.

The block schema of the Simulink framework for phase 5 is shown in Fig. 3. The tool set includes many different custom-made libraries in the form of m-files as well as auxiliary software tools, e.g. conversion of DTREG [25] output code to convention used in Simulink framework, conversion of decision tree code obtained from Matlab to m-function which can be used in slx project, etc. The subsystem created as a final slx project (Fig. 3) can also be tested offline, before real-time tests, by the use of data from experiments.

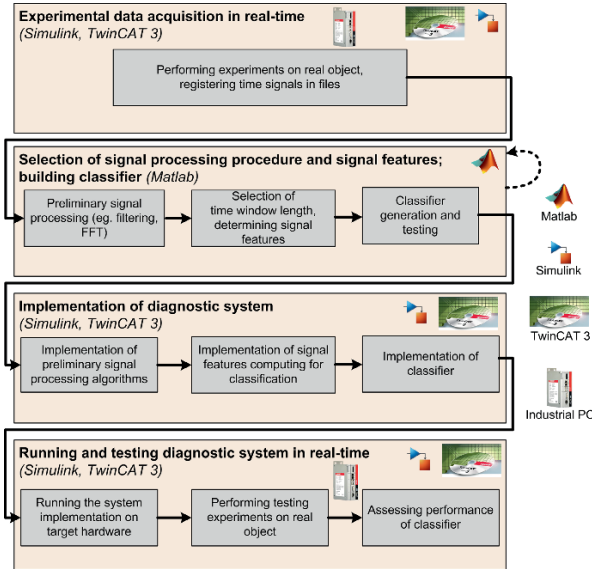


Fig. 2. Procedure for rapid prototyping of intelligent diagnostic system.

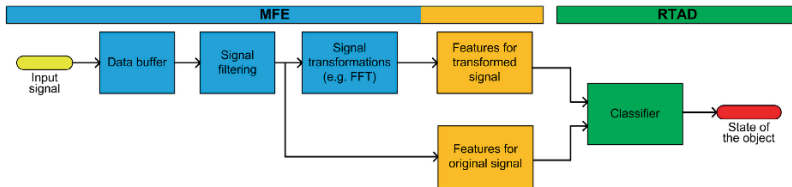


Fig. 3. Implementation of diagnostic system in real-time layer.

4 Application of the Platform for Milling Tool Head Imbalance Prediction

Detection of spindle or tool head mechanical imbalance is an important task in industrial practice. The mechanical imbalance of rotating parts, i.e. spindle, cutting tools (milling, cutters, drills) has significant negative influence on durability of CNC machines and quality of produced parts. Current industrial practice involves periodical imbalance spindle tests performed by maintenance personnel and balancing procedure of cutting tools performed by qualified personnel with the use of special balancing machines as a part of the process setting phase. Online imbalance monitoring of CNC machines rotating parts is a crucial part of PdM paradigm and Industry 4.0 requirements.

The platform described in Sections 2 and 3 was used to develop CNC milling tool head (Fig. 4) online imbalance prediction system in Haas Factory Out-

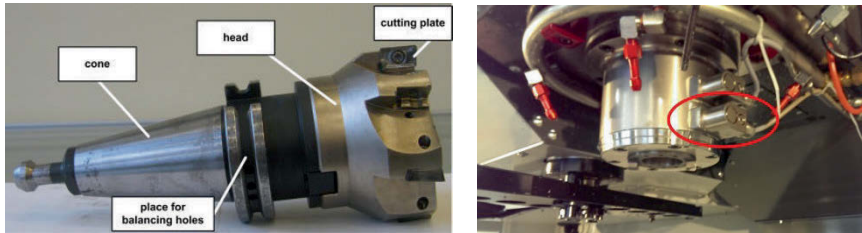


Fig. 4. Milling tool head and spindle with acceleration sensors.

let (HFO) and Haas Technical Education Center (HTEC) located in Rzeszow University of Technology. The industrial testbed consists of Haas VM-3 CNC machine equipped with an inline direct-drive spindle and set of sensors: acceleration and temperature (6 on the spindle: 2 on lower bearing, 2 on upper bearing, 2 on Z axis; 1 on sample), acoustic emission, spindle velocity, spindle load, three axis force and momentum (on sample). The rapid prototyping platform consists of IPC C6920 from Beckhoff, equipped with distributed input-output system (EtherCAT protocol, analog and digital inputs) and software modules (Matlab/Simulink custom-made m-files and slx projects, TwinCAT3 project and custom-made software modules).

The balance quality grade adequate for individual elements (i.e. spindle unit, drawbar components, milling tool) of the spindle-tool system is specified in the ISO 19401:2003 norm [23]. Four imbalance classes were examined in this study, i.e. class 1: G 0.4, class 2: G 2.5, class 3: G 6.3 and class 4: G 40. Preferable balance quality grade for milling tool is G 0.4. The grade G 2.5 is acceptable but not preferable. Grade G 6.3 is not permitted due to deterioration of machining quality. G 40 level is forbidden and may result in damage of the spindle unit. During conducted tests, diverse imbalance classes were obtained by mounting cutting plates of different weight in the milling tool. Precise imbalance value for each milling tool configuration and class as well as for each experiment was evaluated using the Haimer Tool Dynamic 2009 balancing machine.

The rapid prototyping tool set described in Section 3, was applied to perform experiments and collect data in the testbed as well as to develop the main elements (MFE, RTAD, FD) of CNC milling tool head imbalance prediction system. The imbalance test was performed for service speed of the spindle, i.e. 12000 rpm.

The research devoted to select: appropriate sensors, signals features (in time and frequency domain) and computational intelligence methods appropriate for the tool head imbalance prediction was described in [9] and [18]. On the basis of the research results, the acceleration sensor (Hansford HS-100ST) mounted on the spindle lower bearing (Fig 4) was chosen. Different methods for selection of sensors/signals and their features were used, e.g. support vector machine, Sherrod's method [25], principal component analysis, single decision tree. The acceleration signal measurements were divided into the constant length buffers

with duration equal to 640 ms. Sampling interval was 40 μ s. Fourteen acceleration signal features, calculated for each buffer, were examined during the research [9], [18], both in time and frequency domain.

For RTAD module autoassociative neural network (AANN) was selected due to its low computational power demands and availability of Matlab/Simulink tools which enable automatic code generation of the trained network. Data collected for G 0.4 (class 1) was treated as normal state (9625 records) and used to train AANN. Data for grades: G 2.5 (class 2), G 6.3 (class 3) and G 40 (class 4) was treated as anomaly state (17709 records) and used to perform offline classifiers tests. The final AANN structure was 3-4-1-4-3, the threshold value δ was calculated as $\delta = \mu + r \cdot \sigma$ [24], where μ is a mean value and σ - standard deviation. The obtained indicators were: Spe=100% and FAR=0%.

For KFC module in FD subsystem the local outlier factor (LOF) method was selected [18]. Three novelty detection methods were examined, i.e. AANN, LOF and one-class support vector machine (OC-SVM). During the offline experiments, data for grades G 2.5 (class 2), G 6.3 (class 3) and G 40 (class 4) was considered as normal state, i.e. known by FC module. These classes represent imbalance grades which should be recognized by the FC module (Fig. 1). Data for G 0.4 (class 1) was used for testing the classifier's ability to detect unknown state, as data for this class should never be provided to the FD in normal system operation. The LOF classifier achieved the best results, i.e. Sen=98.7%.

For FC module in FD subsystem the classifier based on multilayer perceptron (MLP) was chosen. During the research described in [9] seven methods were examined, i.e. K-Means, probabilistic neural network, single decision tree, boosted decision trees, radial basis function neural network, support vector machine and MLP, to detect four defined above classes. The results were adopted to the 3 class classification problem, i.e. G 2.5, G 6.3, G 40. The indicators: accuracy defined as $Acc=(TP+TN)/(TP+FP+TN+FN) \cdot 100\%$, Sen and Spe were used to assess the performance of the algorithms. The values of all indicators were very high independently of the method. The MLP classifier was chosen due to the same reasons like in the case of AANN in RTAD. The final structure of MLP was: 3-5-3. Three normalized attributes were used and scaled conjugate gradient method was applied for MLP training.

The main elements of CNC milling tool head imbalance prediction system were developed and tested during offline and real-time tests. In the future work the real-time experiments are to be performed for the whole integrated system.

5 Conclusions

The new real-time monitoring and diagnostic industrial platform architecture which utilizes novelty detection CI methods, known also as one-class classifiers, multiclass classifiers, rapid prototyping tool set and industrial automation equipment was described. In contrast to monitoring and diagnostic system structures known from literature [12,13], the proposed platform utilizes only industrial automation equipment, what enables its direct implementation in real production

environments. The main elements of the platform were developed and tested for the prototype CNC milling tool head mechanical imbalance online prediction system. Online imbalance monitoring of CNC machines is a crucial part of PdM and Industry 4.0 paradigms and is particularly important in aviation industry. In the future work, the real-time tests for the complete system are to be performed for long time operation of Haas VM-3 CNC machine. The main elements of the intelligent platform architecture proposed in this paper, has been created in co-operation between Rzeszów University of Technology (Department of Computer and Control Engineering), Żbik company and companies from clusters: Green Forge Innovation Cluster and Aviation Valley located in southeastern Poland region. The system has been used to conduct research in different aspects of IMS practical implementations [4,5], [9,10,11]. The basic hardware and software tools which satisfy industrial requirements and allow intelligent monitoring methods development were defined.

Acknowledgements This research was partially supported by the Grant INNO-TECH-K2/IN2/41/182370/NCBR/13 from the National Centre for Research and Development in Poland and by the Rzeszow University of Technology, Poland, funds for young researchers; No U-733/DS/M.

References

1. Institute for Prospective Technological Studies: „Technical report: The future of manufacturing in Europe 2015-2020 - The challenge for sustainability”. European Commission’s Joint Research Centre (2003).
2. National Research Council: „Visionary manufacturing challenges for 2020”. Committee on visionary manufacturing challenges, board on manufacturing and engineering design, commission on engineering and technical systems, National Academy Press, Washington D.C., www.nap.edu, (1998).
3. Oztemel E.: „Intelligent manufacturing systems”. L. Benyoucef, B. Grabot, (ed) Artificial Intelligence Techniques for Networked Manufacturing Enterprises Management, Springer-Verlag, London, pp 141 (2010).
4. Żabiński T.: „Implementation of Programmable Automation Controllers - Promising Perspective for Intelligent Manufacturing Systems”. Management and Production Engineering Review, Polish Academy of Sciences, Vol. 1 (2), pp 56-63 (2010).
5. Żabiński T., Mączka T.: „Implementation of Human-System Interface for Manufacturing Organizations”. Human-Computer Systems Interaction. Backgrounds and Applications 2, Advances in Soft Computing, Hippe, Z., Kulikowski, J., Mroczek, T. (eds.), pp. 1332 (2011).
6. Henshall E., Campean F.: „Implementing Failure Mode Avoidance”. SAE Technical Paper 2009-01-0990 (2009).
7. Ahmed N., Day J. A., Victory L. J., Zeall L., Young B.: „Condition Monitoring in the Management of Maintenance in a Large Scale Precision CNC Machining Manufacturing Facility”. IEEE Int. Conf. on Condition Monitoring and Diagnosis, September 23-27, pp. 842-845, Bali Indonesia (2012).
8. Mączka T., Żabiński T.: „Platform for Intelligent Manufacturing Systems with elements of knowledge discovery”, Manufacturing System, pp. 183-204, InTech, Croatia (2012).

9. Żabiński T., Mączka T., Kluska J., Kusy M., Gierlak P., Hanus R., Prucnal S., Sep J.: „CNC Milling Tool Head Imbalance Prediction”. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *Artificial Intelligence and Soft Computing: 14th International Conference, ICAISC 2015, Zakopane, Poland, June 14-18, 2015, Proceedings, Part I*, pp 503-514, Zakopane, Poland (2015).
10. Mączka T., Żabiński T., Kluska J.: „Computational Intelligence application in fasteners manufacturing”. *Proceedings of 13th IEEE International Symposium on Computational Intelligence and Informatics (CINTI)*, pp. 335-340, Budapest (2012).
11. Żabiński T., Mączka T., Kluska J., Kusy M., Hajduk Z., Prucnal S.: „Failures Prediction in the Cold Forging Process Using Machine Learning Methods”. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2014, Part I. LNAI vol. 8467*, pp. 622-633, Springer, Heidelberg (2014).
12. Ge M., Xu Y., Du R.: „An Intelligent Online Monitoring and Diagnostic System for Manufacturing Automation”. *IEEE Trans. on Automation Science and Engineering*, vol. 5, No. 1, pp. 127-139 (2007).
13. Uraikul V., Chan W. C., Tontiwachwuthikul P.: „Artificial intelligence for monitoring and supervisory control of process systems”. *Engineering Applications of Artificial Intelligence*, vol. 20, pp. 115-131 (2007).
14. Wang K.: „Intelligent Condition Monitoring and Diagnosis Systems”. IOS Press, ISBN 1-58603-312-3 (2003).
15. Marwala T.: „Condition Monitoring Using Computational Intelligence Methods”. *Applications in Mechanical and Electrical Systems*, Springer, ISBN 978-1-44712379-8 (2012).
16. Pimentel M. A. F., Clifton D. A., Clifton L., Tarassenko L.: „A review of novelty detection”. *Signal Processing* 99, pp. 215-249 (2014).
17. Ge M., Xu Y., Du R.: „An Intelligent Online Monitoring and Diagnostic System for Manufacturing Automation”. *IEEE Trans. on Automation Science and Engineering*, vol. 5, No. 1, pp 127-139 (2007).
18. Mączka T.: „The application of Computational Intelligence and decision support methods in production systems”. PhD thesis, Faculty of Electrical and Computer Engineering, Rzeszów University of Technology, Rzeszów (2016).
19. Skiba G., Żabiński T., Bożek A.: „Rapid Control Prototyping with Scilab/Scicos/RTAI for PC-Based and ARM-based Platforms”. In: *Proc. Of the IMCSIT*, pp. 739-744, Wisla, Poland, (2008).
20. Skiba G., Żabiński T., Wiktorowicz K.: „Rapid prototyping of servo controllers in RTAI-Lab”. *VI Conference on Computer Methods and Systems*, Cracow, pp. 141-146 (2007).
21. Bożek A., Żabiński T., Wiktorowicz K.: „Rapid control prototyping system with industrial embedded PC controller”. *VII Conference on Computer Methods and Systems*, Cracow, pp. 379-384 (2009).
22. Grepl R.: „Real-Time control prototyping in Matlab/Simulink: review of tools for research and education in mechatronics”. *2011 IEEE International Conference on Mechatronics*, April 13-15, Istanbul, Turkey, pp. 881-886 (2011).
23. ISO 19401:2003 Mechanical vibration - Balance quality requirements for rotors in a constant (rigid) state - Part 1: Specification and verification of balance tolerances.
24. Wang G., Cui Y.: „On line tool wear monitoring based on auto associative neural network”. *J. Intell. Manuf.*, 24, pp. 1085-1094 (2013).
25. Sherrod, P.H.: DTREG predictive modelling software, <http://www.dtreg.com>

Evolutionary music composition system with statistically modeled criteria

Zdzisław Kowalczyk¹, Marek Tatar¹, and Adam Bąk¹

Department of Robotics and Decision Systems,
Faculty of Electronics, Telecommunications and Informatics
Gdansk University of Technology,
ul. Narutowicza 11/12, 80-233 Gdańsk, Poland,
kova@pg.gda.pl

Abstract. The paper concerns an original evolutionary music composition system. On the basis of available solutions, we have selected a finite set of music features which appear to have a key impact on the quality of composed musical phrases. Evaluation criteria have been divided into rule-based and statistical sub-sets. Elements of the cost function are modeled using a Gaussian distribution defined by the expected value and variance obtained from an analysis of recognized music pieces. An evolutionary algorithm, considering a reference sequence of chords as an input, is created, implemented and tested. The results of a sampling survey (poll) proves that the melodies generated by the system arouse the interest of a listener.

Keywords: evolutionary optimization, evaluation criteria, music features, chords, music composition, recognized music patterns

1 Introduction

Music is an important element of man's everyday live since prehistoric times. Despite a wide variety of developed music pieces, artists continuously amaze their listeners with new concepts. With so many works developed to date, the question arises of how many unique works can be still invented. Composition process can be seen as a form of a strive for the best solution; however, the main problem lies in the appropriate definition of a fitness function. Assuming that a searched space is finite, a computer system can be used to seek for a solution inside it. If the intuition and experience of a composer is translated into an adequate fitness function, there will be a chance to obtain a new and interesting music compositions.

While developing a fitness function one should keep in mind that composing a music work is a complex process, which can be based not only on know-how and experience, but also on invention and creativity. In the recent years many attempts are undertaken to employ a computer to compose music. Review of the state of the art is provided in [1], where among the utilized methods, genetic algorithms, Markov chains, grammatical methods and artificial neural networks

are enumerated. In this project, we decided to implement genetic algorithms with two types of fitness function.

As mentioned above, the key to successful composing is a proper definition of the fitness function. Among the existing programs, *GenJam* developed by Biles [2] provides a solution, resulting in melodies which are rated by a supervisor, who helps the system to learn. Liu and Ting [3] define 42 rules founded on music theory and calculate the times that the specific rules occur in selected songs. On this basis, and taking into account the listeners' reviews, the authors develop a final fitness function. Another approach presented in [4] is based on the Zipf-Mandelbrot law, where the assumption is made that pleasant music is characterized by certain values of selected features. An alternative approach, emphasizing the importance of the initial population rather than a properly designed fitness function, is reported in [5].

In this paper we describe a project called Music Composer, developed for two kinds of fitness function: statistical and rule-based. A set of statistical features, which are evaluated during the operation of the algorithm, are normalized as in [6]. For the rule-based fitness function, a part of the features is inspired by [3, 7].

2 Numerical Representation of a Music Score

Due to numerous possible combinations in composing music, the issue of its numerical representation arises. Since we consider the problem of evolutionary composition of melody lines based on a given sequence of chords, the numerical representation of melodies must embrace the possibility of representing different rhythmic values. Moreover, the genetic operations used in genetic algorithms (GA) should be adapted to this representation. In order to fulfill the above conditions, the following assumptions for a generated melody are applied:

- rhythmic values of notes are multiples of a sixteenth note
- time signature is fixed and set to $\frac{4}{4}$
- tempo is set by the user
- resulting music line is monophonic.

Due to the constant rhythmic values of particular notes, these assumptions facilitate implementation. Moreover, there is no need to check and verify the sum of rhythmic values inside each bar, because the GA itself does not affect the overall length of a chromosome; thus, the sum of rhythmic values remain unchanged, which is important in performing crossover or mutation.

The structure implemented in our solution is the one described in [3], where a single chromosome consists of a sequence of signed integers, and its length corresponds to the duration of a melody: each bar is represented by 16 values, and each of them corresponds to a rhythmic value of a sixteenth note. The set of possible allele values is shown in Table 1.

An exemplary genotype is presented as a sequence of integers in Table 2 and as a corresponding music score in Fig. 1.

Table 1. Projection of music values on a genotype

Numerical value	Meaning
<0,127>	MIDI note value (from C-1 to G9)
-1	rest (pause)
-2	prolongation of a preceding note or rest

**Fig. 1.** Music score corresponding to the exemplary genotype

2.1 Description of Genetic Algorithm Implementation

Two types of genetic operators are implemented: single point crossover and mutation with musical meaning. The former one is simple mechanism of exchanging equally long parts from the same locus between two chromosomes. The latter one consists of six different types of mutations:

- **interval mutation** - changes the interval between two consecutive notes to another within one octave
- **a single note transposition** - changes the pitch of a selected note by a random interval within one octave
- **prolongation of a note** - changes the rhythmic length of two selected notes or rests (pauses)
- **mutation of rests to notes ratio** - changes a selected note to a rest with the same rhythmic value, or, if a rest is chosen, it is changed to a note with the same rhythmic value
- **rhythmic mutation** - adds a random number of the value "-2" at a selected locus of the chromosome
- **mutation of long-to-short notes ratio** - prolongs a note if classified as short, and divides into two notes if classified as long (a short note means a rhythmic value lower than a quarter note, and a long note means a rhythmic value greater than a quarter note).

The effects of the described mutations are numerically presented in Table 3 and in Fig. 2 as a corresponding music score. The first bar (and the corresponding first row of the table) contains an original track on which mutations are performed. Next six bars (and the rows of the table) present the effects of the introduced mutation operators.

Note that the last three mutations have been added after an analysis of the obtained results. Mutation of the rests-to-notes ratio has been implemented because the resulting melodies consisted only of pitches (rests were not present); thus causing this mutation have resulted in injection of rests to some output melodies. Mutations of the rhythmic and long-to-short notes ratio have been

Table 2. Exemplary genotype composed of three different pitches and a rest

Note	D5	C5	Rest	F5				C5							
Value	74	72	-1	-2	77	-2	-2	-2	74	-2	-2	-2	-2	-2	-2

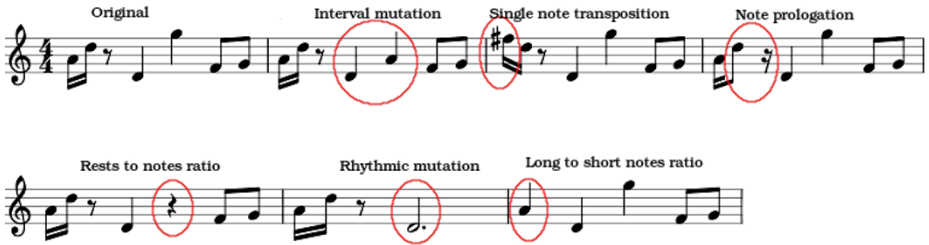


Fig. 2. Music score with the effects of mutation performed on an exemplary melody

Table 3. Influence of the implemented mutations on an exemplary genotype

Original	69	74	-1	-2	62	-2	-2	-2	79	-2	-2	-2	65	-2	67	-2
Interval mutation	69	74	-1	-2	62	-2	-2	-2	69	-2	-2	-2	65	-2	67	-2
Single note transposition	78	74	-1	-2	62	-2	-2	-2	79	-2	-2	-2	65	-2	67	-2
Single note prolongation	69	74	-2	-1	62	-2	-2	-2	79	-2	-2	-2	65	-2	67	-2
Rests to notes ratio	69	74	-1	-2	62	-2	-2	-2	-1	-2	-2	-2	65	-2	67	-2
Rhythmic mutation	69	74	-1	-2	62	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
Long to short notes ratio	69	-2	-2	-2	62	-2	-2	-2	79	-2	-2	-2	65	-2	67	-2

introduced because the resulting melodies were constructed of sixteenth notes only; thus the introduction of such a rhythmic variety appeared to be necessary.

The binary tournament selection is implemented, where each parent is chosen as a fitter one from a pair of randomly picked individuals. Table 4 shows the parameterization of GA used in the conducted experiments. The newly generated population completely replaces the previous population at the end of each epoch (no elitism). If the number of chords is lower than the number of bars, the chord sequence is automatically repeated to fit the number of bars.

Table 4. Settings of the genetic algorithm

Population count	512
Number of bars	12
Mutation probability	0.2
Crossover probability	0.8
Epochs count	500
Scale	G major
Chords progression	G C D C

3 Fitness Function

Many authors take into account a wide range of quality criteria for automatic process of composing music, there is a need for pre-selection of them.

Two approaches to the construction of matching (fitness) functions, based on rules and statistics, are applied here. The former one utilizes the fundamentals of music composition theory, where the technical correctness of a melody is usually measured by checking the level of deviation from predefined musical rules. It requires knowledge in the field of music theory. In the statistical approach, the probability theory is used to describe dependencies between notes in a melody. Frequently, there is an assumption made that a musically pleasant melody is characterized by certain values of selected statistical features [4]. In this approach no theoretical knowledge is required. However, to accurately tune the fitness function, an analysis of recognized compositions, taking into account particular statistics, can be helpful. In our solution, both approaches have been implemented and compared.

Since the process is based on a composer's creativity, the degree of compliance to a specific feature describing a musical work may differ from one musical piece to another. Thus, we propose to model the level of compliance with each feature r_i , being an element of the fitness function, as a Gaussian distribution:

$$f_{r_i}(x) = w_i \cdot \exp\left(\frac{-(r_i(x) - \mu_i)^2}{2\sigma_i^2}\right) \quad (1)$$

where x is the analyzed music composition, w_i is weight of the feature r_i , σ_i is standard deviation of the feature, μ_i is the mean value of this feature, and $r_i(x)$ is a measure of the feature r_i for composition x . Note that (1) takes values from the range $< 0, w_i >$. In order to create a final cost function, the distributions for all features must be summed:

$$f(x) = \sum_{i=1}^N w_i \cdot \exp\left(\frac{-(r_i(x) - \mu_i)^2}{2\sigma_i^2}\right) \quad (2)$$

Note that (2) consists of only statistical features and therefore it is implemented only in the statistic approach. The rule-based fitness function is modeled as a weighted sum of elements, each corresponding to the number of occurrences of a specific feature.

The features included in the fitness function take values between 0 and 1, and the weights w_i are (for now) equal to 1 for all features. Musical meanings of numerical values for particular features are described in the further part of this paper, distinguishing the rule-based and statistical features.

3.1 Rule-based Features

Considering a set of features within the rule-based approach, the inspiration was partially taken from [3, 7]. Musical rules regarding the presence of specific

intervals desired in the outcome melody, and regarding the relation between pitch and chord, and between pitch and scale or one-line octave, are adapted. Moreover, in order to maintain the possibility for producing longer notes, rules rewarding the presence of particular notes (in the sense of their rhythmic value) are also introduced. Similarly, to maintain rests, another rule is included. The rules are listed in Table 5.

Table 5. Selected rule-based features, taken into account in the fitness function

#	Feature
1.	Perfect consonances
2.	Imperfect consonances
3.	Dissonances
4.	Minor and major seconds
5.	Intervals smaller than octave
6.	Whole note rhythmic values
7.	Half note rhythmic values
8.	Quarter note rhythmic values
9.	Eighth note rhythmic values
10.	Sixteenth note rhythmic values
11.	Sum of rhythmic values in one-line octave
12.	Sum of rhythmic values in scale
13.	Sum of rhythmic values in current chord
14.	Sum of rests' rhythmic values
15.	Pitches in strong beat

3.2 Statistical Features

The second selected fitness function is based on statistical measures of describing important notes relationships in the composed or produced melody. Again, a set of features in this cost function has been pre-selected and modified on the basis of literature review, as shown in Table 10.

3.3 Tuning of the Fitness Function

Since the score for each statistical feature in GA is modeled using a Gaussian distribution, the mean value and standard deviation for each of them is required to obtain a complete description of the fitness function. These values can be tuned through a time-consuming observation of input-output pairs (where by an input we mean the set of mean values and standard deviations, and by an output - composed melodies), when iteratively changing the parameters. Such approach requires, however, an expertise from a GA designer in the field of music theory, in order to see whether the melodies are technically correct and pleasant for the listener, or not. Another approach is based on the analysis of recognized music

works, where the mean value and standard deviation are calculated in a statistic way. The melodies taken from the following works have been analyzed:

- “Swan theme from Swan Lake” by P.I. Tchaikovsky
- “Canon in D” by J. Pachelbel
- “Prelude from Cello Suite No. 1” by J.S. Bach
- “Allegro from Eine Kleine Nachtmusik” by W.A. Mozart
- “Entr’acte from Carmen” by G. Bizet
- “Danse des petits cygnes from Swan Lake” by P.I. Tchaikovsky
- “Air on the G String from Orchestral Suite No. 3” by J.S. Bach
- “Can Can” by J. Offenbach
- “Humoresque” by A. Dvorak
- “Entertainer” by S. Joplin
- “Ave Maria” by F. Schubert
- “Arioso from Ich steh mit einem Fuß im Grabe” by J.S. Bach.

The parameters resulting from the analysis of these music works, as well as the ones obtained through subjective tuning are gathered in Table 6.

Table 6. Expected values and standard deviation for the set of selected normalized statistical features (‘-’ indicates a ‘no-result’ case, ignored in fitness functions)

Feature	Music work analysis		Subjective tuning	
	Expected value	Standard deviation	Expected value	Standard deviation
Mean pitch	0.564	0.065	0.5	0.3
Pitch deviation	0.053	0.013	0.1	0.2
Off-scale pitches	-	-	0	0.3
Chord pitches	-	-	0.5	0.3
Dissonances	0.029	0.045	0.1	0.3
Minor and major seconds	0.553	0.078	0.5	0.3
Intervals larger than octave	0.007	0.010	0	0.3
Mean rhythmic value	0.282	0.120	0.4	0.3
Rhythm deviation	0.156	0.058	0.2	0.3
Strong beat	0.788	0.218	0.9	0.3
Rests to notes ratio	0.105	0.092	-	-

The rule-based features have been tuned via properly adjusted weights based on a subjective analysis of the input-output pairs, and the resulted weights are listed in Table 7. Note that shorter notes receive a proportionally lower reward for each occurrence due to the fact that shorter notes in a bar results in a larger number of intervals, each rewarded accordingly. Such a tuning procedure has resulted in the presence of each type of notes in the output melodies. Similar relationship holds between the rules concerning the type of intervals (perfect consonances, imperfect consonances, dissonances) and the ones concerning the intervals being in a scale or in a current chord; which (in some cases) has led to rewarding the same interval by two rules.

Table 7. Weights of rule-based features, obtained from subjective tuning

Feature	Weight
Perfect consonances	1
Imperfect consonances	3
Dissonances	3
Minor and major seconds	4
Intervals smaller than octave	2
Whole note rhythmic values	80
Half note rhythmic values	40
Quarter note rhythmic values	20
Eighth note rhythmic values	5
Sixteenth note rhythmic values	0
Sum of rhythmic values in one-line octave	3
Sum of rhythmic values in scale	1
Sum of rhythmic values in current chord	1
Sum of rests' rhythmic values	5
Pitches in strong beat	5

4 Evaluation of the System

For the final evaluation, 9 melodies, generated with different settings, were selected. There were two criteria: fitness function and configuration. Among the fitness functions we distinguished the rule-based, statistic and those based on various music work analysis. The three configurations are presented in Table 8.

Table 8. Musical configurations used to create melodies for the poll

#	Name	Tempo	Scale	Chord progression	Bars count
1.	Slow	64	G major	G C D C	4
2.	Moderate	96	a minor	a d a e	6
3.	Fast	128	E major	E A H E	8

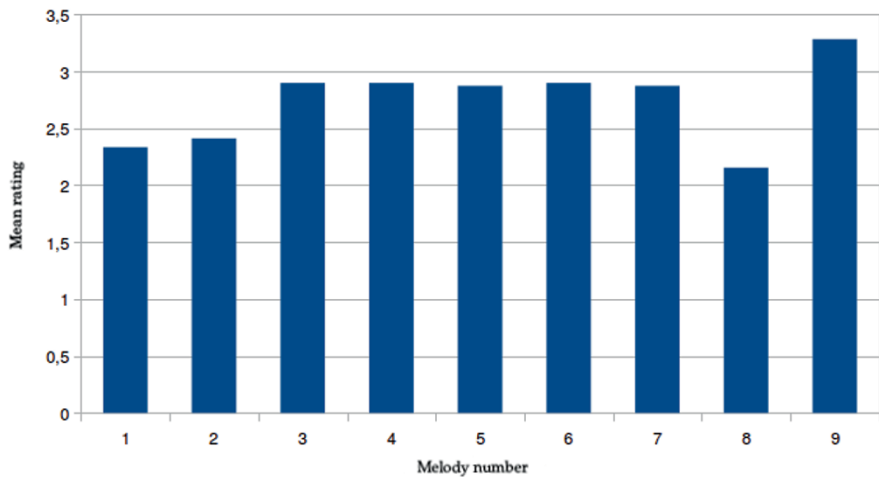
As a result, combining the configurations using the three fitness functions, 9 melodies were obtained.

The poll was conducted by 39 persons via the Internet. Participants, after listening to one of the melodies, were asked to rate it in the scale from 1 (unpleasant) to 5 (pleasant), and to try to determine its 3D psychological estimation (in terms of valence, arousal and dominance, in short VAD) from a certain Self-Assessment Manikin model [8]. Moreover, every participant had the possibility to comment each composition. The order of displaying different compositions to each participant, was fixed randomly. The answers to the question concerning the pleasantness of the analyzed melodies are presented in Fig. 3.

Melody 9 have obtained mainly positive comments, and thus has achieved the best mean rating (equal to 3.28). The music score for this melody is presented

Table 9. Numeration of the melodies rated by poll respondents

Track number	Fitness function variant	Configuration
1.	Statistical based on recognized works	Slow
2.	Subjective statistical	
3.	Subjective rule-based	
4.	Statistical based on recognized works	Moderate
5.	Subjective statistical	
6.	Subjective rule-based	
7.	Statistical based on recognized works	Fast
8.	Subjective statistical	
9.	Subjective rule-based	

**Fig. 3.** Mean rating obtained from poll for the selected melodies

in Fig. 4. Similarly, for the melody with a lowest rating the score is presented in Fig. 5, for which the respondents pleaded the lack of any melody outline. Such comments pointed out important issue, which should be taken into account while developing fitness function - the concept of musical phrases, which are short musical fragments, which are repeated throughout the musical work with some variations. They can be encountered in the recognized music works.

The attempt to measure emotions invoked by the melodies turned out to be beyond the scope of this work. Respondents found it difficult to determine proper parameters of his/her VAD emotional state, which resulted in high variances in a VAD plot. Thus, the results seemed to be more of a random guess than a truly emotionally-based response. Certainly, such test can be suitable for more complex polyphonic compositions.



Fig. 4. Highest rated melody generated using Music Composer



Fig. 5. Lowest rated melody generated using Music Composer

5 Conclusions

In this paper we have described an innovative program for composing melodies using genetic algorithms. The developed solution implements a crossover operator and six musically pertinent mutations. The user chooses between rule-based and statistical fitness functions, and tune their parameters. Three sets of tuned parameters are possible to import: subjective statistics, subjective rule-based settings, and 'objective' statistics based on selected music works. For the purpose of program evaluation, nine melodies have been picked out and rated in a poll, in which respondents rated these melodies and marked their emotions induced by the melodies, taking into account an emotional VAD model (valence, arousal and dominance).

The rating of the melodies varied from 2.15 to 3.28, which indicates that they may arouse the interest of a listener. In case of some melodies, respondents drew attention to the fact that there was a lack of musical connection between consecutive parts. Indeed, the melodies were evaluated in the GA as a whole by considering the applied principles of statistics, bypassing the important relationship between successive bars.

It is thus clear that the absence of such factors in the fitness function, results in omitting the concept of musical phrases - i.e. music parts repeated in original

or modified form in subsequent bars or groups of bars. Therefore, inclusion of a feature considering musical phrases appears to be a milestone, necessary in further development of this project.

Difficulty in evaluating the listener VAD emotion can be caused by the fact that the program is currently focused on composing monophonic melody lines, disregarding various accents or possible harmonies. Moreover, the choice of an adopted instrument, which plays the melody, is also important, and may influence the emotion.

Further development of this project will focus on the introduction of features that will bring musical phrases into the tunes produced by the program. We plan to make the program for the production of sounds (pitches) occurring simultaneously. We also want to go beyond the monophonic texture (which will require new methods for evaluation of the features). Similarly, addition of accents can contribute to a higher quality of the outcome melodies.

References

1. Fernández J. D., Vico F.: AI methods in algorithmic composition: A comprehensive survey. *Journal of Artificial Intelligence Research* 48, 513–582 (2013).
2. Biles J.: GenJam: A genetic algorithm for generating jazz solos. *ICMC Proceedings*, 131–137 (1994).
3. Liu C., Ting C.: Evolutionary composition using music theory and charts. *IEEE Symposium on Computational Intelligence for Creativity and Affective Computing (CICAC)*, 63–70 (2013).
4. Manaris B., Vaughan D., Wagner C., Romero J., Davis R. B.: Evolutionary music and the Zipf-Mandelbrot Law: Developing fitness functions for pleasant music. *Applications of Evolutionary Computing*, 522–534, Springer, Berlin-Heidelberg (2003).
5. Waschka R.: Avoiding the fitness "bottleneck": Using genetic algorithms to compose orchestral music. *ICMC Proceedings*, 201–203 (1999).
6. Towsey M., Brown A., Wright S., Diederich J.: Towards melodic extension using genetic algorithms. *Educational Technology & Society* 4(2) 54–65 (2001).
7. Wiggins G., Papadopoulos G., Phon-Amnuaisuk S., Tuson A.: Evolutionary methods for musical composition. *International Journal of Computing Anticipatory Systems* (1999).
8. Bradley M. M., Lang P. J. Measuring emotion: the Self-Assessment Manikin and the Semantic Differential, *Journal of behavior therapy and experimental psychiatry*, 25(1) pp. 49–59, 1994.

Table 10. Description of implemented statistical features and meaning of its values

Average pitch:		$\frac{\text{Mean pitch MIDI value}}{\text{The highest possible difference in MIDI values (127)}}$
Limiting values	0	Only lowest MIDI pitches are present
	1	Only highest MIDI pitches are present
Pitch deviation:		$\frac{\text{Standard deviation of pitches' MIDI values}}{\text{The highest possible standard deviation}}$
Limiting values	0	All notes have the same pitch
	1	There are highest and lowest possible notes
Off-scale pitches:		$\frac{\text{Number of pitches from a scale}}{\text{Number of pitches out of a scale}}$
Limiting values	0	All notes are from a given scale
	1	All notes are out of a given scale
Comments	Rests are ignored	
Chord pitches:		$\frac{\text{Sum of rhythmic values of pitches from a given chord}}{\text{Sum of all rhythmic values in a melody}}$
Limiting values	0	All pitches are out of a given chord
	1	All pitches are from a given chord
Dissonances:		$\frac{\text{Number of dissonances}}{\text{Number of all intervals}}$
Limiting values	0	No dissonance is present
	1	Every interval is dissonance
Comments	Rests between notes and single notes are ignored. The following intervals are treated as a dissonance: tritone, major seventh, minor seventh, and all intervals larger than one octave.	
Minor and major seconds:		$\frac{\text{Number of second interval occurrence}}{\text{Number of all intervals}}$
Limiting values	0	There is no second interval
	1	Each interval is either major or minor second
Comments	Rests and single notes are ignored; intervals are examined ignoring rests between two notes	
Intervals larger than octave:		$\frac{\text{Number of intervals larger than octave}}{\text{Number of all intervals}}$
Limiting values	0	There are no intervals larger than octave
	1	Every interval is larger than octave
Comments	Rests and single notes are ignored; intervals are examined ignoring rests between two notes	
Mean rhythmic value:		$\frac{\text{Mean value of logarithm of (rhythmic value + 1)}}{\text{Logarithm of (whole note rhythmic value + 1)}}$
Limiting values	0	Impossible, but value close to 0 means that melody is constructed only from sixteenth note
	1	Melody is constructed only from whole notes
Comments	Whole note has rhythmic value 4, sixteenth note 0.25	
Rhythm deviation:		$\frac{\text{Standard deviation of logarithm of (rhythmic value + 1)}}{\text{Logarithm of (whole note rhythmic value + 1)}}$
Limiting values	0	Every rhythmic value is the same
	1	Melody has the longest and the shortest possible notes
Comments		
Rests to notes ratio:		$\frac{\text{Sum of rests rhythmical values}}{\text{Sum of all rhythmic values}}$
Limiting values	0	There are no rests in a melody
	1	Melody is made only of rests

Iterative Learning Control for a class of spatially interconnected systems^{*}

Błażej Cichy¹, Petr Augusta², Krzysztof Gałkowski¹, and Eric Rogers³

¹ Institute of Control and Computation Engineering, University of Zielona Góra,
ul. Szafrana 2, 65-516 Zielona Góra, Poland
{b.cichy, k.galkowski}@issi.uz.zgora.pl

² Institute of Information Theory and Automation, The Czech Academy of Sciences,
Pod Vodárenskou věží 4, CZ-182 08 Prague, Czech Republic
augusta@utia.cas.cz

³ Department of Electronics and Computer Science, University of Southampton,
Southampton SO17 1BJ, UK
etar@ecs.soton.ac.uk

Abstract. An unconditionally stable finite difference discretization motivated by the well-known *Crank–Nicolson* method is used to develop an Iterative Learning Control (ILC) design for systems whose dynamics are described by a fourth-order partial differential equation. In particular, a discrete in time and space model of a deformable rectangular mirror, as an exemplar application, is derived and used in the ILC design. Finally, the feasibility of the new ILC design is confirmed by numerical simulations.

Keywords: Crank–Nicolson Discretization Method, Iterative Learning Control, Rectangular Plate.

1 Introduction

Discretisation of partial differential equations (PDE) describing systems with spatial and temporal dynamics is often required for the design and implementation of control laws. A critical factor in this general approach is numerical stability, i.e., the discrete approximation must produce trajectories as close as possible to those produced by the PDE. One group of methods that can be applied in this case are based on a finite difference approximation [12]. A particular sub-class are those known as explicit methods that were used in [7].

Explicit discretization methods are conditionally numerically stable, i.e., the time discretization period is related to its spatial counterpart and hence the need to use dense time and spatial discretization grids. One way of overcoming this drawback is to use implicit methods, such as *Crank–Nicolson* [8], to obtain an unconditionally stable discrete approximation to the dynamics of the original

^{*} This work is partially supported by National Science Centre in Poland, grant No. 2015/17/B/ST7/03703.

PDE. In this case the time and spatial grids are not related and can therefore be less dense. However, the resulting discrete model is in implicit, or singular, form, i.e., there is no straightforward recurrent dependence between discrete values at successive time instants. Instead, the dependence is between spatial windows defined at the current and previous time instants, see, e.g., [6].

Formulating and solving control problems for singular systems of the class that arises in the subject area of this paper requires the use of a lifting approach, i.e., absorbing the spatial structure of the system into possibly high dimensional vectors, again see [6] for a detailed treatment. In this paper, the *Crank–Nicolson* method is extended to systems described by a PDE defined over time and two spatial variables. As a particular example, a thin flexible rectangular plate is considered, which, e.g., can be used to model the vibrations of a deformable mirror subject to a transverse external force.

A particular feature [3] of the resulting discrete approximation is the presence of the unconditionally numerically stable property and hence a significantly less dense discretization grid can be used with no degradation of the approximate model dynamics. This, in turn, means a much smaller number of sensors and actuators distributed over the plate to be controlled can be deployed, with obvious advantages in terms of control law design and implementation. This accurate discrete model is used in this paper to contribute new results on ILC law design for classes of PDEs based on linear repetitive process stability theory [11].

Repetitive processes make a series of sweeps, termed passes, through a set of dynamics defined over a fixed finite duration known as the pass length. In particular, a pass is completed and then the process is reset to the starting location and the next pass can begin, either immediately after the resetting is complete or after a further period of time has elapsed. On each pass, an output, termed the pass profile, is produced which acts as a forcing function on, and hence contributes to, the dynamics of the next pass profile. Repetitive processes are therefore a particular case of 2D systems where there are two independent directions of the information propagation.

The application area for ILC [2] is systems that execute the same finite duration task over and over again, where each execution is known as a pass, or trial, and the finite duration the pass length. On each pass the error between the supplied reference trajectory and the output, termed the pass profile, can be formed and the problem is the design of a control law to force this error to zero in some suitable norm or, in more practical terms, to within a specified tolerance over the passes. Once a pass is complete, all information produced during its production is available for use in the control law, including temporal information that would be non-causal in the standard case, provided it is generated on a previous pass. Moreover, the controlled dynamics can be written as a repetitive process and therefore the systems theory for these processes can be used in analysis and design of ILC laws. To conform with the repetitive process literature pass instead of trial is the terminology used in this paper.

Since the first work, ILC has remained an active area of control systems theory and applications. The survey papers [5, 1] are one starting point for the

literature. Also there has been a technology transfer to robotic-assisted upper limb stroke rehabilitation, see, e.g., [9]. This paper continues the development of repetitive process based ILC design for systems described by PDEs. The resulting design algorithm can be computed using Linear Matrix Inequalities (LMIs) and is illustrated in simulation by application to the deformable mirror example.

Throughout this paper the null and identity matrices, respectively, with compatible dimensions are denoted by 0 and I . Also $\succ 0$ denotes a symmetric positive definite matrix, $\prec 0$ a symmetric negative definite matrix and the following matrix notation is used

$$\begin{aligned} \text{diag}(II, \Sigma) &= \begin{bmatrix} II & 0 \\ 0 & \Sigma \end{bmatrix}, & \text{odiag}_\eta(II) &= \begin{bmatrix} II & & & & 0 \\ & \ddots & & & \\ & & II & & \\ & & & \ddots & \\ 0 & & & & II \end{bmatrix}_{\eta \times \eta}, \\ \text{tri}_\eta(II, \Sigma) &= \begin{bmatrix} II & \Sigma & & & 0 \\ \Sigma & II & \Sigma & & \\ & \ddots & \ddots & \ddots & \\ & & \Sigma & II & \Sigma \\ 0 & & & \Sigma & II \end{bmatrix}_{\eta \times \eta}, & \text{pent}_\eta(II, \Sigma, \Upsilon) &= \begin{bmatrix} II & \Sigma & \Upsilon & & & & 0 \\ \Sigma & II & \Sigma & \Upsilon & & & \\ \Upsilon & \Sigma & II & \Sigma & \Upsilon & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & \Upsilon & \Sigma & II & \Sigma & \Upsilon \\ & & & \Upsilon & \Sigma & II & \Sigma \\ 0 & & & & \Upsilon & \Sigma & II \end{bmatrix}_{\eta \times \eta}. \end{aligned}$$

2 Partial differential equation representation and discretization

The dynamics of the continuous deformable mirror considered in this paper are modeled by the following Lagrangian PDE

$$\frac{\partial^4 w(x, y, t)}{\partial x^4} + 2 \frac{\partial^4 w(x, y, t)}{\partial x^2 \partial y^2} + \frac{\partial^4 w(x, y, t)}{\partial y^4} + \frac{\rho}{D} \frac{\partial^2 w(x, y, t)}{\partial t^2} = \frac{q(x, y, t)}{D}, \quad (1)$$

where w is the lateral deflection in the z direction [m], ρ is the mass density per unit area [kg/m²], q is the transverse external force, with dimension of force per unit area [N/m²], $\frac{\partial^2 w}{\partial t^2}$ is the acceleration in the z direction [m/s²], $D = Eh^3/(12(1 - \nu^2))$, ν is the Poisson ratio, h is thickness of the plate [m] and E is Young's Modulus [N/m²]. The edges of the mirror are clamped and hence both the mirror deflection at the edge and its derivative are zero. Further details on this PDE can be found in, e.g., [13] and in this paper control action based on an array of actuators and sensors is considered. To derive a model suitable for control design, the use of an actuator array requires the discretization of (1) in the spatial variables. Moreover, since the control will be implemented digitally, (1) must also be discretized with respect to time.

The discretization technique used in this paper is adapted from [4], where a method based on *Crank–Nicolson* discretization [8] was applied to the PDE (1). Also it is known (again see [4] for the details) that the derived approximation is unconditionally numerically stable. To obtain the discrete model, let p, l, m denote time instant t_p and the coordinates of nodal points x_l, y_m , respectively. Applying the *Crank–Nicolson* discretization give the following partial recurrence equation as a finite dimensional discrete model approximation of the PDE (1)

$$\begin{aligned}
 & \frac{1}{16\delta_x^4} (w_{p+2,l+2,m} + w_{p+2,l-2,m}) + \frac{1}{8\delta_x^2\delta_y^2} (w_{p+2,l+1,m+1} + w_{p+2,l+1,m-1} \\
 & + w_{p+2,l-1,m+1} + w_{p+2,l-1,m-1}) + \left(-\frac{1}{4\delta_x^2\delta_y^2} - \frac{1}{4\delta_x^4} \right) (w_{p+2,l+1,m} + w_{p+2,l-1,m}) \\
 & + \frac{1}{16\delta_y^4} (w_{p+2,l,m+2} + w_{p+2,l,m-2}) + \left(-\frac{1}{4\delta_x^2\delta_y^2} - \frac{1}{4\delta_y^4} \right) (w_{p+2,l,m+1} + w_{p+2,l,m-1}) \\
 & + \left(\frac{1}{2\delta_x^2\delta_y^2} + \frac{3}{8\delta_x^4} + \frac{3}{8\delta_y^4} + \frac{\rho}{D\delta_t^2} \right) w_{p+2,l,m} + \frac{1}{8\delta_x^4} (w_{p+1,l+2,m} + w_{p+1,l-2,m}) \\
 & + \frac{1}{4\delta_x^2\delta_y^2} (w_{p+1,l+1,m+1} + w_{p+1,l+1,m-1} + w_{p+1,l-1,m+1} + w_{p+1,l-1,m-1}) \\
 & + \left(-\frac{1}{2\delta_x^2\delta_y^2} - \frac{1}{2\delta_x^4} \right) (w_{p,l+1,m} + w_{p+2,l-1,m}) + \left(-\frac{1}{8\delta_y^4} \right) (w_{p+1,l,m+2} \\
 & + w_{p+1,l,m-2}) + \left(-\frac{1}{2\delta_x^2\delta_y^2} - \frac{1}{2\delta_y^4} \right) (w_{p+1,l,m+1} + w_{p+1,l,m-1}) \\
 & + \left(\frac{1}{\delta_x^2\delta_y^2} + \frac{3}{4\delta_x^4} + \frac{3}{4\delta_y^4} - \frac{2\rho}{D\delta_t^2} \right) w_{p+1,l,m} + \left(\frac{1}{16\delta_x^4} \right) (w_{p,l+2,m} + w_{p,l-2,m}) \\
 & + \frac{1}{8\delta_x^2\delta_y^2} (w_{p,l+1,m+1} + w_{p,l+1,m-1} + w_{p,l-1,m+1} + w_{p,l-1,m-1}) \\
 & + \left(-\frac{1}{4\delta_x^2\delta_y^2} - \frac{1}{4\delta_x^4} \right) (w_{p,l+1,m} + w_{p+2,l-1,m}) + \left(\frac{1}{16\delta_y^4} \right) (w_{p,l,m+2} + w_{p,l,m-2}) \\
 & + \left(-\frac{1}{4\delta_x^2\delta_y^2} - \frac{1}{4\delta_y^4} \right) (w_{p+1,l,m+1} + w_{p+1,l,m-1}) \\
 & + \left(\frac{1}{2\delta_x^2\delta_y^2} + \frac{3}{8\delta_x^4} + \frac{3}{8\delta_y^4} + \frac{\rho}{D\delta_t^2} \right) w_{p,l,m} = \frac{1}{D} q_{p,l,m}. \quad (2)
 \end{aligned}$$

This equation represents the planar equivalent of a singular wave repetitive process, which is second order in discrete time (p) and gives the dependence between the rectangles $(l-2, m+2), \dots, (l+2, m+2), \dots, (l-2, m-2), \dots, (l+2, m-2)$ at sample instant $p+2$ and the same rectangles at the preceding two sample instants, i.e., $p+1$ and p respectively. Moreover, $q_{p,l,m}$ can be considered as a distributed control input to the system. As a practically relevant special case, a square plate is considered and hence in (2) $\delta_x = \delta_y = \delta$ and the total number of the nodes is n^2 .

3 Derivation of the 1D equivalent model

The discrete dynamics of (2) to be written in the matrix form

$$A W_{p+2} + B W_{p+1} + A W_p = C Q_p, \quad (3)$$

where

$$W_p = \begin{bmatrix} w_{p,1,1} \\ w_{p,1,2} \\ \vdots \\ w_{p,n,n} \end{bmatrix}, \quad Q_p = \begin{bmatrix} q_{p,1,1} \\ q_{p,1,2} \\ \vdots \\ q_{p,n,n} \end{bmatrix}, \quad (4)$$

and A, B and C are $n^2 \times n^2$ Toeplitz matrices constructed from the coefficients of (2). These matrices can also be written as

$$A = \text{pent}_n(\mathbb{R}_1, \mathbb{P}_1, \mathbb{Q}_1), \quad B = \text{pent}_n(\mathbb{R}_2, \mathbb{P}_2, \mathbb{Q}_2), \quad C = \text{odiag}_n(D^{-1}) \quad (5)$$

and

$$\begin{aligned} \mathbb{R}_1 &= \text{pent}_n(S_1, R_{11}, R_{12}), \quad \mathbb{P}_1 = \text{tri}_n(Q_{11}, P_1), \quad \mathbb{Q}_1 = \text{odiag}_n(Q_{12}), \\ \mathbb{R}_2 &= \text{pent}_n(S_2, R_{21}, R_{22}), \quad \mathbb{P}_2 = \text{tri}_n(Q_{21}, P_2), \quad \mathbb{Q}_2 = \text{odiag}_n(Q_{22}), \end{aligned} \quad (6)$$

where

$$\begin{aligned} P_1 &= \frac{1}{2\delta^4}, \quad Q_{11} = -\frac{2}{\delta^4}, \quad Q_{12} = \frac{1}{4\delta^4}, \quad R_{11} = -\frac{2}{\delta^4}, \quad R_{12} = \frac{1}{4\delta^4}, \\ S_1 &= \frac{\rho}{D\delta_t^2} + \frac{5}{\delta^4}, \quad P_2 = \frac{1}{\delta^4}, \quad Q_{21} = -\frac{4}{\delta^4}, \quad Q_{22} = \frac{1}{2\delta^4}, \quad R_{21} = -\frac{4}{\delta^4}, \\ R_{22} &= \frac{1}{2\delta^4}, \quad S_2 = \frac{10}{\delta^4} - \frac{2\rho}{D\delta_t^2}. \end{aligned} \quad (7)$$

Assuming that matrix A is nonsingular, (3) can be written in the form

$$W_{p+2} = -A^{-1} B W_{p+1} - W_p + A^{-1} C Q_p. \quad (8)$$

which is a standard discrete linear difference equation, also termed 1D in some of the literature.

4 ILC problem formulation

Rewrite (8) in the form

$$W_{p+2} = \hat{A}_1 W_{p+1} + \hat{A}_2 W_p + \hat{B} Q_p, \quad (9)$$

where

$$\hat{A}_1 = -A^{-1} B, \quad \hat{A}_2 = -I, \quad \hat{B} = A^{-1} C. \quad (10)$$

This is a second-order difference equation and to transform it to first order form introduce

$$\mathbf{W}_p = \begin{bmatrix} W_{p+1} \\ W_p \end{bmatrix}, \quad \mathbf{Q}_p = Q_p \tag{11}$$

Hence the state equation is

$$\mathbf{W}_{p+1} = \mathbf{A}\mathbf{W}_p + \mathbf{B}\mathbf{Q}_p \tag{12}$$

with $p \geq 0$ and

$$\mathbf{A} = \begin{bmatrix} \hat{A}_1 & \hat{A}_2 \\ I & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \hat{B} \\ 0 \end{bmatrix}. \tag{13}$$

To formulate the ILC design problem, a positive integer variable k denoting the pass-to-pass update is introduced. Then (12) can be written as

$$\mathbf{W}_{p+1}(k) = \mathbf{A}\mathbf{W}_p(k) + \mathbf{B}\mathbf{Q}_p(k). \tag{14}$$

Introduce the output equation

$$\mathbf{Y}_p(k) = \mathbf{C}\mathbf{W}_p(k) = W_{p+1}(k) \tag{15}$$

with

$$\mathbf{C} = [I \ 0]. \tag{16}$$

Also define the tracking error $\mathbf{E}_p(k)$ as

$$\mathbf{E}_p(k) = \mathbf{Y}_p^* - \mathbf{Y}_p(k) = W_{p+1}^* - W_{p+1}(k), \tag{17}$$

where \mathbf{Y}_p^* denotes a spatial/temporal reference signal.

Introduce the state and control increments as

$$\begin{aligned} \Theta_{p+1}(k+1) &= \mathbf{W}_p(k+1) - \mathbf{W}_p(k), \\ \Delta\mathbf{Q}_p(k+1) &= \mathbf{Q}_p(k+1) - \mathbf{Q}_p(k) \end{aligned} \tag{18}$$

and apply the ILC law (where $\mathbf{K}_1, \mathbf{K}_2$ are the controller gain matrices)

$$\Delta\mathbf{Q}_p(k+1) = \mathbf{K}_1 \Theta_{p+1}(k+1) + \mathbf{K}_2 \mathbf{E}_{p+1}(k) \tag{19}$$

to obtain the following ILC dynamics written in the form of a discrete linear repetitive process [11]

$$\begin{aligned} \Theta_{p+1}(k+1) &= \bar{A}\Theta_p(k+1) + \bar{B}\mathbf{E}_p(k), \\ \mathbf{E}_p(k+1) &= \bar{C}\Theta_p(k+1) + \bar{D}\mathbf{E}_p(k), \end{aligned} \tag{20}$$

where

$$\bar{A} = \mathbf{A} + \mathbf{B}\mathbf{K}_1, \quad \bar{B} = \mathbf{B}\mathbf{K}_2, \quad \bar{C} = -\mathbf{C}\bar{A}, \quad \bar{D} = I - \mathbf{C}\bar{B}. \tag{21}$$

Also the control input vector for (14) can be computed from (19) as

$$\mathbf{Q}_p(k) = \mathbf{Q}_p(k-1) + \mathbf{K}_1(\mathbf{W}_p(k) - \mathbf{W}_p(k-1)) + \mathbf{K}_2(\mathbf{Y}_{p+1}^* - \mathbf{Y}_{p+1}(k-1)). \tag{22}$$

5 ILC design

Introduce the Lyapunov function

$$V_p(k) = \Theta_p(k+1)^T P_1 \Theta_p(k+1) + \mathbf{E}_p(k)^T P_2 \mathbf{E}_p(k) \quad (23)$$

where $P_1 \succ 0$ and $P_2 \succ 0$ and the associated increment

$$\begin{aligned} \Delta V_p(k) = & \Theta_{p+1}(k+1)^T P_1 \Theta_{p+1}(k+1) - \Theta_p(k+1)^T P_1 \Theta_p(k+1) \\ & + \mathbf{E}_p(k+1)^T P_2 \mathbf{E}_p(k+1) - \mathbf{E}_p(k)^T P_2 \mathbf{E}_p(k). \end{aligned} \quad (24)$$

Then using the stability theory for discrete linear repetitive processes [11], ILC pass-to-pas convergence occurs when

$$\Delta V_p(k) < 0, \quad \forall k, p > 0, \quad (25)$$

or

$$\begin{bmatrix} \bar{A}^T P_1 \bar{A} - P_1 + \bar{C}^T P_2 \bar{C} & \bar{A}^T P_1 \bar{B} + \bar{C}^T P_2 \bar{D} \\ \bar{B}^T P_1 \bar{A} + \bar{D}^T P_2 \bar{C} & \bar{B}^T P_1 \bar{B} + \bar{D}^T P_2 \bar{D} - P_2 \end{bmatrix} \prec 0. \quad (26)$$

Introduce

$$\bar{\mathcal{A}} = \begin{bmatrix} \bar{A} & \bar{B} \\ \bar{C} & \bar{D} \end{bmatrix}, \quad \bar{\mathcal{P}} = \text{diag}(P_1, P_2) \quad (27)$$

to rewrite (26) in the form

$$\bar{\mathcal{A}}^T \bar{\mathcal{P}} \bar{\mathcal{A}} - \bar{\mathcal{P}} \prec 0, \quad (28)$$

which, however, is not an LMI.

Rewrite $\bar{\mathcal{A}}$ as

$$\bar{\mathcal{A}} = \begin{bmatrix} \mathbf{A} + \mathbf{BK}_1 & \mathbf{BK}_2 \\ -\mathbf{CA} - \mathbf{CBK}_1 & I - \mathbf{CBK}_2 \end{bmatrix} \quad (29)$$

and introduce

$$\begin{aligned} \bar{\mathbb{A}} &= \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix}, & \bar{\mathbb{B}} &= \begin{bmatrix} \mathbf{B} & \mathbf{B} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, & \bar{\mathbb{C}} &= \begin{bmatrix} I & \mathbf{0} \\ -\mathbf{C} & I \end{bmatrix}, \\ \mathbb{K} &= \text{diag}(\mathbf{K}_1, \mathbf{K}_2), & \mathbb{A} &= \bar{\mathbb{C}} \bar{\mathbb{A}}, & \mathbb{B} &= \bar{\mathbb{C}} \bar{\mathbb{B}} \end{aligned} \quad (30)$$

to obtain (29) in the form

$$\bar{\mathcal{A}} = \mathbb{A} + \mathbb{BK}. \quad (31)$$

Then, (28) becomes

$$(\mathbb{A} + \mathbb{BK})^T \bar{\mathcal{P}} (\mathbb{A} + \mathbb{BK}) - \bar{\mathcal{P}} \prec 0, \quad (32)$$

and the major result of this paper can now be established.

Theorem 1. *The discrete linear repetitive process describing the ILC dynamics (20) is stable along the pass and hence ILC pass-to-pass error convergence occurs if there exists $\tilde{\mathcal{P}} \succ 0$, where $\tilde{\mathcal{P}} = \text{diag}(\tilde{\mathcal{P}}_1, \tilde{\mathcal{P}}_2)$, and $\tilde{\mathcal{N}} = \text{diag}(\tilde{\mathcal{N}}_1, \tilde{\mathcal{N}}_2)$ such that the LMI*

$$\begin{bmatrix} -\tilde{\mathcal{P}} & \tilde{\mathcal{P}}\mathbb{A}^T + \tilde{\mathcal{N}}^T\mathbb{B}^T \\ \mathbb{A}\tilde{\mathcal{P}} + \mathbb{B}\tilde{\mathcal{N}} & -\tilde{\mathcal{P}} \end{bmatrix} \prec 0, \quad (33)$$

is feasible. If this LMI is feasible, the control law matrices \mathbf{K}_1 and \mathbf{K}_2 in (19) can be computed using

$$\mathbb{K} = \tilde{\mathcal{N}}\tilde{\mathcal{P}}^{-1} = \text{diag}(\mathbf{K}_1, \mathbf{K}_2) = \text{diag}(\tilde{\mathcal{N}}_1\tilde{\mathcal{P}}_1^{-1}, \tilde{\mathcal{N}}_2\tilde{\mathcal{P}}_2^{-1}). \quad (34)$$

Proof. An obvious application of the Schur's complement formula to (32) gives

$$\begin{bmatrix} -\bar{\mathcal{P}} & (\mathbb{A} + \mathbb{B}\mathbb{K})^T \\ \mathbb{A} + \mathbb{B}\mathbb{K} & -\bar{\mathcal{P}}^{-1} \end{bmatrix} \prec 0. \quad (35)$$

Next, pre- and post-multiply this last inequality by $\text{diag}(\bar{\mathcal{P}}^{-1}, I)$ to obtain

$$\begin{bmatrix} -\bar{\mathcal{P}}^{-1} & \bar{\mathcal{P}}^{-1}\mathbb{A}^T + \bar{\mathcal{P}}^{-1}\mathbb{K}^T\mathbb{B}^T \\ \mathbb{A}\bar{\mathcal{P}}^{-1} + \mathbb{B}\mathbb{K}\bar{\mathcal{P}}^{-1} & -\bar{\mathcal{P}}^{-1} \end{bmatrix} \prec 0. \quad (36)$$

Finally, set $\tilde{\mathcal{P}} = \bar{\mathcal{P}}^{-1}$ and $\tilde{\mathcal{N}} = \mathbb{K}\tilde{\mathcal{P}}$ to obtain (33) and the proof is complete.

6 Numerical example

The dynamics of a square 1×1 m deformable mirror is considered in the case when the thickness $h = 0.003$ m, mass density per unit area $\rho = 2700$ kg m⁻², Young's Modulus $E = 7.11 \cdot 10^{10}$ N m⁻², Poisson ratio $\nu = 0.3$, and discretization by applying the rectangular (square) regular grid developed in Section 2. By assumption, the edges of the mirror are clamped and hence both the mirror deflection at the edges and their derivative are zero, which leads to the boundary conditions

$$\begin{aligned} w_{p,l,m} &= 0, \\ w_{p+1,l+1,m} - w_{p+1,l,m} + w_{p,l+1,m} - w_{p,l,m} &= 0, \\ w_{p+1,l,m+1} - w_{p+1,l,m} + w_{p,l,m+1} - w_{p,l,m} &= 0 \end{aligned} \quad (37)$$

for $l = -1, 0$, or $m = -1, 0$, or $l = n + 1, n + 2$, or $m = n + 1, n + 2$.

The number of nodes in both the dimensions (square case) is $n = 7$ and including boundaries $N = n + 2 = 9$. These values result in $\delta = \delta_x = \delta_y = 0.125$ m. Also it is assumed that the sampling period is $\delta_t = 1$ secs, which is quite large but allowed by the *Crank-Nicolson* method. The initial conditions are assumed to be zero and the duration of the reference signal shown in the left-hand side plot in Fig. 1 is $t_f = 11$ secs (sample instants $p = 0, 1, \dots, 11$). The middle point

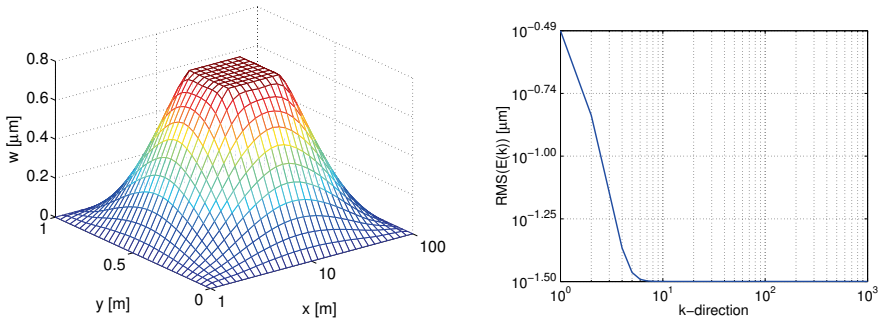


Fig. 1. Reference signal at time instants $t = 4, 5, 6, 7$ secs (left) and RMS of the ILC design (right).

of the plate corresponds to the maximum value of the reference trajectory, which is taken to be symmetric in its time duration, i.e., at the time instant $t = 0$ secs the trajectory is zero for the whole plate and then rises linearly to achieve its maximum deflection at time instant $t = 4$ secs. Its value is then constant to $t = 7$ secs, when it begins to decrease linearly to zero at $t = 11$ secs.

Applying Theorem 1 results in the control law matrices \mathbf{K}_1 and \mathbf{K}_2 of (34) that are not shown here due to their large dimensions ($n^2 \times 2n^2 = 49 \times 98$ and $n^2 \times n^2 = 49 \times 49$). However, to limit the number of actuators it is assumed that the spatially constant input is adjusted only at a selection of the central plate grid points, particularly 9 of the total number of 49 points. This spatially constant control signal value is calculated for each time instant as the mean of the original input signals from the points $(l-1, m+1), \dots, (l, m), \dots, (l+1, m-1)$.

To examine ILC convergence the Root Mean Square (RMS) error along the trials is defined as

$$\text{RMS}(\mathbf{E}(k)) = \sqrt{(\mathbf{E}(k)^T \mathbf{E}(k)) / (\beta n^2)} \quad (38)$$

where $\beta = 12$ is the number of sample instants. The right-hand side plot in Fig. 1 shows this quantity for the numerical data considered and validates the design.

7 Conclusion

This paper has developed an ILC design for controlling the vibrations of a thin plate described by a PDE in Lagrange form by applying the *Crank–Nicolson* discretization scheme to construct an unconditionally stable discrete approximate model of the dynamics. To avoid an extremely large dimensioned model, resulting in the requirement for a large number of actuators, it is assumed that the control action is adjusted a selection of them and also is constant space-wise. Further research should include actuator dynamics and the development of robust control methods and fault tolerant control. Progress in these areas will

advance the case for experimental verification. Also similar approach could be applied to systems governed by different classes of PDEs, provided the *Crank-Nicolson* discretization produces an unconditionally stable approximation. Other discretization methods, e.g., finite element methods, which find many applications in solving practical problems, see, e.g. [10] will also be investigated.

References

1. Ahn, H.S., Chen, Y., Moore, K.: Iterative Learning Control: Brief Survey and Categorization. *Systems, Man, and Cybernetics, Part C: Applications and Reviews*, IEEE Transactions on 37(6), 1099–1121 (2007)
2. Arimoto, S., Kawamura, S., Miyazaki, F.: Bettering operation of robots by learning. *Journal of Robotic Systems* 2(1), 123–140 (1984)
3. Augusta, P., Cichy, B., Gałkowski, K., Rogers, E.: An unconditionally stable finite difference scheme systems described by second order partial differential equations. In: *The 2015 IEEE 9th International Workshop on Multidimensional (nD) Systems (nDS)*. pp. 134–139. IEEE (2015)
4. Augusta, P., Cichy, B., Gałkowski, K., Rogers, E.: An unconditionally stable approximation of a circular flexible plate described by a fourth order partial differential equation. In: *Proceedings of the 21st International Conference on Methods and Models in Automation and Robotics*. pp. 1039–1044 (2016)
5. Bristow, D., Tharayil, M., Alleyne, A.: A survey of iterative learning control. *Control Systems*, IEEE 26(3), 96–114 (2006)
6. Cichy, B., Gałkowski, K., Rogers, E.: Iterative learning control for spatio-temporal dynamics using Crank-Nicolson discretization. *Multidimensional Systems and Signal Processing* 23, 185–208 (2012)
7. Cichy, B., Gałkowski, K., Rogers, E., Kummert, A.: An approach to iterative learning control for spatio-temporal dynamics using nD discrete linear systems models. *Multidimensional Systems and Signal Processing* 22, 83–96 (2011)
8. Crank, J., Nicolson, P.: A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. *Proceedings of the Cambridge Philosophical Society* 43, 50–67 (1947)
9. Freeman, C.T., Tong, D., Meadmore, K.L., Cai, Z., Rogers, E., Hughes, A.M., Burrige, J.H.: *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 225(6), 850–859 (2011)
10. Rauh, A., Senkel, L., Aschemann, H., Saurin, V.V., Kostin, G.V.: An integro-differential approach to modeling, control, state estimation and optimization for heat transfer systems. *International Journal of Applied Mathematics and Computer Science* 26(1), 15–30 (2016)
11. Rogers, E., Gałkowski, K., Owens, D.H.: *Control Systems Theory and Applications for Linear Repetitive Processes*, Lecture Notes in Control and Information Sciences, vol. 349. Springer (2007)
12. Strikwerda, J.C.: *Finite difference schemes and partial differential equations*. Belmont: Wadsworth and Brooks (1989)
13. Timoshenko, S., Woinowski-Krieger, S.: *Theory of plates and shells*. New York: McGraw-Hill (1959)

Iterative Learning Control for a discretized sub-class of spatially interconnected systems

Bartłomiej Sulikowski¹, Krzysztof Galkowski¹, and Eric Rogers²

¹ Institute of Control and Computation Engineering, University of Zielona Góra, ul. Podgórna 50, 65-246 Zielona Góra, Poland,

{b.sulikowski,k.galkowski}@issi.uz.zgora.pl

² Department of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK etar@ecs.soton.ac.uk

Abstract. This paper develops an Iterative Learning Control (ILC) design applied to an example, active long ladder circuits, of spatially interconnected systems. The design is based on writing the dynamics as those of an equivalent standard differential linear system and then converting to the discrete domain. For this reason, a standard linear systems equivalent model is developed and then discretized. A numerical case study is also given to illustrate the new design.

Keywords: Spatially interconnected systems, Ladder circuits, Iterative Learning Control.

1 Introduction

Electrical ladder networks or circuits (also known as ladder systems or series-shunt networks) are formed by a chain of cells all of which are assumed to be identical in structure. The cells are realized by longitudinal and transversal resistances, reactances, or, in general, impedances. Due to their inherent time and spatial dynamics, they form a particular case of spatially interconnected systems, see, e.g., [1].

Ladder circuits have many possible applications, such as filter analysis and design, modeling delay lines and equivalent circuits for transmission lines, chains of transmission gates or long wire interconnections, e.g. [2], in the approximation of distributed parameter systems, e.g. [3] and in the simulation of physical systems. In this paper, the emphasis is on developing underlying systems theory.

Two-dimensional, or 2D systems, propagate information in two independent directions, where the indeterminates can both be discrete, both continuous or one continuous and one discrete. Ladder circuits can be modeled as 2D systems where the continuous or discrete time variable and the node number, a discrete variable, are the indeterminates. However, due to the left–right and right–left dependence between neighboring cells models ladder circuits have dynamics that cannot be represented by the commonly used 2D Roesser [4] and the Fornasini Marchesini [5] state-space models. In particular, 2D causality, which is defined over the right upper quadrant of the 2D plane, does not apply in this case since

if this property was present, then any ladder circuit cell could only influence its right-hand side neighbor cell. Moreover, the stability analysis for these 2D systems cannot be used.

One setting for the analysis and design control schemes of ladder circuits with a finite number of circuit nodes is to use a form of lifting based on the discrete spatial variable, i.e., the node number. In this approach the state vectors of the cells are assembled into a single vector, starting with the first cell and continuing to the last and likewise for the input and output vectors. The result is a standard, also termed 1D in 2D systems analysis, linear systems state-space model, see [6] for a detailed treatment. Hence standard linear systems theory can be used.

The construction of discrete models of the dynamics of these systems encounters problems similar to those for systems described by partial differential equations (PDE) and is a nontrivial task. This paper avoids such difficulties by constructing the discrete representation of the dynamics from the 1D equivalent model, see [7] for further discussion. Having constructed the discrete model, this paper then develops an ILC design for the ladder networks considered.

Throughout this paper $M \succ 0$ (respectively $\prec 0$) denotes a real symmetric positive (respectively negative) definite matrix. The null and identity matrices with the required dimensions are denoted by 0 and I , respectively, the symbol $\text{diag}\{W_1, W_2, \dots, W_M\}$ denotes a block diagonal matrix with diagonal blocks W_1, W_2, \dots, W_M , and also

$$\text{tri}\{\beta, \gamma, \eta\} \triangleq \begin{bmatrix} \gamma & \beta & & 0 \\ \eta & \gamma & \beta & \\ & \ddots & \ddots & \ddots \\ & & \eta & \gamma & \beta \\ 0 & & & \eta & \gamma \end{bmatrix}.$$

2 Background

Previous work, (e.g., [6] and references therein) has established that ladder circuits can be described by a 2D differential-discrete state-space model of the following form over $p = 0, 1, \dots, \alpha - 1$,

$$\frac{d}{dt}x(p, t) = \mathcal{A}_1x(p - 1, t) + \mathcal{A}_2x(p, t) + \mathcal{A}_3x(p + 1, t) + \mathcal{B}u(p, t), \quad (1)$$

where p and t denote the node number and time, respectively. Also, $x(p, t)$ and $u(p, t)$, respectively, denote the state and input vectors.

Consider a particular case of the active ladder circuit of the form shown in Figure 1, where $i(p, t) = \gamma U_c(p - 1, t)$ and the controlled sources $E(p, t) = u(p, t)$ added to the nodes represent possible control input variables.

Let

$$x(p, t) = \begin{bmatrix} U_C(p, t) \\ i_L(p, t) \end{bmatrix}. \quad (2)$$

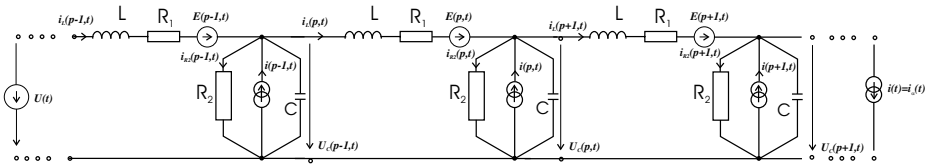


Fig. 1. A ladder chain.

denote the state vector of the p -th ladder cell. Then, the matrices in the model (1) are

$$\mathcal{A}_1 = \begin{bmatrix} \frac{\gamma}{C} & 0 \\ \frac{1}{L} & 0 \end{bmatrix}, \mathcal{A}_2 = \begin{bmatrix} -\frac{1}{R_2 C} & \frac{1}{C} \\ -\frac{1}{L} & -\frac{R_1}{L} \end{bmatrix}, \mathcal{A}_3 = \begin{bmatrix} 0 & -\frac{1}{C} \\ 0 & 0 \end{bmatrix}, \mathcal{B} = \begin{bmatrix} 0 \\ -\frac{1}{L} \end{bmatrix}. \tag{3}$$

and the output equation is

$$y(p, t) = \mathcal{C}x(p, t), \tag{4}$$

where the matrix \mathcal{C} is to be specified depending on the particular circuit considered.

To complete the description, the following boundary conditions are assumed

$$\begin{aligned} x(-1, t) &= \begin{bmatrix} U(t) \\ 0 \end{bmatrix}, \quad x(\alpha, t) = \begin{bmatrix} 0 \\ i(t) \end{bmatrix}, \\ x(p, 0) &= 0, \quad 0 \leq p \leq \alpha - 1. \end{aligned} \tag{5}$$

3 1D equivalent model and its discretization

As discussed previously, the ladder circuit systems considered in this paper can be represented by a 2D continuous-discrete state-space model, which is not upper right quadrant casual and hence the systems theory for Roesser [4] or Fornasini Marchesini [5] state-space models, see, e.g., [8] is not applicable. To provide a setting for control relevant analysis the route used is to write the model in 1D equivalent form using the state, input and output super-vectors $\mathbf{x}(t)$, $\mathbf{u}(t)$ and $\mathbf{y}(t)$, where

$$\mathbf{x}(t) = [x(0, t)^T, x(1, t)^T, \dots, x(\alpha - 1, t)^T]^T, \tag{6}$$

$$\mathbf{u}(t) = [u(0, t)^T, u(1, t)^T, \dots, u(\alpha - 1, t)^T]^T. \tag{7}$$

and

$$\mathbf{y}(t) = [y(0, t)^T, y(1, t)^T, \dots, y(\alpha - 1, t)^T]^T.$$

This leads to the augmented state-space model description of the dynamics

$$\begin{aligned} \frac{d}{dt} \mathbf{x}(t) &= \Phi \mathbf{x}(t) + \Psi \mathbf{u}(t), \\ \mathbf{y}(t) &= \Gamma \mathbf{x}(t), \end{aligned} \tag{8}$$

where

$$\Phi = \text{tri}\{\mathcal{A}_3, \mathcal{A}_2, \mathcal{A}_1\}, \Psi = \text{diag}\{\mathcal{B}, \mathcal{B}, \dots, \mathcal{B}\}, \Gamma = \text{diag}\{\mathcal{C}, \mathcal{C}, \dots, \mathcal{C}\}, \quad (9)$$

and Φ is a block tridiagonal Toeplitz matrix. In this representation the dynamics in p is absorbed into the super-vectors.

As discussed previously in this paper, direct discretization of the dynamics of a ladder circuit encounters difficulties that are not present for discretization of the equivalent 1D model. In this paper, a commonly used invariant impulse response formula is applied to obtain a discrete approximation of (8) in the form

$$\begin{aligned} \mathbf{x}(l+1) &= \mathbf{A}\mathbf{x}(l) + \mathbf{B}\mathbf{u}(l), \\ \mathbf{y}(l) &= \mathbf{C}\mathbf{x}(l), \end{aligned} \quad (10)$$

where l denotes the sampling instances.

4 Iterative Learning Control of the discretized model

Many physical systems are required to repeat the same finite duration operation over and over again. The sequence is that the system completes an execution and then resets to the starting location and the next execution can begin, either immediately or after a further period of time has elapsed. Each execution is known as a trial, or pass, and the finite duration of operation the trial (or pass) length. Given a reference trajectory the error on each trial can be constructed and hence the sequence where each member is the error along a trial. The ILC design problem is to force this error sequence to converge using a control input on the current trial that includes a contribution from the error on the previous trial or a finite number of thereof.

Since the first work, widely credited to [9], ILC has been a continually expanding area of research. More recent applications with experimental support include [10–12]. In ILC information propagates in two independent directions, i.e., along the trials and from trial-to-trial. Hence it is a 2D system and therefore the systems theory for this general area is available for use in design.

To formulate the ILC design problem for the systems considered in this paper let the nonnegative integer k denote the trial number and rewrite (10) as

$$\begin{aligned} \mathbf{x}_k(l+1) &= \mathbf{A}\mathbf{x}_k(l) + \mathbf{B}\mathbf{u}_k(l) \\ \mathbf{y}_k(l) &= \mathbf{C}\mathbf{x}_k(l) \end{aligned} \quad (11)$$

where on trial k $\mathbf{u}_k(l) \in \mathbb{R}^{r\alpha}$ is the input supervector, $\mathbf{y}_k(l) \in \mathbb{R}^{m\alpha}$ is the output supervector and $\mathbf{x}_k(l) \in \mathbb{R}^{n\alpha}$ is the state supervector. Also let $\mathbf{y}_{ref}(k)$ denote the reference vector, consisting of the values to be attained along the ladder and suppose that the duration of each trial is β , i.e., $l \in [0, \beta]$. The boundary conditions are

$$\begin{aligned} \mathbf{x}_k(0) &= x_0, \quad k = 0, 1, \dots, \\ \mathbf{u}_0(l) &= 0, \quad l = 0, 1, \dots, \beta. \end{aligned} \quad (12)$$

In application, ILC is designed to ensure that the error sequence converges from trial-to-trial and also regulate the dynamics along the trials, where

$$\mathbf{e}_k(l) = \mathbf{y}_{ref}(l) - \mathbf{y}_k(l). \quad (13)$$

Most often an ILC law constructs the current trial input as the sum of previous trial input plus a correction term computed using previous trial data, i.e.,

$$\mathbf{u}_{k+1}(l) = \mathbf{u}_k(l) + \Delta \mathbf{u}_{k+1}(l), \quad (14)$$

where $\Delta \mathbf{u}_{k+1}(l)$ denotes the correction term.

Introduce, for analysis purposes only,

$$\eta_{k+1}(l) = \mathbf{x}_{k+1}(l-1) - \mathbf{x}_k(l-1).$$

Then combining (11), (13) and (14) gives

$$e_{k+1}(l) - e_k(l) = -\mathbf{CA}\eta_{k+1}(l) - \mathbf{CB}\Delta \mathbf{u}_k(l-1)$$

and using (11) and (14),

$$\eta_k(l+1) = \mathbf{A}\eta_k(l) + \mathbf{B}\Delta \mathbf{u}_k(l-1).$$

Also, the ILC law correction term can be written as

$$\Delta \mathbf{u}_{k+1}(l) = K_1 \eta_{k+1}(l+1) + K_2 e_k(l+1), \quad (15)$$

and hence the controlled ILC dynamics can be written as

$$\begin{bmatrix} \eta_{k+1}(l+1) \\ e_{k+1}(l) \end{bmatrix} = \begin{bmatrix} \mathbf{A} + \mathbf{BK}_1 & \mathbf{BK}_2 \\ -\mathbf{CA} - \mathbf{CBK}_1 & I - \mathbf{CBK}_2 \end{bmatrix} \begin{bmatrix} \eta_{k+1}(l) \\ e_k(l) \end{bmatrix}. \quad (16)$$

which is in the form of a discrete linear repetitive process [13].

Repetitive processes are a distinct class of 2D systems characterized by a number of sweeps, or passes, through a set of dynamics. On each pass the output is termed the pass profile and the 2D structure arises from the previous pass profile acting as a forcing function on, and hence contributing to, the dynamics of the next pass profile. The result can be the presence of oscillations that increase in amplitude from pass-to-pass that cannot be removed by standard control action. The finite pass, or trial, length makes repetitive processes a more natural setting for ILC design. A considerable volume of literature exists on ILC designs in this setting, see, e.g., [10, 11], with experimental verification. To complete the description, the boundary conditions are taken as

$$\begin{aligned} \eta_k(0) &= \mathbf{x}_{k+1}(0) - \mathbf{x}_k(0) = \mathbf{x}_0 - \mathbf{x}_0 = 0, \quad k = 0, 1, \dots, \\ e_0(l) &= \mathbf{y}_{ref}(l) - \mathbf{y}_0(l) = \mathbf{y}_{ref}(l) - \mathbf{CA}^l \mathbf{x}_0, \quad l = 0, 1, \dots, \beta. \end{aligned} \quad (17)$$

To conform with the ILC literature pass is replaced by trial in the remainder of this paper.

The stability theory [13] for linear repetitive process is of the bounded input bounded output form where a bounded initial trial profile is required to produce a bounded sequence of trial profiles, where the boundedness property is defined in terms of the norm on the underlying function space. Two forms of stability are possible, where the first of these, known as asymptotic stability, requires the boundedness property over the finite and fixed trial length whereas stability along the trial is stronger, since it requires this property uniformly, i.e., for all possible trial lengths. Design for stability along the trial is a one step procedure that produces a control law that enforces trial-to-trial error convergence and regulates the dynamics along the trials.

Applied to ILC, asymptotic stability is sufficient to ensure trial-to-trial error convergence but, as the trial length is finite, this property holds even if the state matrix is unstable, i.e., not all of its eigenvalues lie in the open unit circle in the complex plane. The along the trial dynamics of such a design are not acceptable and hence stability along the trial is used. In the current application, the structure of (16) suggests the following form for the control law matrices K_1 and K_2

$$K_1 = \text{tri}(K_1^1, K_1^2, K_1^3), K_2 = \text{diag}(K^2, K^2, \dots, K^2), \tag{18}$$

which leads to the following result.

Theorem 1. *The discrete linear repetitive process (16) with boundary conditions (17) is stable along the trial if there exist compatibly dimensioned matrices $P = \text{diag}(P_1, P_2) \succ 0$ and N_1, N_2 such that the following LMI is feasible*

$$\begin{bmatrix} -P & P\Upsilon^T + N^T\Omega^T \\ P\Upsilon + \Omega N & -P \end{bmatrix} \prec 0, \tag{19}$$

where

$$\begin{aligned} \Upsilon &= \begin{bmatrix} \mathbf{A} & 0 \\ -\mathbf{CA} & I \end{bmatrix}, \quad \Omega = \begin{bmatrix} \mathbf{B} & 0 \\ 0 & -\mathbf{CB} \end{bmatrix}, \quad N = \begin{bmatrix} N_1 & N_2 \\ N_1 & N_2 \end{bmatrix}, \\ P_1 &= \text{diag}(P^1, P^1, \dots, P^1), \quad P_2 = \text{diag}(P^2, P^2, \dots, P^2), \\ N_1 &= \text{tri}(N_1^1, N_1^2, N_1^3), \quad N_2 = \text{diag}(N^2, N^2, \dots, N^2). \end{aligned}$$

If this LMI is feasible, the control law matrices are given by

$$K_1 = N_1 P_1^{-1}, \quad K_2 = N_2 P_2^{-1}.$$

This design applies the same control action at each node. A generalization to apply different control action at each node is possible. This would increase the number of decision variables over which the LMI has to be solved but would also lead to a reduction in the LMI conservativeness.

5 Numerical example

Consider the active circuit of the form of Figure 1 described by (1) where each cell is constructed from elements with the following values: $C = 3 \times 10^{-4} [F]$, $L =$

$4 \times 10^{-3} [H]$, $R_1 = 20 [\Omega]$, $R_2 = 200 [\Omega]$ and $\gamma = 0.006$. The length of ladder is set to $\alpha = 20$, the sampling period has been chosen as $T = 0.01 [s]$ and the length of the trial has been set to $\gamma = 100$ complete the model, the boundary conditions are chosen as

$$x(-1, t) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, x(\alpha, t) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, x(p, 0) = 0, 0 \leq p \leq \alpha - 1.$$

As a distributed output, the voltage R_1 along the ladder, i.e. $y(p, t) = U_{R_1}(p, t)$ has been chosen and hence $C = [0 \ R_1]$.

The discrete state-space model for design has been obtained by application of the well known invariant impulse response method. This results in very large dimensioned model matrices **A** and **B** that are not given due to space limitations, but the banded structure of their differential counterparts is retained.

Application of Theorem 1 gives the following control law matrices

$$K_1^1 = [1.0106 \ -0.9698], \quad K_1^2 = [0.5111 \ 1.1530], \\ K_1^3 = [-1.2694 \ -1.1469], \quad K^2 = 0.8375.$$

To evaluate this design, simulations have been run for the reference signal shown in Figure 2.

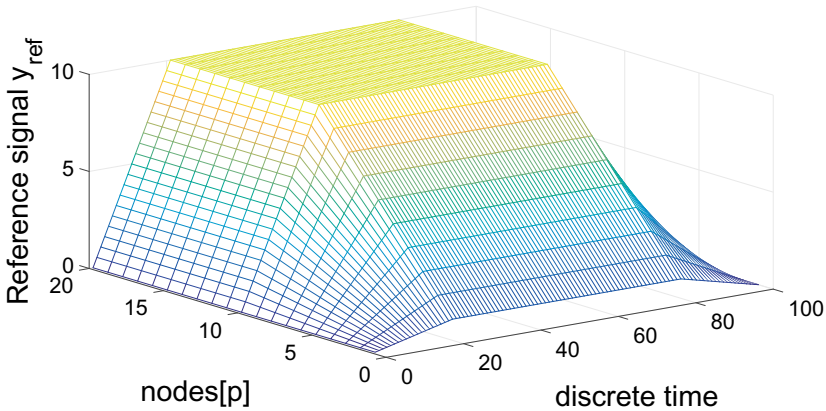


Fig. 2. The reference signal for the ladder circuit example.

The results of the simulation are shown in Figure 3. To assess the convergence of the ILC law the root mean square error along the trials is used

$$\sqrt{\frac{1}{\beta} \sum_{l=0}^{\beta} \mathbf{e}_k^T(l) \mathbf{e}_k(l)} \tag{20}$$

and given in Figure 4. The control signal applied at node 15 is shown in Figure 5. These results demonstrate that highly acceptable performance is possible from the ILC design developed in this paper.

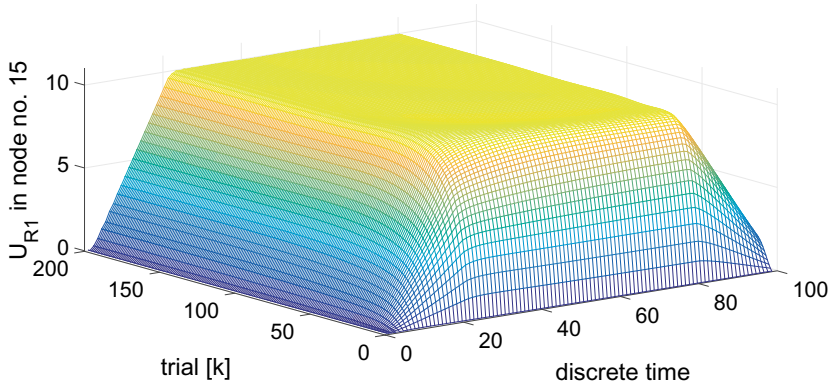


Fig. 3. The voltage across R_1 in node 15.

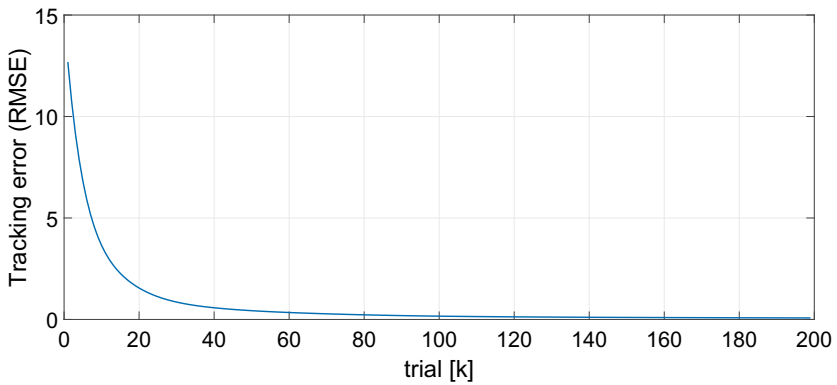


Fig. 4. Tracking error of the voltage across R_1 along the ladder over the trials.

6 Conclusions and Future Work

This paper has developed a new ILC design using repetitive process stability theory for a class of spatially interconnected dynamic systems. As a particular case, long, finite ladder circuits are considered. The resulting design has been illustrated in a simulation study, which has confirmed that high performance is possible. Possible future research include robustness analysis, improvement of the trial-to-trial error convergence speed and consideration of the more complicated structures.

References

1. DAndrea, R., Dullerud, G.: Distributed control design for spatially interconnected systems. *IEEE Transactions on Automatic Control* **48**(9) (2003) 1478–1495

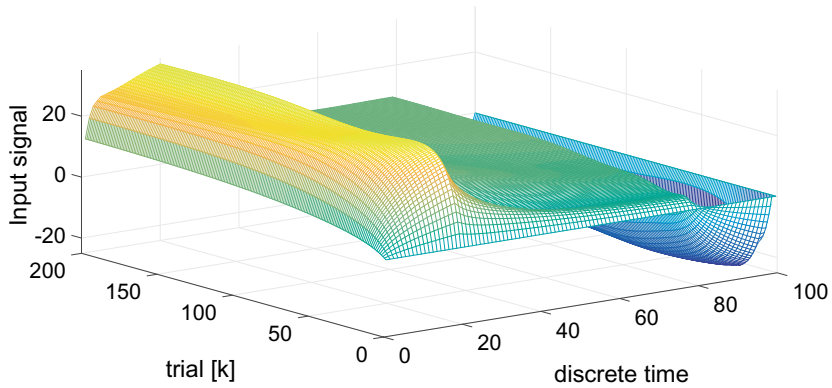


Fig. 5. The input signal used in node 15

2. Alioto, M., Palumbo, G., Poli, M.: Evaluation of energy consumption in RC ladder circuits driven by a ramp input. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* **12**(10) (2004) 1094–1107
3. Schanbacher, T.: Aspects of positivity in control theory. *SIAM Journal on Control and Optimization* **27**(3) (1989) 457–475
4. Roesser, R.P.: A discrete state-space model for linear image processing. *IEEE Transactions on Automatic Control* **20**(1) (1975) 1–10
5. Fornasini, E., Marchesini, G.: Doubly-indexed dynamical systems: state-space models and structural properties. *Mathematical Systems Theory* **12** (1978) 59–72
6. Sulikowski, B., Galkowski, K., Kummert, A.: Proportional plus integral control of ladder circuits modeled in the form of two-dimensional (2D) systems. *Multidimensional Systems and Signal Processing* **26**(1) (2015) 267–290 DOI 10.1007/s11045-013-0256-1.
7. Sulikowski, B., Galkowski, K.: Control of discretised sub-class of 2D systems. In: *Proceedings of the 35th Chinese Control Conference (CCC)*, Chengdu, China (2016) 61–66
8. Busłowicz, M., Ruszewski, A.: Computer methods for stability analysis of the Roesser type model of 2D continuous-discrete linear systems. *International Journal of Applied Mathematics and Computer Science* **22**(2) (2012) 401–408
9. Arimoto, S., Kawamura, S., Miyazaki, F.: Bettering operation of robots by learning. *Journal of Robotic Systems* **1**(2) (1984) 123–140
10. Hladowski, L., Galkowski, K., Cai, Z., Rogers, E., Freeman, C.T., Lewin, P.L.: Experimentally supported 2D systems based iterative learning control law design for error convergence and performance. *Control Engineering Practice* **18**(4) (2010) 339–348
11. Paszke, E., Rogers, E., Galkowski, K., Cai, Z.: Robust finite frequency range iterative learning control design and experimental verification. *Control Engineering Practice* **21**(10) (2013) 1310–1320
12. Sornmo, O., Bernhardsson, B., Kroling, O., Gunnarsson, P., Tenhamm, H.: Frequency-domain iterative learning control of a marine vibrator. *Control Engineering Practice* **47** (2016) 70–80

13. Rogers, E., Galkowski, K., Owens, D.: Control systems theory and applications for linear repetitive processes. Lecture Notes in Control and Information Sciences, 349. Springer-Verlag, Berlin (2007)

Funding

This work is partially supported by National Science Centre in Poland, grant No. 2015/17/B/ST7/03703.

Learning filter design for ILC schemes using FIR approximation over a finite frequency range

Marcin Boski¹, Wojciech Paszke¹, Eric Rogers²

¹Institute of Control and Computation Engineering,
University of Zielona Góra, ul. Szafrana 2, 65-516 Zielona Góra, Poland.
{m.boski,w.paszke}@issi.uz.zgora.pl

²Department of Electronics and Computer Science,
University of Southampton, Southampton SO17 1BJ, UK.
etar@ecs.soton.ac.uk

Abstract. This paper addresses the design of learning filters for a class of iterative learning control (ILC) schemes. In particular, the paper develops a method for the design of finite impulse response (FIR) filters to approximate the inverse of the dynamics resulting from the feedback controller design. The filter design is linear matrix inequality (LMI) based and guarantees convergence of the ILC scheme. Also application of the generalized Kalman-Yakubovich-Popov (KYP) lemma allows the inclusion of finite frequency range performance specifications. Finally, a simulation study illustrates the effectiveness of the new design procedure.

Keywords: Iterative learning control, linear matrix inequalities, finite impulse response, learning filter.

1 Introduction

Iterative learning control (ILC) is a feedforward control scheme for improving tracking response of systems that repeat a given task or operation defined over a finite duration. Each repetition is known as a trial, or pass, and when a trial is complete, the system resets to the same initial conditions and the next trial can begin, either immediately or after a further period of time has elapsed. The advantage of this control structure is the use information from the previous trial to update the control input applied on the next trial and thereby improve performance from trial-to-trial. This feature has meant that ILC has had a significant influence on high precision control systems where already reported applications include robotic manipulators, batch processes, wafer stage motion systems and rapid thermal processing, see, e.g., [1, 2] as starting points for the literature.

The objective of ILC is to construct the control input signal such that the output tracks the reference as accurately as possible. Let y_d be the supplied reference trajectory or vector in the multiple-input multiple output case. Also discrete-time systems are considered and the notation for ILC variables is of the form $y_k(p)$, $0 \leq p \leq N - 1$ where the integer $k \geq 0$ denotes the trial number and N denotes the number of samples along the trial (N times the sampling period

gives the trial length). The error on trial k is $e_k(p) = y_d - y_k(p)$ and let $\{e_k\}_k$ denote the error sequence generated. Then the basic ILC problem is to design control action to ensure that $\{e_k\}_k$ converges in k .

Once a trial is complete in ILC, all information generated is available for use in design. One ILC law that includes previous trail information is

$$u_{k+1}(p) = u_k(p) + L e_k(p),$$

where $u_k(t)$ denotes the input on trial k , L controller acting on the previous error and in some cases this controller, also termed the learning filter, is augmented by a current trial feedback controller that stabilizes the system and suppresses unknown disturbances. The learning controller is designed to guarantee convergence in the trial domain and in many cases its design is based on the inverse of the dynamics (see the relevant references in [2]) resulting from design and application of the feedback loop.

In many cases the ideal learning filter L can not be designed since the exact plant model is not available due to the presence of modelling errors. Moreover, a fundamental problem arises if the closed loop transfer-function is strictly proper and hence its exact inverse is improper and cannot be implemented. Even if this inverse can be constructed, performance could be compromised by the presence of high frequency noise and/or non-repeating disturbances, see, e.g., [2] for further discussion and illustrative examples.

Application of the ILC scheme considered in this paper requires an implementable approximation of the L filter. This paper develops a new design method for this problem. The design is based on constructing an FIR filter to approximate the inverse of the system over a finite frequency range based on results reported in the signal processing literature [5]. The limitation on the filter bandwidth allows emphasis to be placed on a particular frequency range where L is a good approximation to the inverse of the transfer-function involved.

The remainder of this paper is organized as follows: the next section defines the design problem considered and considers trial-to-trial error convergence. Then the main result is developed in the next section and this is followed by a simulation study to illustrate the possible performance that could be achieved in application. In the last section the contributions of the paper are summarized and suggestions for possible future work are given.

The notation adopted in this paper is as follows. The null and identity matrices with compatible dimensions are denoted by 0 and I respectively. The notation $X \succ Y$ (respectively $X \prec Y$) means that the symmetric matrix $X - Y$ is positive definite (respectively negative definite). The symbol (\star) denotes block entries in symmetric matrices and $\rho(\cdot)$ and $\bar{\sigma}(\cdot)$ denote the spectral radius and maximum singular value, respectively, of their matrix arguments.

The following result, known as the generalized KYP lemma, is also used in this paper.

Lemma 1. [3] *For linear discrete time-invariant systems with transfer-function matrix $M(z)$ and frequency response matrix*

$$M(e^{j\omega}) = C(e^{j\omega}I - A)^{-1}B + D,$$

the following inequalities are equivalent

(i)

$$\begin{bmatrix} M(e^{j\omega}) \\ I \end{bmatrix}^T \Pi \begin{bmatrix} M(e^{j\omega}) \\ I \end{bmatrix} \prec 0, \quad \forall \omega \in \Theta,$$

where Π is a given real symmetric matrix and Θ denotes the following frequency ranges

	LF (low freq.)	MF (middle freq.)	HF (high freq.)
Θ	$ \omega \leq \varpi_l$	$\varpi_1 \leq \omega \leq \varpi_2$	$ \omega \geq \varpi_h$

(ii)

$$\begin{bmatrix} A & B \\ I & 0 \end{bmatrix}^T \Xi \begin{bmatrix} A & B \\ I & 0 \end{bmatrix} + \begin{bmatrix} C & D \\ 0 & I \end{bmatrix}^T \Pi \begin{bmatrix} C & D \\ 0 & I \end{bmatrix} \prec 0, \tag{1}$$

where $Q \succ 0$, P is a symmetric matrix and the matrix Ξ is specified as follows

- for the LF range

$$\Xi = \begin{bmatrix} -P & Q \\ Q & P - 2 \cos(\varpi_l)Q \end{bmatrix},$$

- for the MF range

$$\Xi = \begin{bmatrix} -P & e^{j(\varpi_1+\varpi_2)/2}Q \\ e^{-j(\varpi_1+\varpi_2)/2}Q & P - (2 \cos((\varpi_2-\varpi_1)/2))Q \end{bmatrix},$$

- and for the HF range

$$\Xi = \begin{bmatrix} -P & -Q \\ -Q & P + 2 \cos(\varpi_h)Q \end{bmatrix}.$$

2 Problem formulation

This paper considers the case when the plant dynamics to be controlled can be modeled by the following discrete linear time-invariant state-space model written in the ILC setting as

$$\begin{aligned} x_{k+1}(p+1) &= Ax_{k+1}(p) + Bu_{k+1}(p), \\ y_{k+1}(p) &= Cx_{k+1}(p), \end{aligned} \tag{2}$$

where on trial k , $x_k(p) \in \mathbb{R}^n$ is the state vector, $y_k(p) \in \mathbb{R}^m$ is the output vector and $u_k(p) \in \mathbb{R}^l$ is the control input vector.

The form of ILC considered in this paper is shown in the block diagram of Figure 1 and consists of a unity negative feedback control loop with controller C applied on the current trial k to ensure stability. The memory block in Figure 1 represents the use of previous trial information in the computation of the current trial control input and y_d denotes the supplied reference vector. In the literature

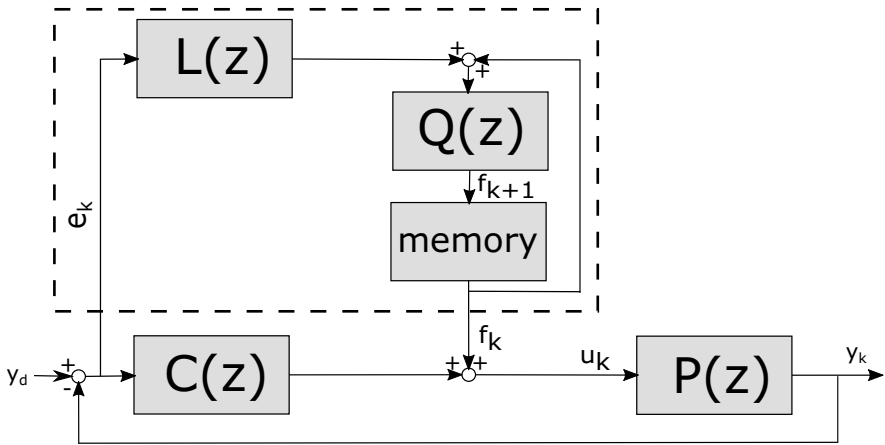


Fig. 1. ILC block diagram representation.

L is often termed the learning filter and Q the robustness filter. All computations within the dashed box of Figure 1 are completed in the time elapsed between the end of one trial and the beginning of the next, i.e., off-line.

In the block diagram of Figure 1 the ILC law is

$$F_{k+1}(z) = Q(z) (F_k(z) + L(z)E_k(z))$$

and hence the previous trial error feedforward contribution (assuming $Y_d(z) = 0$) to the current trial error is

$$E_k(z) = - [(I + G(z)C(z))^{-1}G(z)] F_k(z) = -S_P(z)F_k(z),$$

where $S_P(z) = (I + G(z)C(z))^{-1}G(z)$ denotes the sensitivity function and the propagation of the error from trial-to-trial is given by

$$E_{k+1}(z) = Q(z) (I - S_P(z)L(z)) E_k(z).$$

Introducing

$$M(z) = Q(z) (I - S_P(z)L(z)), \tag{3}$$

it follows that the condition for trial-to-trial error convergence can be formulated in \mathcal{H}_∞ control terms (see also, e.g., [2]) as

$$\|M(z)\|_\infty \triangleq \sup_{\omega \in [-\pi, \pi]} \bar{\sigma}(M(e^{j\omega})) < 1, \tag{4}$$

and minimizing $\|M(z)\|_\infty$ increases the trial-to-trial convergence speed.

Clearly, based on (3) and (4) we have that fast error convergence will occur if $L \approx S_P^{-1}$ for entire frequency range. However, if S_P is strictly proper its exact inverse is improper. Hence, we limit our attention to the frequency range where

L can be a good approximation to S_P^{-1} . The remaining frequencies should be cut-off by the Q -filter because the inverse of S_P is not sufficiently matched at higher frequencies.

The next section considers the design of $L(z)$ and $Q(z)$ with a suitable cut-off frequency.

3 Main result

Suppose that the feedback control system has been designed to give stability closed-loop and to meet any other prescribed performance requirements for this loop and hence the sensitivity function S_P of (3) has been determined. Consider also the single-input single-output (SISO) case, for simplicity, with $Q = 1$. Then the task now is to design the L -filter, i.e., compute matrices A_L , B_L , C_L and D_L that define its minimal state-space realization. Moreover, the L -filter guarantees trial-to-trial error convergence if the Nyquist plot generated by $M(z)$ of (3) lies inside the unit circle in the complex plane. If, however, some part of Nyquist plot lies outside unit circle in the complex plane then the bandwidth of the Q -filter has to be selected to cutoff the frequencies for which (4) is not satisfied. Also in many applications it is more relevant to impose performance specifications over a finite frequency range, i.e., $\omega \in [\omega_l, \omega_u]$ with $\omega_l > 0$. Then it is routine to show that (4) limited to such a frequency range is equivalent to

$$\begin{bmatrix} M(e^{j\omega}) \\ 1 \end{bmatrix}^T \begin{bmatrix} I & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} M(e^{j\omega}) \\ 1 \end{bmatrix} \prec 0, \quad \forall \omega \in [\omega_l, \omega_u], \quad (5)$$

Also choosing $\Pi = \text{diag}\{I, -1\}$ and making direct use of Lemma 1 gives that (5) applied to this case is equivalent to (1) - see [6] for more details on considering the finite frequency ranges for ILC design. The remaining problem is that (1) is not convex due to the product of the L -filter parameters and the matrices P and Q .

As one way of obtaining a convex formulation of the problem considered, assume that $L(z)$ is an n -th order FIR filter given by

$$L(z) = \alpha_0 + \alpha_1 z^{-1} + \alpha_2 z^{-2} + \dots + \alpha_n z^{-n}, \quad (6)$$

and the associated minimal state-space realization for $L(z) = C_L(zI - A_L)^{-1}B_L + D_L$ given by

$$A_L = \begin{bmatrix} 0 & I_{n-1} \\ 0 & 0 \end{bmatrix}, B_L = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C_L = [\alpha_n \ \alpha_{n-1} \ \dots \ \alpha_2 \ \alpha_1], D_L = [\alpha_0]. \quad (7)$$

In the above form, the filter parameters $\alpha_0, \alpha_1, \dots, \alpha_n$ to be designed are only present in the matrices C_L and D_L . This allows products of two matrix variables in their entries to be avoided and hence the filter parameters can be computed via a convex optimization procedure with constraints using LMIs. To establish this fact, let the known matrices A_{sp} , B_{sp} , C_{sp} and D_{sp} form a state-space model

realization of the sensitivity function $S_P(z)$. Then a state-space realization of $L(z)S_P(z)$ in (3) is given by the following state-space model (state, input, output and direct feedthrough respectively) matrices, where the filter parameters appear in \mathcal{C} and \mathcal{D} only.

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} A_{sp} & 0 \\ B_L C_{sp} & A_L \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} B_{sp} \\ B_L D_{sp} \end{bmatrix}, \\ \mathbf{C} &= [D_L C_{sp} \quad C_L], \quad \mathcal{D} = D_L D_{sp}, \end{aligned} \tag{8}$$

Suppose that the matrix Ξ of (1) is compatibly partitioned as

$$\Xi = \left[\begin{array}{c|c} \Xi_{11} & \Xi_{12} \\ \hline \Xi_{12}^T & \Xi_{22} \end{array} \right], \tag{9}$$

Then the following result gives an LMI design for the L -filter.

Theorem 2. *Consider a given system or plant to which an ILC scheme of the form shown in Figure 1 is applied. Then a stable n -order filter L of the form (6) can be designed such that the ILC convergence condition (5) holds for a chosen finite frequency range Θ of Lemma 1 if there exist matrices $P, Q \succ 0, \mathcal{C}$, and \mathcal{D} such that the following LMI is feasible*

$$\left[\begin{array}{ccc|c} \Upsilon & -\mathbf{A}^T \mathbf{P} \mathbf{B} + \Xi_{11} \mathbf{A}^T \mathbf{Q} \mathbf{B} + \Xi_{12}^T \mathbf{Q} \mathbf{B} & -\mathcal{C}^T & \\ -\mathbf{B}^T \mathbf{P} \mathbf{A} + \Xi_{11} \mathbf{B}^T \mathbf{Q} \mathbf{A} + \Xi_{12} \mathbf{B}^T \mathbf{Q} & -\mathbf{B}^T \mathbf{P} \mathbf{B} + \Xi_{11} \mathbf{B}^T \mathbf{Q} \mathbf{B} - \gamma^2 \mathbf{I} & 1 - \mathcal{D}^T & \\ -\mathcal{C} & 1 - \mathcal{D} & -1 & \end{array} \right] \prec 0, \tag{10}$$

where

$$\Upsilon = -\mathbf{A}^T \mathbf{P} \mathbf{A} + \Xi_{11} \mathbf{A}^T \mathbf{Q} \mathbf{A} + \Xi_{12} \mathbf{A}^T \mathbf{Q} + \Xi_{12}^T \mathbf{Q} \mathbf{A} + \mathbf{P} + \Xi_{22} \mathbf{Q}.$$

Moreover, the required Q -filter can be chosen as a low-pass filter with cut-off frequency equal to the highest frequency for which the above result is valid.

Proof. This is given for the low frequency (LF) range since this choice is often encountered in physical applications and the others follow by identical steps. For this range, the matrix Ξ of (1) can be partitioned as (see (9))

$$\Xi = \left[\begin{array}{c|c} \Xi_{11} & \Xi_{12} \\ \hline \Xi_{12}^T & \Xi_{22} \end{array} \right] = \left[\begin{array}{c|c} -P & Q \\ \hline Q & P - 2 \cos(\varpi_l) Q \end{array} \right]$$

and with the notation introduced in (8), the state-space representation of $M(z)$ in (3) for $Q(z) = 1$ is

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline -\mathcal{C} & 1 - \mathcal{D} \end{array} \right].$$

Direct application of Lemma 1 for above state-space model matrices give (10) and the proof is complete.

Remark 3. The Q -filter can be implemented as a zero-phase filter (e.g., by using the `filtfilt` routine in MATLAB) since such filtering is performed off-line using previous trial information and the known reference trajectory.

Remark 4. To minimize the inaccuracies between the computed L and the known S_P^{-1} , the term γ in (10) has to be minimized. This can be achieved by using the linear objective minimization procedure

$$\min_{Q>0, \mu>0} \mu$$

subject to (10) where $\mu = \gamma^2$.

4 Simulation example

In this section, a simulation example is given to illustrate the validity of the design developed in the previous section. The example chosen is the model of laboratory servomechanism system consisting of a DC motor and the inertial mass which are connected through the rigid shaft - see Figure 2 where a configuration scheme is given. The rotational speed of the mass is taken as the output

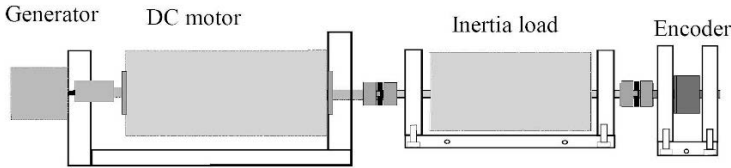


Fig. 2. Diagram configuration of the servomechanism system.

and the armature voltage as the input. Hence, the following transfer-function represents model of the controlled plant

$$G(s) = \frac{\dot{\theta}(s)}{V(s)} = \frac{K}{(Js + b)(Ls + R) + K^2}, \quad (11)$$

where $K = 0.056$ represents both the motor torque constant (K_t) and the back emf constant (K_e), $J = 0.001118$ is the total moment of inertia of the rotor and the mass, $b = 3.5077 \cdot 10^{-6}$ is the motor viscous friction constant, $L = 0.001$ is the electric inductance and $R = 2$ is the electric resistance. Hence

$$G(s) = \frac{0.056}{1.118 \cdot 10^{-6} s^2 + 0.002236s + 0.003143}. \quad (12)$$

Suppose also that the simple proportional gain is chosen as the feedback controller

$$K_p = 0.4$$

The transfer-function of the controlled system has been discretized with a sampling time of $T_s = 0.01$ secs to give a minimal discrete linear state-space model

with

$$A_{Sp} = \begin{bmatrix} 0.8735 & -0.09532 \\ 0.0625 & 0 \end{bmatrix}, B_{Sp} = \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}, C_{Sp} = [-0.4552 \quad -0.3813], D_{Sp} = [0].$$

Executing the design procedure of Theorem 2 gives the following FIR polynomial coefficients

$$\alpha_0 = 7.7126, \alpha_1 = [3.9546 \quad -11.1428]. \quad (13)$$

Given (7) and (13) a state-space model of the learning controller is given by the matrices

$$A_L = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, B_L = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C_L = [3.955 \quad -11.14], D_L = 7.713$$

Figure 3 shows the inverse frequency responses of the learning filter $L^{-1}(z)$ and $S_p(z)$ respectively. These responses are almost identical in the low frequency range up to 10 Hz. Moreover, the test system has a tendency to amplify high frequency signals (noise) and hence Q should be chosen as a low-pass filter with the desired cut-off frequency. In this example a sixth order low-pass digital Butterworth filter with cut-off frequency of 10[Hz] has been used.

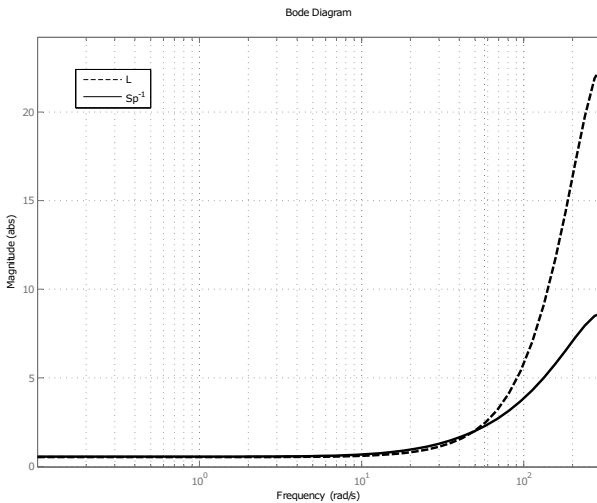


Fig. 3. Magnitude plots of L and S_p^{-1}

With the design completed, the controlled system has been simulated and after completion of each trial, the root mean square (RMS) value of the tracking

error e has been calculated with

$$\text{RMS}(e) = \sqrt{\frac{1}{N} \sum_{p=1}^N e(p)^2},$$

where $N = 1200$ is the number of data samples over the trial length. The resulting convergence of the tracking error with k is clear from Figure 4. Figure 5

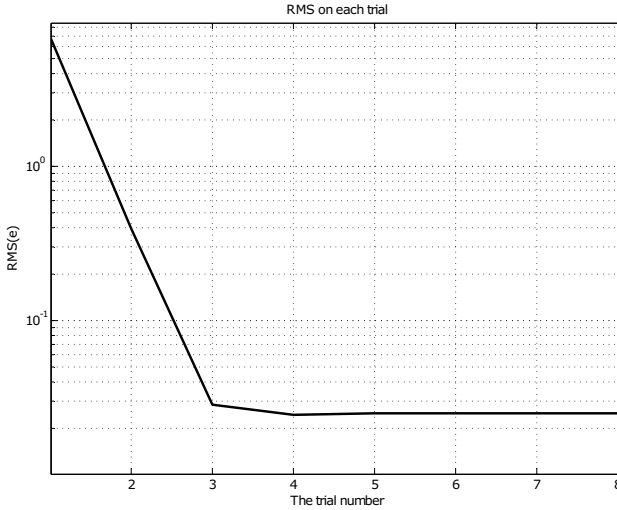


Fig. 4. The tracking error convergence.

shows the controlled system response for $k = 1$ and $k = 2$ together with the reference trajectory, which is the solid line. These results exhibit the fast trial-to-trial error convergence.

5 Conclusions

This paper has developed a systematic procedure for designing the learning filter in a commonly used ILC scheme. The procedure has been established by use of signal processing theory in the form of algorithms for designing FIR filters. A major advantage of the new design is that it enables relatively easy filters synthesis over a finite frequency range of interest where the computations are LMI based. Additionally, the bandwidth of the robustness filter Q can be easily assigned as the highest frequency for which the main result is valid. Finally, the

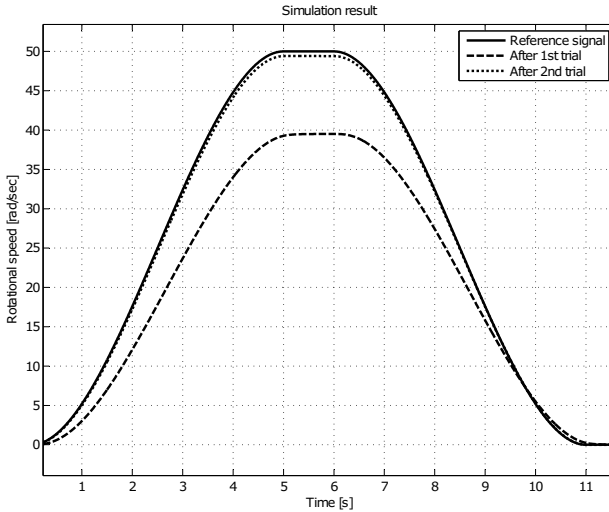


Fig. 5. The controlled system response at 1st and 2nd trial.

theoretical findings have been illustrated by simulation results from a servomechanism system model. Future developments will include experimental verification and multiple input multiple output systems.

Acknowledgments. This work is partially supported by National Science Centre in Poland, grant No. 2014/15/B/ST7/03208.

References

- [1] Ahn, H.S., Chen, Y.Q., and Moore, K.L. (2007), "Iterative learning control: brief survey and categorization," *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, vol. 37, no 6, pp. 1109–1121.
- [2] Bristow, D.A., Tharayil, M., and Alleyne, A. (2006), "A survey of Iterative Learning Control," *IEEE Control Systems Magazine*, vol. 26, no 3, pp. 96–114.
- [3] Iwasaki, T., and Hara, S. (2005), "Generalized KYP lemma: unified frequency domain inequalities with design applications," *IEEE Transactions on Automatic Control*, vol. 50, no 1, pp. 41–59.
- [4] Boski, M. and Paszke, W. (2015), "Application of a repetitive process setting to design of monotonically convergent iterative learning control," *Journal of Physics: Conference Series*, vol.659.
- [5] Nagahara, M. (2011), "Min-Max Design of FIR Digital Filters by Semidefinite Programming," *Applications of Digital Signal Processing*, , pp. 193–210.
- [6] Paszke, W., Rogers, E. and Gałkowski, K. (2016), "Experimentally verified generalized KYP lemma based iterative learning control design," *Control Engineering Practice*, vol. 53, pp. 57–67.

An example of adaptive fuzzy control design with the use of frequency-domain methods

Krzysztof Wiktorowicz

Rzeszow University of Technology,
Department of Computer and Control Engineering,
Rzeszów, Poland
kwiktor@prz.edu.pl

Abstract. This paper presents an example of adaptive fuzzy control design for an unstable plant with a pole changing location in the right half-plane. The design procedure utilizes the Nyquist and the circle stability criteria that can be graphically tested using Nyquist plots. It is assumed that the function of the fuzzy controller is a nonlinearity described by a sector condition. An adaptation mechanism ensures that during adaptation this function stays in a safe sector.

Keywords: adaptive control, fuzzy control, nonlinear control systems, stability

1 Introduction

The design of adaptive fuzzy control (AFC) is carried out mainly using the Lyapunov method. From the literature review given in [23] and recent advances in this area [1, 2, 4–15, 17, 19–21, 24–32] it can be seen that frequency-domain methods have not been applied in the design of AFC systems. These methods have only been used for the stability analysis of nonadaptive (time-invariant) fuzzy controllers.

The frequency methods offer simple graphical interpretation on the Nyquist plane. They are independent of system order and applicable to plants with irrational transfer functions. Moreover, these methods can be understood and applied by engineers because they do not require knowledge of advanced mathematics.

Therefore, research has been undertaken to propose new solutions using frequency-domain methods in the area of adaptive fuzzy control [23], [22].

This paper considers the control of linear dynamic systems using an adaptive fuzzy controller with a single-input and single-output structure. The controller operates in the direct adaptation mode with the output-feedback and with a linear or nonlinear reference model, which determines the desired response of the closed-loop system. It is assumed that the controller function is a time-varying nonlinearity that belongs to some bounded sector. The adaptation mechanism guarantees that the controller function remains in a safe sector.

The proposed method was applied in the paper [23] for an unstable plant with a pole changing location in the left half-plane. In this paper, we consider a similar problem but with a pole changing location in the right half-plane.

2 Problem formulation

We consider an adaptive system (see Fig. 1) with the *plant*, which is the time-invariant linear dynamic element and the *fuzzy controller*, which is the time-varying nonlinear static element. Other elements are the *adaptation mechanism*, which modifies the rule base of the controller and the nonlinear *reference model*, which characterizes the desired response of the system. In the system we distinguish: $r(t)$ — reference signal, $y(t)$ — output of the closed-loop system, $e(t) = r(t) - y(t)$ — control error, $u(t)$ — output of the controller, $y_m(t)$ — output of the reference model and $\varepsilon(t) = y(t) - y_m(t)$ — adaptation error.

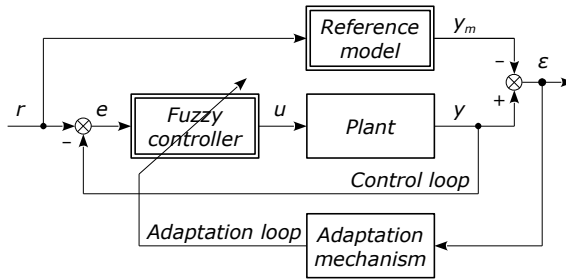


Fig. 1. Block diagram of the direct model reference fuzzy adaptive controller

2.1 Stability criteria

We assume that: 1) the plant is described by the transfer function $G(s) = \frac{B(s)}{A(s)}e^{-\tau s}$, 2) $A(s)$ may have unstable roots, 3) the unstable system is stabilizable by the linear feedback $u(t) = re(t)$. To assess the stability of the system we use the Nyquist and the circle theorems that are based on the analysis of Nyquist plots. These theorems for stable and unstable plants are recalled below.

Theorem 1 ([3]). *The closed-loop system with the proportional controller $u(t) = re(t)$ and a stable plant is stable if and only if the Nyquist plot $G(j\omega)$ does not encircle the point $(-1, j0)$.*

If the plant is unstable, the Nyquist criterion requires knowledge of the number of P poles in the right half-plane.

Theorem 2 ([3]). *The closed-loop system with the proportional controller $u(t) = re(t)$ and an unstable plant is stable if and only if the Nyquist plot $G(j\omega)$ encircles P times the point $(-1, j0)$ counterclockwise.*

The circle criterion applies for systems with a linear dynamic element and a nonlinear static element whose function $u(t) = f(e, t)$ is a sector-bounded nonlinearity satisfying the condition $0 \leq f(e, t)/e \leq k, f(0, t) = 0$. In this case we also say that the controller function lies in the sector $[0, k]$. If the nonlinearity lies in the sector $[\beta, \beta + k]$ we use the transformation $u_1(t) = f_1(e, t) = f(e, t) - \beta e(t)$ where the function f_1 lies in the sector $[0, k]$. This transformation can be applied for unstable systems if such β exists that the transfer function

$$G_1(s) = \frac{G(s)}{1 + \beta G(s)} \tag{1}$$

is stable.

Theorem 3 ([18]). *A nonlinear system with the stable transfer function $G(s)$ and nonlinearity bounded in the sector $[0, k]$ is absolutely stable if the following condition is met:*

$$\inf_{\omega} \operatorname{Re} [G(j\omega)] + \frac{1}{k} > 0. \tag{2}$$

If the nonlinearity $f(e, t)$ lies in the sector $[\beta, \beta + k]$, then the circle criterion is used for the transformed transfer function (1).

2.2 Time-varying Takagi-Sugeno fuzzy controller

We consider the Takagi-Sugeno fuzzy controller [16] with the input $e(t)$ and the output $u(t)$, which is described by the nonlinear time-varying function $u(t) = f[e(t), t]$. For the input e , we define r triangular fuzzy sets with the peaks in p_i :

$$A_i(e; p_{i-1}, p_i, p_{i+1}) = \max \left(0, \min \left(\frac{e - p_{i-1}}{p_i - p_{i-1}}, \frac{p_{i+1} - e}{p_{i+1} - p_i} \right) \right) \tag{3}$$

where p_{i-1}, p_i, p_{i+1} are the parameters of the membership function, and $p_{i-1} < p_i < p_{i+1}$. It is assumed that the number of sets A_i is odd ($r \in \{3, 5, 7, \dots\}$) and the sum of the membership grades for any argument e is equal to unity.

The function of the controller is described by r fuzzy control rules in the form of

$$R_i : \text{IF } e(t) \in A_i, \text{ THEN } u(t) = c_i(t) \tag{4}$$

where $c_i(t) \in \mathbb{R}$ is the consequent of the rule R_i . The rules (4) are written as the following table:

$e(t)$	A_1	A_2	\dots	A_{i_0}	\dots	A_{r-1}	A_r
$u(t)$	c_1	c_2	\dots	c_{i_0}	\dots	c_{r-1}	c_r

(5)

The index $i_0 = (r + 1)/2$ denotes the set for which the controller output takes the value 0, which means that for A_{i_0} we have $p_{i_0} = 0$ and $c_{i_0} = 0$. This results from the sector condition where $f(0, t) = 0$.

The output of the controller is determined by

$$u(t) = \sum_{i=1}^r \xi_i(e(t))c_i(t) \tag{6}$$

where

$$\xi_i(e(t)) = \frac{A_i(e(t))}{\sum_{i=1}^r A_i(e(t))} \tag{7}$$

is the *fuzzy basis function*.

In the paper [23] it was proven that for

$$c_1, \dots, c_{i_0-1} \leq 0, \quad c_{i_0} = 0, \quad c_{i_0+1}, \dots, c_r \geq 0, \tag{8}$$

the controller function lies in the sector $[\beta, \beta + k]$ where

$$\beta = \min_{\substack{i=1, \dots, r \\ i \neq i_0}} (c_i/p_i), \quad \beta + k = \max_{\substack{i=1, \dots, r \\ i \neq i_0}} (c_i/p_i) \tag{9}$$

Using this property, the adaptation mechanism was proposed, in which the consequents c_i are adapted using the formula $c_i(t + 1) = c_i(t) - \gamma \varepsilon \xi_i$, where γ is the adaptation gain. The adaptation is carried out in such a way that the consequents lie in the safe sector $[\beta, \beta + k]$ determined by the circle criterion.

3 Example: Unstable plant with a pole changing location in the right half-plane

We consider an unstable third order plant described by the transfer function

$$G(s, a) = \frac{s + 1}{s(s - a)(0.2s + 1)} e^{-0.1s} \tag{10}$$

where $a > 0$. The transfer function (10) has one stable pole $p_1 = -5$, two unstable poles $p_2 = 0, p_3 = a$ and zero $z = -1$. The goal is to construct a sector bounded adaptive fuzzy controller which will be stable for $a \in [0.4, 0.6]$. We assume that $a_n = 0.6$ is the nominal value of a .

Step 1. Determining the common Hurwitz sector

The Hurwitz sector (HS) is a set of gains of the linear controller determined for a certain value of a for which the closed-loop system is stable. The common Hurwitz sector (CHS) is the intersection of sectors HS determined for $a \in [0.4, 0.6]$.

The Nyquist plots $G(j\omega, a)$ for $\omega \in [0.8, 60]$ rad/s are presented in Fig. 2. Because the plant has one pole in the right half-plane, the system is stable if

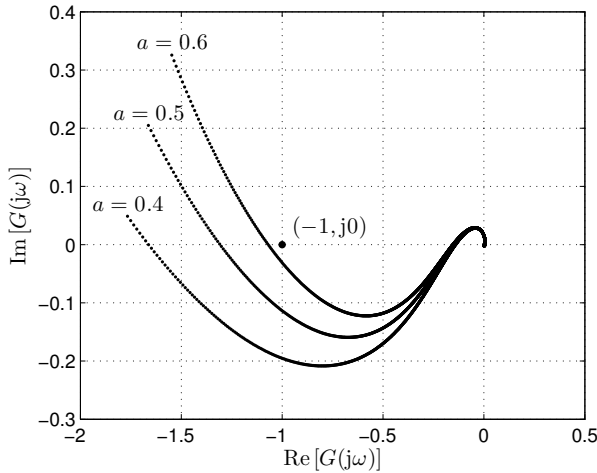


Fig. 2. Nyquist plots $G(j\omega, a)$ of the plant (10) for $a = 0.4, 0.5, 0.6$

the Nyquist plot encircles one time the point $(-1, j0)$ counter-clockwise (see Theorem 2). In order to determine the Hurwitz sector we have to calculate two critical gains r_{c_1} and r_{c_2} for the sector (r_{c_1}, r_{c_2}) (see Fig. 3).

For example, for $a = 0.5$ the linear system is stable in the sector $HS = (0.76, 6.80)$. It can be seen in Fig. 3 that for a changing in the interval $[0.4, 0.6]$ the linear system is stable in the sector

$$CHS = (0.94, 6.50). \tag{11}$$

Step 2. Determining the common circle sector

The circle sector (CS) is the sector determined by the circle criterion for a certain value of a and the common circle sector (CCS) is the intersection of sectors CS determined for $a \in [0.4, 0.6]$.

Taking into account the result from the previous step we select $\beta \in CHS = (0.94, 6.50)$. For example, for $\beta = 2.2$ we obtain the transfer function (1) of the form

$$G_1(s, a) = \frac{G(s, a)}{1 + \beta G(s, a)} = \frac{(s + 1)e^{-0.1s}}{s(s - a)(0.2s + 1) + 2.2(s + 1)e^{-0.1s}}. \tag{12}$$

The Nyquist plots $G_1(j\omega, a)$ for $a = 0.4, 0.5, 0.6$ and $\omega \in [0.2, 20]$ rad/s are presented in Fig. 4. Using Theorem 3 we obtain the sectors CS presented in Fig. 3. For example, for $a = 0.5$ the nonlinear system is stable in the sector $CS = [2.2, 3.78]$. For $a \in [0.4, 0.6]$ and $\beta = 2.2$ the adaptive system is stable in the sector

$$CCS = [2.2, 3.55]. \tag{13}$$

Selecting another value of β we can obtain other sectors CS and CCS.

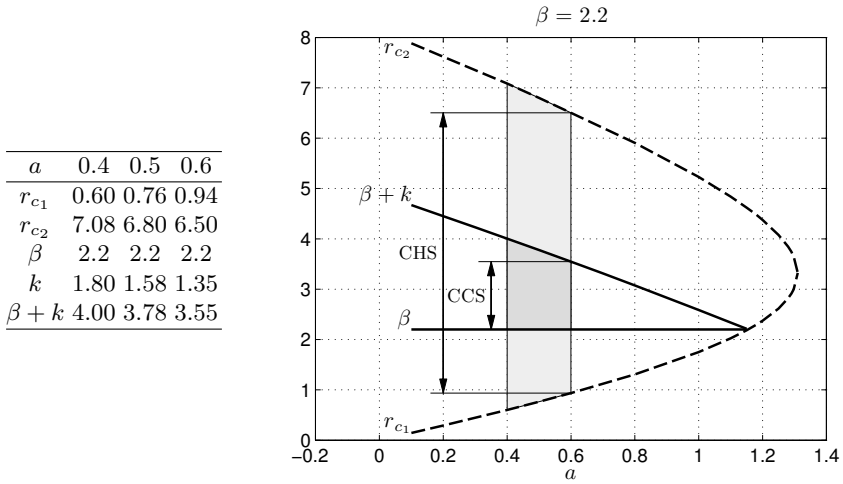


Fig. 3. Plots of the Hurwitz sector (r_{c1}, r_{c2}) and the circle sector $[\beta, \beta + k]$ as a function of the parameter a with $\beta = 2.2$ for the transfer function (10); common sectors are: CHS = (0.94, 6.50), CCS = [2.2, 3.55]

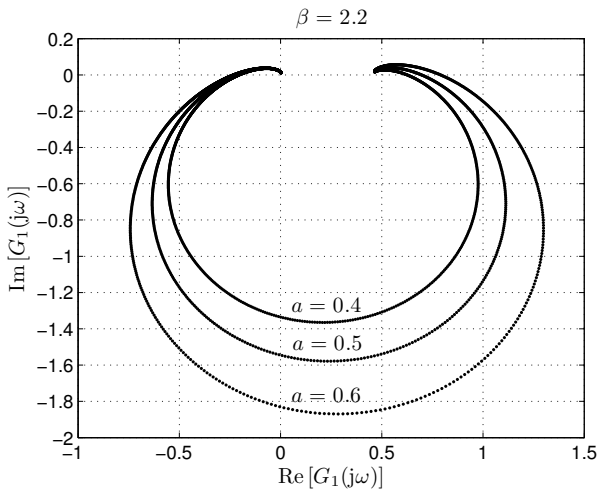


Fig. 4. Nyquist plots $G_1(j\omega, a)$ of the transformed transfer function (12) for $\beta = 2.2$ and $a = 0.4, 0.5, 0.6$

Step 3. Choosing the reference model

In the example presented in [23] a nonlinear reference model was used. It was designed by taking into account the settling time. In this example, we use a linear reference model in which $u_m(t) = r_m e_m(t)$ and

$$G_m(s) = G(s, a_n) = \frac{s + 1}{s(s - 0.6)(0.2s + 1)} e^{-0.1s} \tag{14}$$

As a quality criterion we apply the integral of squared error $ISE = \int_0^\infty e_m^2(t) dt$. Based on numerical experiments, the value $r_m = 3.4$ was chosen, for which the ISE criterion has the smallest value equal to $ISE(r_m) = 1.18$. Because $r_m \in CHS$, then the linear reference model is stable.

Step 4. Setting the parameters of the controller

In the considered example we define five triangular fuzzy sets with the peaks in $p_i = -1.2, -0.6, 0, 0.6, 1.2$. The consequents c_i of the fuzzy controller rules are set initially to random values in the CCS sector.

Step 5. Simulations of the system

In Fig. 5 and Fig. 6, the simulation results for $a = 0.45$ and the reference signal $r(t) = 20 \sin(0.04t)$ are shown. The adaptation gain was equal to $\gamma = 0.8$ and the dead zone $\delta = 0.002$ was used for the adaptation error. The adaptation mechanism was activated at time $t = 200$ s. It can be seen that the relation $f(e, t)/e$ from the sector condition does not exceed the safe sector $CCS = [2.2, 3.55]$, and thus the adaptive fuzzy system is stable. As a result of the adaptation, the fuzzy control rules were obtained in the form of

$$\begin{array}{c} \hline e \quad A_1 \quad A_2 \quad A_3 \quad A_4 \quad A_5 \\ \hline u \quad -3.56 \quad -1.53 \quad 0 \quad 1.55 \quad 2.72 \\ \hline \end{array} \tag{15}$$

Using (9) it can be checked that the obtained controller function lies in the sector $[2.27, 2.97] \subset CCS$.

4 Conclusions

This paper presented an example of adaptive fuzzy controller design with the use of frequency-domain methods. In the example, an unstable plant with a pole changing location in the right half-plane was considered. The design method is based on the Nyquist and the circle theorems that are graphically tested using Nyquist plots. In this method, it is assumed that the fuzzy controller function is a sector-bounded nonlinearity. The consequents of the fuzzy rules are checked during adaptation so this function stays in a safe sector.

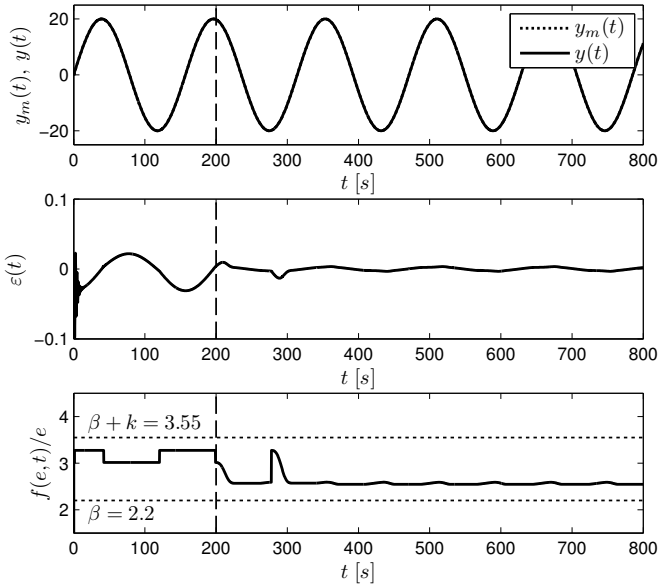


Fig. 5. System responses $y_m(t)$, $y(t)$, adaptation error $\varepsilon(t)$ and relation $f(e,t)/e$ for $a = 0.45$ and sector $\text{CCS} = [\beta, \beta + k] = [2.2, 3.55]$

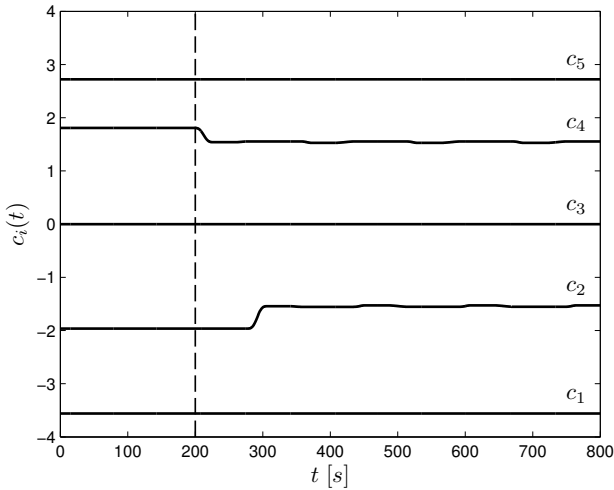


Fig. 6. Consequents c_i of the fuzzy controller rules as a function of time; initial values $c_i(t = 0)$ were set randomly in the sector CCS

References

1. Chen, C.L.P., Ren, C.E., Du, T.: Fuzzy observed-based adaptive consensus tracking control for second-order multiagent systems with heterogeneous nonlinear dynamics. *IEEE Transactions on Fuzzy Systems* **24**(4), 906–915 (2016)
2. Er, M.J., Mandal, S.: A survey of adaptive fuzzy controllers: Nonlinearities and classifications. *IEEE Transactions on Fuzzy Systems* **24**(5), 1095–1107 (2016)
3. Franklin, G., Powell, J., Emami-Naeini, A.: *Feedback control of dynamic systems*. Pearson (2010)
4. Gao, Y., Tong, S., Li, Y.: Observer-based adaptive fuzzy output constrained control for MIMO nonlinear systems with unknown control directions. *Fuzzy Sets and Systems* **290**, 79–99 (2016)
5. Hwang, C.L., Fang, W.L.: Global fuzzy adaptive hierarchical path tracking control of a mobile robot with experimental validation. *IEEE Transactions on Fuzzy Systems* **24**(3), 724–740 (2016)
6. Lai, G., Liu, Z., Zhang, Y., Chen, C.L.P.: Adaptive fuzzy tracking control of nonlinear systems with asymmetric actuator backlash based on a new smooth inverse. *IEEE Transactions on Cybernetics* **46**(6), 1250–1262 (2016)
7. Li, P., Li, P., Sui, Y.: Adaptive fuzzy hysteresis internal model tracking control of piezoelectric actuators with nanoscale application. *IEEE Transactions on Fuzzy Systems* **24**(5), 1246–1254 (2016)
8. Li, Y., Tong, S., Li, T.: Hybrid fuzzy adaptive output feedback control design for uncertain MIMO nonlinear systems with time-varying delays and input saturation. *IEEE Transactions on Fuzzy Systems* **24**(4), 841–853 (2016)
9. Li, Y.X., Yang, G.H.: Fuzzy adaptive output feedback fault-tolerant tracking control of a class of uncertain nonlinear systems with nonaffine nonlinear faults. *IEEE Transactions on Fuzzy Systems* **24**(1), 223–234 (2016)
10. Liu, Y.J., Tong, S., Li, D.J., Gao, Y.: Fuzzy adaptive control with state observer for a class of nonlinear discrete-time systems with input constraint. *IEEE Transactions on Fuzzy Systems* **24**(5), 1147–1158 (2016)
11. Liu, Z., Wang, F., Zhang, Y., Chen, C.L.P.: Fuzzy adaptive quantized control for a class of stochastic nonlinear uncertain systems. *IEEE Transactions on Cybernetics* **46**(2), 524–534 (2016)
12. Long, L., Zhao, J.: Adaptive fuzzy output-feedback dynamic surface control of MIMO switched nonlinear systems with unknown gain signs. *Fuzzy Sets and Systems* **302**, 27–51 (2016)
13. Ren, C.E., Chen, L., Chen, C.L.P.: Adaptive fuzzy leader-following consensus control for stochastic multiagent systems with heterogeneous nonlinear dynamics. *IEEE Transactions on Fuzzy Systems* **25**(1), 181–190 (2017)
14. Rigatos, G., Zhu, G., Yousef, H., Boulkroune, A.: Flatness-based adaptive fuzzy control of electrostatically actuated MEMS using output feedback. *Fuzzy Sets and Systems* **290**, 138–157 (2016)
15. Sui, S., Tong, S.: Fuzzy adaptive quantized output feedback tracking control for switched nonlinear systems with input quantization. *Fuzzy Sets and Systems* **290**, 56–78 (2016)
16. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics* (1), 116–132 (1985)
17. Tong, S., Li, Y., Sui, S.: Adaptive fuzzy output feedback control for switched nonstrict-feedback nonlinear systems with input nonlinearities. *IEEE Transactions on Fuzzy Systems* **24**(6), 1426–1440 (2016)

18. Vidyasagar, M.: *Nonlinear Systems Analysis*. Prentice-Hall Networks Series. Prentice-Hall (1978)
19. Wang, H., Liu, X., Liu, K.: Adaptive fuzzy tracking control for a class of pure-feedback stochastic nonlinear systems with non-lower triangular structure. *Fuzzy Sets and Systems* **302**, 101–120 (2016)
20. Wang, N., Er, M.J., Sun, J.C., Liu, Y.C.: Adaptive robust online constructive fuzzy control of a complex surface vehicle system. *IEEE Transactions on Cybernetics* **46**(7), 1511–1523 (2016)
21. Wang, T., Qiu, J., Yin, S., Gao, H., Fan, J., Chai, T.: Performance-based adaptive fuzzy tracking control for networked industrial processes. *IEEE Transactions on Cybernetics* **46**(8), 1760–1770 (2016)
22. Wiktorowicz, K.: Design of state feedback adaptive fuzzy controllers for second order-systems using a frequency stability criterion. *IEEE Transactions on Fuzzy Systems* (2016). DOI 10.1109/TFUZZ.2016.2566671
23. Wiktorowicz, K.: Output feedback direct adaptive fuzzy controller based on frequency-domain methods. *IEEE Transactions on Fuzzy Systems* **24**(3), 622–634 (2016)
24. Wu, L.B., Yang, G.H.: Adaptive fuzzy tracking control for a class of uncertain nonaffine nonlinear systems with dead-zone inputs. *Fuzzy Sets and Systems* **290**, 1–21 (2016)
25. Wu, T.S., Karkoub, M., Yu, W.S., Chen, C.T., Her, M.G., Wu, K.W.: Anti-sway tracking control of tower cranes with delayed uncertainty using a robust adaptive fuzzy control. *Fuzzy Sets and Systems* **290**, 118–137 (2016)
26. Yin, S., Shi, P., Yang, H.: Adaptive fuzzy control of strict-feedback nonlinear time-delay systems with unmodeled dynamics. *IEEE Transactions on Cybernetics* **46**(8), 1926–1938 (2016)
27. Yue, M., An, C., Du, Y., Sun, J.: Indirect adaptive fuzzy control for a nonholonomic/underactuated wheeled inverted pendulum vehicle based on a data-driven trajectory planner. *Fuzzy Sets and Systems* **290**, 158–177 (2016)
28. Zhai, D., An, L., Li, J., Zhang, Q.: Adaptive fuzzy fault-tolerant control with guaranteed tracking performance for nonlinear strict-feedback systems. *Fuzzy Sets and Systems* **302**, 82–100 (2016)
29. Zhai, D.H., Xia, Y.: Adaptive fuzzy control of multilateral asymmetric teleoperation for coordinated multiple mobile manipulators. *IEEE Transactions on Fuzzy Systems* **24**(1), 57–70 (2016)
30. Zhang, L., Li, Y., Tong, S.: Adaptive fuzzy output feedback control for MIMO switched nonlinear systems with prescribed performances. *Fuzzy Sets and Systems* **306**, 153–168 (2017)
31. Zhang, Y., Tao, G., Chen, M., Wen, L.: Parameterization and adaptive control of multivariable noncanonical T-S fuzzy systems. *IEEE Transactions on Fuzzy Systems* **25**(1), 156–171 (2017)
32. Zhao, X., Shi, P., Zheng, X.: Fuzzy adaptive control design and discretization for a class of nonlinear uncertain systems. *IEEE Transactions on Cybernetics* **46**(6), 1476–1483 (2016)

Detection of atypical elements with fuzzy and intuitionistic fuzzy evaluations

Piotr Kulczycki^{1,2} and Damian Kruszewski²

¹AGH University of Science and Technology,
Faculty of Physics and Applied Computer Science, Division for Information
Technology and Systems Research

²Polish Academy of Sciences, Systems Research Institute, Centre of Information
Technology for Data Analysis Methods

kulczycki@agh.edu.pl; kulczycki@ibspan.waw.pl

Abstract. The task for detection of atypical elements is one of the fundamental tasks of contemporary data analysis, finding applications in numerous problems in practically all areas of sciences and engineering. As an example, in the classic approach of automatic control, e.g. fault detection problems, the appearance of an unusual value of a vector describing a system's technical state may testify to the occurrence of a malfunction. This paper presents a procedure for the detection of atypical elements, understood in the sense that they happen rarely. Particularly, in the case of multimodal distributions with more distant factors, such an approach allows atypical elements to be located not only in peripheral regions, but also potentially inside, between modes. The outcome indicating whether an examined observation should be classed as atypical is defined here in fuzzy and intuitionistic fuzzy forms.

Keywords: rare element, atypical element, outlier, fuzzy evaluation, intuitionistic fuzzy evaluation.

1. Introduction

Imagine a single number, or vector of quantities characterizing the technical state of a system. Assume we have a representative sample of its values. If the subsequent tested element seems to be atypical, it most often proves the appearance of some anomaly. Depending on the type of problem, it can be for example a malfunction

(fault) of a supervised device or an error in information processing. In medical tasks a similar situation may point to a condition of illness or pathology, in marketing that an examined object is uncommon and so should be treated differently, in banking it can signal a fraud attempt, while in sociology it indicates the arrival of a new, unusual trend.

There is no one universal definition of atypical elements [Aggarwal, 2013; Barnett and Lewis, 1994]. In the most popular, distance-based approach it is considered that they are "outliers" – elements lying far from the others. This paper will apply the frequency approach, whereby atypical elements are rare, i.e. the probability of their appearance is faint. Thus, we can discover atypical observations not only on the peripheries of a data set, but in the case of multimodal distributions with wide-spreading segments, also those lying in between these segments, even if close to the center of the population. An evaluation of whether the tested element should be termed atypical will be given in the fuzzy [Kacprzyk, 1986; Klir and Yuan, 1995] and intuitionistic fuzzy [Atanassov, 1999; Szmidt, 2014] forms. The investigated procedure is designed on the basis of the nonparametric kernel estimators method [Kulczycki, 2005; Wand and Jones, 1995], which frees it from a distribution characterizing the data set under consideration. Its broader description can be found in the paper [Kulczycki, Kruszewski; 2017], currently in press. Here can be found a comprehensive set of formulas for direct application, without laborious research or literary study.

The structure of this paper is as follows. Section 2 presents the statistical kernel estimators methodology. Then, the basic formula of the procedure for detection of atypical elements is described in Section 3. The quality of this procedure is considerably improved in Section 4 by significantly increasing the set of representative elements. Next, Section 5 provides formulas for fuzzy and intuitionistic fuzzy evaluations. The results obtained in this way will be illustrated in the final Section 6.

2. Nonparametric Kernel Estimators

In the presented method, the characteristics of a data set will be defined using the methodology of kernel estimators (also called Parzen or Rosenblatt estimators). It is distribution-free, i.e. the preliminary assumptions concerning the types of appearing distributions are not required. A broad description can be found in the monographs

[Kulczycki, 2005; Wand and Jones, 1994]. Exemplary applications for data analysis tasks are described in the publications [Kulczycki and Charytanowicz, 2010, 2013; Kulczycki and Daniel, 2009; Kulczycki and Kowalski, 2016; Kulczycki and Waglowski, 2005]; see also [Kulczycki and Lukasik, 2014; Kulczycki et al, 2017].

Let the n -dimensional continuous random variable X be given, with a distribution characterized by the density f . Its kernel estimator $\hat{f}: \mathbb{R}^n \rightarrow [0, \infty)$, calculated using the experimentally obtained m -element random sample x_i for $i = 1, 2, \dots, m$, in its basic form is defined as

$$\hat{f}(x) = \frac{1}{mh^n} \sum_{i=1}^m K\left(\frac{x - x_i}{h}\right), \quad (1)$$

where $m \in \mathbb{N} \setminus \{0\}$, the coefficient $h > 0$ is called a smoothing parameter, while the measurable function $K: \mathbb{R}^n \rightarrow [0, \infty)$ of unit integral $\int_{\mathbb{R}^n} K(x) dx = 1$, symmetrical with respect to zero and having a weak global maximum in this place, takes the name of a kernel. The choice of form of the kernel K and the calculation of the smoothing parameter h value is made most often with the criterion of the mean integrated square error.

Thus, the choice of the kernel form has – from a statistical point of view – no practical meaning and thanks to this, it becomes possible to take into account primarily properties of the estimator obtained or computational aspects, advantageous from the point of view of the applicational problem under investigation; for broader discussion see the books [Kulczycki, 2005 – Section 3.1.3; Wand and Jones, 1994 – Sections 2.7 and 4.5]. In the one-dimensional case (i.e. when $n = 1$) the normal (Gauss) kernel

$$K_j(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) \quad (2)$$

and the uniform kernel

$$K_j(x) = \begin{cases} \frac{1}{2} & \text{for } x \in [-1, 1] \\ 0 & \text{for } x \notin [-1, 1] \end{cases} \quad (3)$$

will be used in the following. In the multidimensional case, a so-called product kernel will be applied hereinafter. The main idea here is the division of particular

variables with the multidimensional kernel then becoming a product of n one-dimensional kernels for particular coordinates. Thus kernel estimator (2) is then given as

$$\hat{f}(x) = \frac{1}{mh_1h_2\dots h_n} \sum_{i=1}^m K_1\left(\frac{x_1 - x_{i,1}}{h_1}\right) K_2\left(\frac{x_2 - x_{i,2}}{h_2}\right) \dots K_n\left(\frac{x_n - x_{i,n}}{h_n}\right), \quad (4)$$

where K_j ($j=1, 2, \dots, n$) denote one-dimensional kernels, e.g. (2) or (3), h_j ($j=1, 2, \dots, n$) are smoothing parameters individualized for particular coordinates, while assigning to coordinates

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad \text{and} \quad x_i = \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,n} \end{bmatrix} \quad \text{for } i=1, 2, \dots, m. \quad (5)$$

3. Basic Version of Procedure

The basic idea of the presented procedure for detection of atypical elements stems from the significance test proposed in the work [Kulczycki and Prochot, 2002]. Let the set be given, with elements representative for the population

$$x_1, x_2, \dots, x_m. \quad (6)$$

Treat these elements as realizations of the n -dimensional continuous random variable X with distribution having density f and calculate – in accordance with Section 2 (using a normal kernel) – the kernel estimator \hat{f} . Next consider the set of its value for elements of set (6), so

$$\hat{f}(x_1), \hat{f}(x_2), \dots, \hat{f}(x_m). \quad (7)$$

Particular values $\hat{f}(x_i)$ characterize the probability of occurrence of the element x_i , therefore the lower the value $\hat{f}(x_i)$, the more the element x_i can be interpreted as "less typical", or rather happening more rarely.

Define now the number

$$r \in (0,1) \quad (8)$$

establishing sensitivity of the procedure for atypical elements detection. This number will determine the assumed proportion of atypical elements in relation to the total population, and therefore the ratio of the number of atypical to the sum of atypical and typical elements. In practice

$$r = 0.01, 0.05, 0.1 \quad (9)$$

is the most often used, with particular attention paid to the second option.

Let us treat set (7) as realizations of a real (one-dimensional) random variable and calculate the estimator for the quantile of the order r . The positional estimator of the second order [Parrish, 1990; Kulczycki, 1998] will be applied in the following, given by the formula

$$\hat{q}_r = \begin{cases} z_i & \text{for } mr \leq 0.5 \\ (0.5 + i - mr)z_i + (0.5 - i + mr)z_{i+1} & \text{for } 0.5 < mr < m - 0.5 \\ z_m & \text{for } mr \geq m - 0.5, \end{cases} \quad (10)$$

where $i = [mr + 0.5]$, while $[d]$ denotes an integral part of the number $d \in \mathbb{R}$, and z_i is the i -th value in size of set (7) after its sorting, thus

$$\{z_1, z_2, \dots, z_m\} = \{\hat{f}(x_1), \hat{f}(x_2), \dots, \hat{f}(x_m)\} \quad (11)$$

with $z_1 \leq z_2 \leq \dots \leq z_m$.

Finally, if for a given tested element $\tilde{x} \in \mathbb{R}^n$, the condition $\hat{f}(\tilde{x}) \leq \hat{q}_r$ is fulfilled, then this element should be considered atypical; for the opposite $\hat{f}(\tilde{x}) > \hat{q}_r$ it is typical.

The above procedure for atypical elements detection, combined with the properties of kernel estimators, allows in the multidimensional case for inferences based not only on values for specific coordinates of a tested element, but above all on the relations between them.

4. Extended Pattern

Although, from a theoretical point of view, the procedure presented in the previous section seems complete, when the values r are applied in practice – see condition (9) – and the size m is not big, the estimator of the quantile \hat{q}_r is encumbered with

a large error, due to the low number of elements z_i smaller than the estimated value. To counteract this, a data set will be extended by generating additional elements with distribution identical to that characterizing the subject population, based on set (6).

The methodology for enlarging a set representative for the investigated population is suggested using von Neumann's elimination concept [Gentle, 2003]. This allows the generation of a sequence of random numbers of distribution with support bounded to the interval $[a, b]$, while $a < b$, characterized by the density f of values limited by the positive number c , i.e.

$$f(x) \leq c \quad \text{for every } x \in [a, b]. \tag{12}$$

In the multidimensional case, the interval $[a, b]$ generalizes to the n -dimensional cuboid $[a_1, b_1] \times [a_2, b_2] \times \dots \times [a_n, b_n]$, while $a_j < b_j$ for $j = 1, 2, \dots, n$.

First the one-dimensional case is considered. Let us generate two pseudorandom numbers u and v of distribution uniform to the intervals $[a, b]$ and $[0, c]$, respectively. Next one should check that

$$v \leq f(u). \tag{13}$$

If the above condition is fulfilled, then the value u ought to be assumed as the desired realization of a random variable with distribution characterized by the density f , that is

$$x = u. \tag{14}$$

In the opposite case the numbers u and v need to be removed and steps (13)-(14) repeated, until the desired number of pseudorandom numbers x with density f is obtained.

In the presented procedure the density f is established by the kernel estimators methodology, described in Section 2. Denote its estimator as \hat{f} . The uniform kernel will be employed, allowing easy calculation of the support boundaries a and b , as well as the parameter c appearing in condition (12). Namely:

$$a = \min_{i=1,2,\dots,m} x_i - h \tag{15}$$

$$b = \max_{i=1,2,\dots,m} x_i + h \tag{16}$$

and

$$c = \max_{i=1,2,\dots,m} \{ \hat{f}(x_i - h), \hat{f}(x_i + h) \}. \tag{17}$$

The last formula results from the fact that the maximum for a kernel estimator with the uniform kernel must occur on the edge of one of the kernels.

In the multidimensional case, von Neumann's elimination algorithm is similar to the previously discussed one-dimensional version. The edges of the n -dimensional cuboid $[a_1, b_1] \times [a_2, b_2] \times \dots \times [a_n, b_n]$ are calculated from formulas comparable to (15)-(17) separately for particular coordinates. The kernel estimator maximum is thus located in one of the corners of one of the kernels; therefore

$$c = \max_{i=1,2,\dots,m} \left\{ \hat{f} \left(\begin{bmatrix} x_{i,1} \pm h \\ x_{i,2} \pm h \\ \vdots \\ x_{i,n} \pm h \end{bmatrix} \right) \right\} \quad \text{following all combinations of } \pm \quad .(18)$$

The number of these combinations is finite and equal to 2^n . Using the formula presented, n particular coordinates of pseudorandom vector u and the subsequent number v are generated, after which condition (14) is checked.

5. Fuzzy and Intuitionistic Fuzzy Evaluations

Let us consider set (6) introduced in Section 3, consisting of elements representative for an investigated population, and extended as described in accordance with Section 4. In taking its subset comprising these observations x_i for which $\hat{f}(x_i) \leq \hat{q}_r$, one can treat it as a pattern of atypical elements. Denote it thus:

$$x_1^{at}, x_2^{at}, \dots, x_{m_{at}}^{at} \quad .(19)$$

Similarly, the set of observations for which $\hat{f}(x_i) > \hat{q}_r$ may be considered as a pattern of typical elements:

$$x_1^t, x_2^t, \dots, x_{m_t}^t \quad .(20)$$

Take the mean values of the kernel estimator \hat{f} on atypical elements (19):

$$s_{at} = \frac{1}{m_{at}} \sum_{i=1}^{m_{at}} \hat{f}(x_i^{at}) \quad , (21)$$

as well as on typical (20):

$$s_t = \frac{1}{m_t} \sum_{i=1}^{m_t} \hat{f}(x_i^t) \quad .(22)$$

Similarly, consider mean squares of deviations for both patterns representing atypical and typical elements respectively

$$v_{at} = \frac{1}{m_{at}} \sum_{i=1}^{m_{at}} [s_{at} - \hat{f}(x_i^{at})]^2 \tag{23}$$

$$v_t = \frac{1}{m_t} \sum_{i=1}^{m_t} [s_t - \hat{f}(x_i^t)]^2. \tag{24}$$

Let us define so-called reference values for sets of atypical w_{at} as well as typical w_t elements

$$w_{at} = 0 \tag{25}$$

$$w_t = \max_{i=1,2,\dots,m_t} \hat{f}(x_i^t) + \min_{i=1,2,\dots,m_{at}} \hat{f}(x_i^{at}) \cong \max_{x \in \mathbb{R}} \hat{f}(x_i^t) + \min_{i=1,2,\dots,m_{at}} \hat{f}(x_i^{at}). \tag{26}$$

Let for any $x \in \mathbb{R}^n$, the functions $d_{at} : \mathbb{R}^n \rightarrow [0, \infty)$ and $d_t : \mathbb{R}^n \rightarrow [0, \infty)$ be given as

$$d_{at}^2(y) = \frac{(y - w_{at})^2}{v_{at}} \tag{27}$$

$$d_t^2(y) = \frac{(y - w_t)^2}{v_t}, \tag{28}$$

informally (they do not fulfil the conditions of a metric or even semi-metric) illustratively interpretable as "distances" from reference values (25)-(26), standardized by variances (23)-(24), in sets of atypical and typical elements. With the above notations, the membership function for the set of atypical elements $\mu_{at} : \mathbb{R}^n \rightarrow [0,1]$ is defined by the formula

$$\mu_{at}(x) = \frac{1}{1 + \left(\frac{d_{at}(x)}{d_t(x)}\right)^{c_f}} = \frac{1}{1 + \left(\frac{d_{at}^2(x)}{d_t^2(x)}\right)^{\frac{1}{c_f}}}, \tag{29}$$

where the parameter $c_f > 0$ makes for the degree of fuzziness (standard assumed $c_f = 1$). Concerning correct interpretation it is worth modifying in formulas (27)-

(28) the parameters v_{at} and v_t inversely proportional, i.e. v_{at} is replaced by av_{at} and v_t by v_t/a , while $a > 0$. Initially it is assumed that $a = 1$, after which its value respectively increases or decreases to get $\mu_{at}(y) \equiv 0.5$, where y is such element that $\hat{f}(y) \equiv \hat{q}_r$.

The above procedure can be supplemented to generate intuitionistic fuzzy evaluation. Similar to formulas (25)-(28) the "distance" from the quantile estimator $d_{hm}(y) : \mathbb{R}^n \rightarrow [0, \infty)$ transposed through the reference point $w_{hm} > 0$ can be introduced, given by

$$d_{hm}^2(x) = \begin{cases} w_{hm} + \frac{(\hat{q}_r - \hat{f}(x))^2}{v_{at}} & \text{for } \hat{f}(x) \leq \hat{q}_r \\ w_{hm} + \frac{(\hat{f}(x) - \hat{q}_r)^2}{v_t} & \text{for } \hat{f}(x) \geq \hat{q}_r \end{cases}. \quad (30)$$

Particular functions defining an intuitionistic fuzzy set are described by the following formulas:

- the function $\mu_{at} : \mathbb{R}^n \rightarrow [0, 1]$ of membership to the set of atypical elements

$$\mu_{at}(x) = \frac{1}{1 + \left(\frac{d_{at}(x)}{d_t(x)}\right)^{\frac{2}{c_f}} + \left(\frac{d_{at}(x)}{d_{hm}(x)}\right)^{\frac{2}{c_f}}} = \frac{1}{1 + \left(\frac{d_{at}^2(x)}{d_t^2(x)}\right)^{\frac{1}{c_f}} + \left(\frac{d_{at}^2(x)}{d_{hm}^2(x)}\right)^{\frac{1}{c_f}}}, \quad (31)$$

- the function $v_{at} : \mathbb{R}^n \rightarrow [0, 1]$ of non-membership to the set of atypical elements (membership to the set of typical elements)

$$v_{at}(x) = \frac{1}{1 + \left(\frac{d_t(x)}{d_{at}(x)}\right)^{\frac{2}{c_f}} + \left(\frac{d_t(x)}{d_{hm}(x)}\right)^{\frac{2}{c_f}}} = \frac{1}{1 + \left(\frac{d_t^2(x)}{d_{at}^2(x)}\right)^{\frac{1}{c_f}} + \left(\frac{d_t^2(x)}{d_{hm}^2(x)}\right)^{\frac{1}{c_f}}}, \quad (32)$$

- the function $\pi_{at} : \mathbb{R}^n \rightarrow [0, 1]$ hesitation margin

$$\pi_{at}(x) = 1 - \mu_{at}(x) - v_{at}(x) \quad , \quad (33)$$

where $c_f > 0$ is a parameter indicating the degree of fuzziness (standard $c_f = 1$). The parameters v_{at} and v_t are modified inversely proportional, i.e. v_{at} is replaced in formulas (27)-(28) and (30) with av_{at} , and v_t with v_t/a , while $a > 0$. Initially it is assumed that $a = 1$, after which its value respectively increases or decreases, to get $\mu_{at}(y) \cong v_{at}(y)$, where y is such an element that $\hat{f}(y) \cong \hat{q}_r$. The value of the parameter w_{hm} should be established on the basis of individual conditions for the task under investigation. Initially one can assume $w_{hm} = 0.001$, and then increase depending on the desired level of $\pi_{at}(y)$, where y as previously is such an element that $\hat{f}(y) \cong q_r$; for instance $\pi_{at}(y) = 0.5$.

6. Verification Results

This section presents the results of illustrative numerical verification, which positively confirmed the correct functioning of the procedure for detection of atypical elements. Consider therefore the one-dimensional case, where the distribution characterizing the data in set (6) is bimodal with the following normal (Gauss) components and shares

$$N(-3,1) \quad 40\% \quad , \quad N(3,1) \quad 60\% . \tag{34}$$

Figure 1 displays the fuzzy evaluation. The membership functions to the sets of atypical and typical elements were shown there. The results are in line with intuition. It is worth noting that part of the membership function for the set of atypical elements in the region of the component $N(-3,1)$ assumes slightly lower values than in the region of the component $N(3,1)$ with a greater and therefore more distinct share. Similar conclusions concern the intuitionistic fuzzy evaluation shown in Fig. 2. Additionally, the hesitation margin function in the area of less distinct component $N(-3,1)$ is bigger than in that of the clearer component $N(3,1)$. Local maximums for the hesitation margin function are located on the assumed level 0.5.

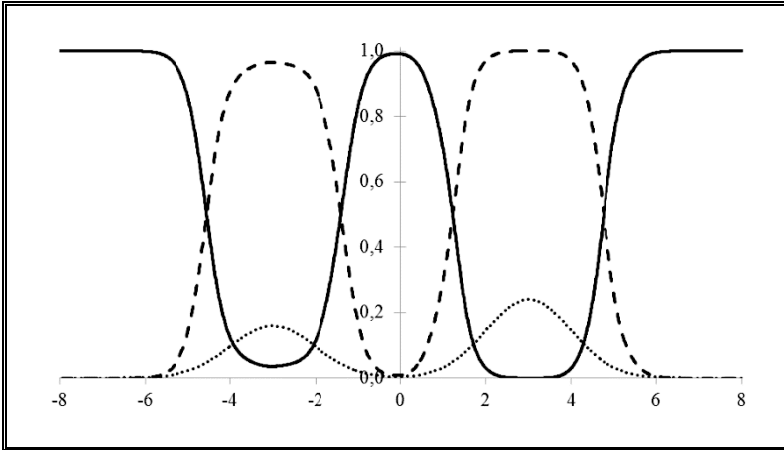


Fig. 1. Fuzzy evaluation; membership functions for sets of atypical (continuous line) and typical (broken line) elements and density (dotted line) for bimodal distribution (34);

$$r=0.1, m=1,000, m^*=10,000.$$

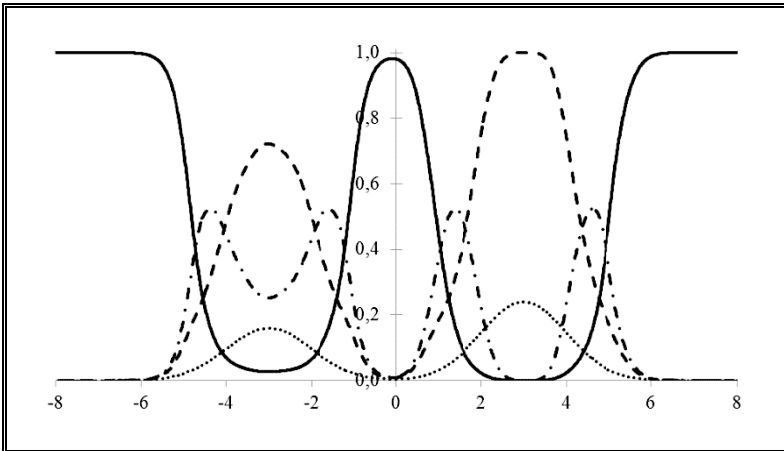


Fig. 2. Intuitionistic fuzzy evaluation; membership functions for sets of atypical (continuous line), typical (broken line) elements and hesitation margin (dotted-broken line), with density (dotted line) for bimodal distribution (34);

$$r=0.1, m=1,000, m^*=10,000.$$

It is also worth mentioning the computational complexity of the investigated method. Thus, calculation of the set (7) values has quadratic complexity with respect

to the size m^* , as does the entire procedure, whose particular algorithms are linear or quadratic. However, after defining the model's parameters, the actual application of the procedure with respect to a single tested element is of linear complexity. It is, therefore, worth stressing the possibility of the problem decomposition, and for practical uses it is to be recommended that the time-consuming computation of the model parameters values be carried out earlier, leaving only rapid testing to be done *on-line*.

A broader description of the concept presented here, in particular proofs of correctness of definitions introduced by formulas (21)-(33), and detailed results of verification research, also based on experimental data from medical tasks, can be found in the paper [Kulczycki, Kruszewski; 2017].

Bibliography

- Aggarwal C.C.: *Outlier Analysis*. Springer, New York, 2013.
- Atanassov K.: *Intuitionistic Fuzzy Sets. Theory and Applications*. Physica-Verlag, Heidelberg-New York, 1999.
- Barnett V., Lewis T.: *Outliers in statistical data*. Wiley, New York, 1994.
- Gentle J.E.: *Random Number Generation and Monte Carlo Methods*. Springer, New York, 2003.
- Kacprzyk J.: *Zbiory rozmyte w analizie systemowej*. PWN, Warsaw, 1986.
- Klir G.J, Yuan B.: *Fuzzy sets and fuzzy logic: theory and applications*. Prentice-Hall, Upper Saddle River, 1995.
- Kulczycki P.: *Wykrywanie uszkodzeń w systemach zautomatyzowanych metodami statystycznymi*. Alfa, Warsaw, 1998.
- Kulczycki P.: *Estymatory jądrowe w analizie systemowej*. WNT, Warsaw, 2005.
- Kulczycki P., Charytanowicz M.: *Conditional Parameter Identification with Different Losses of Under- and Overestimation*. Applied Mathematical Modelling, vol. 37, pp. 2166-2177, 2013.
- Kulczycki P., Charytanowicz M.: *A Complete Gradient Clustering Algorithm Formed with Kernel Estimators*. International Journal of Applied Mathematics and Computer Science, vol. 20, pp. 123-134, 2010.

- Kulczycki P., Charytanowicz M., Kowalski P.A., Łukasik S.: *Identification of Atypical (Rare) Elements – A Conditional, Distribution-Free Approach*. IMA Journal of Mathematical Control and Information, in press, 2017.
- Kulczycki P., Daniel K.: *Metoda wspomagania strategii marketingowej operatora telefonii komórkowej*. Przegląd Statystyczny, vol. 56, no. 2, pp. 116-134, 2009; errata: vol. 56, no. 3-4, s. 3, 2009.
- Kulczycki P., Kowalski P.A.: *A Complete Algorithm for the Reduction of Pattern Data in the Classification of Interval Information*. International Journal of Computational Methods, vol. 13, paper ID: 1650018, 2016.
- Kulczycki P., Kruszewski D.: *Identification of Atypical Elements by Transforming Task to Supervised Form with Fuzzy and Intuitionistic Fuzzy Evaluations*, in press, 2017.
- Kulczycki P., Łukasik S.: *An Algorithm for Reducing Dimension and Size of Sample for Data Exploration Procedures*. International Journal of Applied Mathematics and Computer Science, vol. 24, pp. 133-149, 2014.
- Kulczycki P., Prochot C.: Identyfikacja stanów nietypowych za pomocą estymatorów jądrowych. Bubnicki Z., Hryniewicz O., Kulikowski R. (eds.), *Metody i techniki analizy informacji i wspomagania decyzji*. EXIT, Warsaw, pp. 57-62, 2002.
- Kulczycki P., Waglowski J.: On the application of statistical kernel estimators for the demand-based design of a wireless data transmission system. Control and Cybernetics, vol. 34, pp. 1149-1167, 2005.
- Parrish R.: *Comparison of Quantile Estimators in Normal Sampling*. Biometrics, vol. 46, pp. 247-257, 1990.
- Szmidt E.: *Distances and Similarities in Intuitionistic Fuzzy Sets*. Springer, Cham, 2014.
- Wand M., Jones M.: *Kernel Smoothing*. Chapman and Hall, London, 1995.

Efficient transistor level implementation of selected fuzzy logic operators used in control systems

Tomasz Talaśka, Rafał Długosz and Paweł Skruch

¹ University of Science and Technology, Faculty of Telecommunication, Computer Science and Electrical Engineering, ul. Kaliskiego 7, 85-796, Bydgoszcz, Poland
talaska@utp.edu.pl,

WWW home page: <http://utp.edu.pl>

² Delphi Poland S.A., Podgorki Tynieckie 2, 30-399 Krakow, Poland
rafal.dlugosz@delphi.com & pawel.skruch@delphi.com,

WWW home page: <http://www.delphikrakow.pl/>

³ AGH University of Science and Technology, Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering, Department of Automatics and Biomedical Engineering, al. A. Mickiewicza 30/B1, 30-059 Krakow, Poland

pawel.skruch@agh.edu.pl,

WWW home page: <http://agh.edu.pl>

Abstract. The paper presents a novel, transistor level, implementation of selected fuzzy set operators suitable for fuzzy control systems realized in low-power hardware. We propose a fully digital, asynchronous realization of basic fuzzy logic (FL) functions, such as the bounded sum, bounded difference, bounded product, bounded complement, fuzzy logic union (MAX) and fuzzy logic intersection (MIN). All of the proposed operators has been implemented in the CMOS TSMC 180nm Technology and verified by means of transistor level simulations in Hspice environment. The proposed structures of the FL functions can easily be scaled to any signal resolutions.

Keywords: fuzzy systems, fuzzy operators, CMOS implementation

1 Introduction

A large interest in the fuzzy systems (FSs) and their possibilities are mainly due to the fact that the surrounding world is inherently fuzzy. Using only bivalent logic we are unable to properly describe many real world problems. The bivalent logic allows us for only a hard selection between the TRUE and the FALSE values. Such approach is insufficient in most systems, including, for example, control systems used in the automotive industry. Fuzzy sets offer more natural description of various problems, important for example from the point of view of vehicle control, however there is a demand for efficient rules how to process fuzzy logic (FL) data. Even if such rules are already known, and well described,

there is still a demand for their efficient implementation in hardware with limited resources, e.g. in embedded systems or in application specific integrated circuits (ASIC).

The properties of the fuzzy set theory resulted in its widespread use in many fields of automatic and control systems [1], in electric and electronic engineering [2], [3], [4], [5] in medicine in forecasting, planning and decision-making [6], [7].

It frequently happens also that FSs cooperate with neural networks (NNs) in control systems, leading to even better results [8]. There are also known such cases, in which the NN only supports the work of the FS. In this case we say about the neuro-fuzzy networks [9], [10]. A characteristic feature of these networks is their ability to use fuzzy inference methods to calculate the values of output signals. In contrast to conventional fuzzy systems (fuzzy logic), the systems in which NNs are involved offer adaptive selection of selected parameters, e.g. the shapes and parameters of the membership functions.

Fuzzy systems can be implemented using different techniques. The most popular way is their software realization. This is due to the ease of such implementation and test execution, as well as their large flexibility - modification abilities. On the other hand, such realizations are not suitable for many industry applications that require miniaturization, low energy consumption and low cost per unit.

Hardware implementations of FSs play an important role in many industrial applications in which FL is being used [11]. Such realizations include programmable devices such as microcontrollers (μC) [12] and FPGAs (Field Programmable Gate Arrays) [11], [13], [14]. FL systems are also realized as analog [15], [16], [17], [18], digital or mixed [19] ASICs. In case of analog approach the circuits are usually realized using the current-mode technique, in which summing and subtracting operations (commonly used in FL) are realized simply in junctions.

In the comparison with analog circuits, digital circuits offer several important advantages. The inherent regeneration of logic levels in logic gates involves high noise immunity and low sensitivity to the variance of transistor parameters. This allows for accurate and reliable data and signal processing. Additionally binary data can be easily stored even for a long period of time. This facilitates the realization of even very large, programmable, multi-stage fuzzy data processing.

In case of digital solutions a frequently asked question is whether to use the standard cell or the full-custom technique. The first approach is convenient, as in this case the designer usually only cares about the behaviour of the circuit (the logic), while generation of the layout is done automatically by design environment. The full-custom methodology, on the other hand, is much more flexible, as the designer can decide about every aspect of both the circuit and the layout design. In general, in both cases the target is layout, while the basic question is who is able to design a circuit with better parameters (human or machine). To our best knowledge, FL operators were not implemented as digital full-custom ASICs so far. We decided to use this technique, as particular FL operators proposed in this paper offer simple structure and can be very quickly designed

without using standard-cell approach. Additionally, in standard-cell technique we are limited to existing cells, which is often insufficient especially if there is foreseen a close cooperation of digital blocks with analog components.

To enable usage of FSs in modeling of real processes, it is important to implement families of parametric FL operators that can be easily tuned, in order to obtain better simulation results of the FSs. On-board and real-time applications of FSs require larger data processing rates. To achieve these objectives, we work on programmable, fast, miniaturized FL operators realized in hardware.

In this paper we present the realization of main FL operators, such as: bounded sum, bounded difference, bounded product, bounded complement, fuzzy logic union (MAX) and fuzzy logic intersection (MIN). In next Section we provide formulas describing these operators, as well as the proposed circuits representing them. In following Section we present transistor level verification of the proposed solutions in the 180 nm CMOS technology.

2 Basic Fuzzy Logic functions and their hardware implementation

The FL functions are defined in terms of the membership functions μ_A and μ_B . In the literature one can find more than ten basic FL functions [16], [18]. We focus on selected functions of this group as presented in this Section. All these functions have been implemented by us in the TSMC CMOS 180 nm technology using only basic digital combinational circuits that include standard AND, OR, XOR, NOT gates, as well as more complex circuits, designed from scratch, such as multi-bit full adder and subtractors that in some cases play the role of digital the comparators.

The proposed designs, presented below, can be very easily scaled down to any newer CMOS technology.

2.1 Fuzzy logic union(MAX) and Fuzzy logic intersection (MIN)

The FL union and intersection functions, as described using two formulas, respectively:

$$\mu_{A \vee B} = \max(\mu_A, \mu_B) \quad (1)$$

and

$$\mu_{A \wedge B} = \min(\mu_A, \mu_B) \quad (2)$$

Both these operators have a similar structure. For this reason we propose a programmable circuit, shown in Fig. 1, that can be easily switched over between both these functions. A multi-bit full subtractor (MBFS) is used in the role of the comparator (CMP). This component is realized a chain of 1-bit full subtractors (1BFS), coupled through the borrow in (B_{IN}) and borrow out (B_{OUT}) signals (not shown in the Figure). The B_{OUT} signal from the most significant 1BFS

becomes '1' in case, if the signal provided to the positive input of the comparator is smaller than its negative input signal. The B_{OUT} signal through the XOR gate controls the multiplexer, which provides to the outputs of the overall FL circuit either the smaller or the larger input signal.

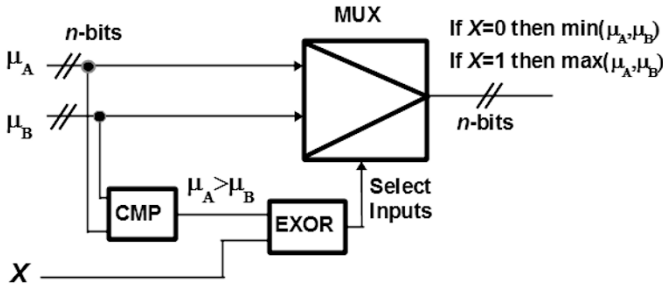


Fig. 1. Hardware implementation of the MIN and MAX FL functions.

2.2 Bounded complement

The bounded complement FL function is a multivalued extension of the binary NOT operation. It can be described using the formula:

$$\mu_{\bar{A}} = \mathbb{1} - \mu_A \tag{3}$$

The proposed corresponding circuit is shown in Fig. 2. It is based on the MBFS that subtracts the input signal μ_A from the maximum possible value, for a given signal resolution, n (in bits). The maximum value, equal to $2^n - 1$, is represented by the $\mathbb{1}$ symbol in 3. For example cases of 4 and 8 bits, the $\mathbb{1}$ equals (in HEX) 0xF or 0xFF, respectively.

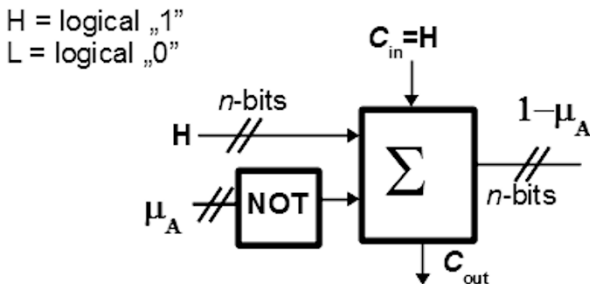


Fig. 2. Hardware implementation of the fuzzy complement function

2.3 Bounded sum

The bounded sum FL function is defined as follows:

$$\mu_{A \oplus B} = \min[\mathbb{1}, (\mu_A + \mu_B)] \quad (4)$$

The proposed circuit that is an example implementation of 4 is shown in Fig. 3. In the circuit, a multi-bit full adder (MBFA) is used to calculate the sum of both input signals. It is composed of a chain of 1-bit full adders (1BFAs), coupled through the carry-in (C_{IN}) and carry-out (C_{OUT}) signals (not shown for the simplicity).

If $\mu_A + \mu_B > \mathbb{1}$ then the C_{OUT} from the most significant 1BFA becomes 1 and this signal through the OR logic gate sets all output signals to '1'. The resolution of the output signal equals the resolution of the μ_A and the μ_B signals. It is so, to avoid the situation in which any operator increases or decreases the resolution of the original input signals.

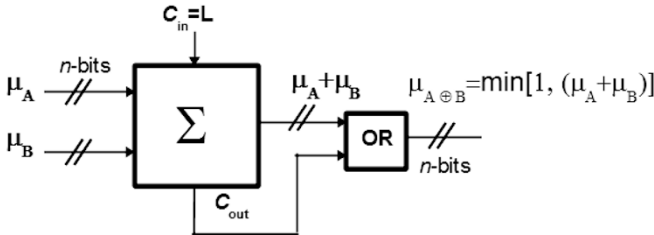


Fig. 3. Hardware implementation of the bounded sum function

2.4 Bounded difference

The bounded difference FL function is defined as follows:

$$\mu_{A \ominus B} = \max[0, (\mu_A - \mu_B)] \quad (5)$$

An example corresponding circuit is shown in Fig. 4. Here the structure of the circuit is a bit similar to the circuit shown in Fig. 3. Instead of using MBFA we use MBFS. If $\mu_A < \mu_B$ then $B_{OUT} = 1$ which throughout the NOT and the AND gates sets all outputs of the circuit to '0'.

2.5 Bounded product

Bounded product FL function is the last one presented in this paper. It is described using the formula:

$$\mu_{A \odot B} = \max[0, (\mu_A + \mu_B - \mathbb{1})] \quad (6)$$

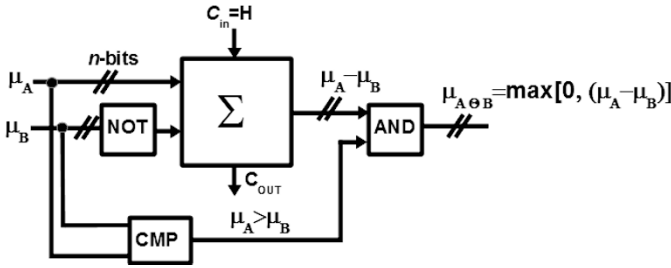


Fig. 4. Hardware implementation of the bounded difference function

The structure of the proposed circuit requires a bit more explanation. Here we have two add / sub operations: $\mu_A + \mu_B - \mathbb{1}$. The $\mathbb{1}$ equals the maximum value, for example 0xFF for the resolution of 8 bits. To subtract this number from the $\mu_A + \mu_B$ term, we can add its reversed and complemented value (U2 code) to the $\mu_A + \mu_B$ term. Independently on the signal resolution, the reversed and complemented value always equals ‘1’, so it is sufficient to add the ‘1’ signal at the least significant position. This allows to substitute the full scale MBFA with an incrementing circuit, that features a substantially simpler structure and thus consumes less power. An example corresponding circuit is shown in Fig. 5

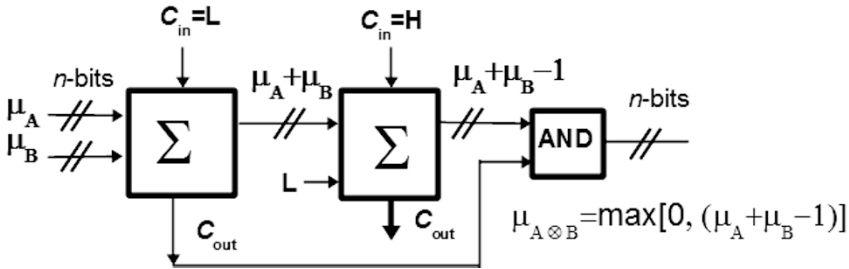


Fig. 5. Hardware implementation of the bounded product function

3 Transistor level verification of the proposed FL circuits

Verification of the proposed circuits have been made in Hspice simulation environment in the CMOS 180 nm technology. We performed the, so called, corner analysis in which the proposed circuits have been verified against the process, voltage and temperature (PVT) variation. A series of performed tests covered several transistor models, namely typical (T), fast (F) and slow (S), for temperatures varying in-between -20 and 100 °C and different values of the supply voltage (from 1.2 to 1.8 V). The signal resolution in the tests shown in Figs. 6–11

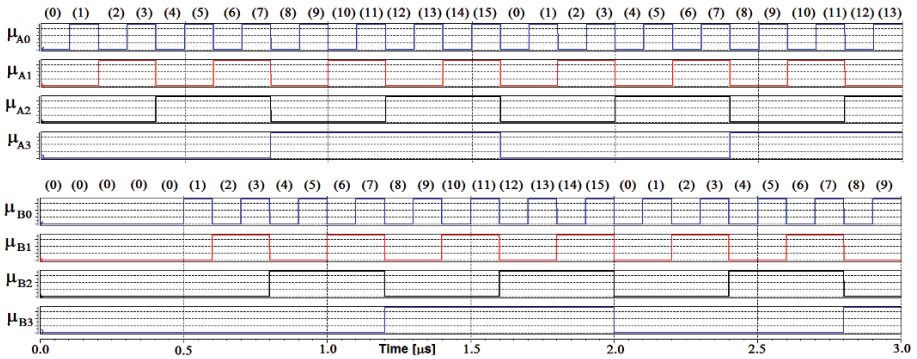


Fig. 6. Input signals

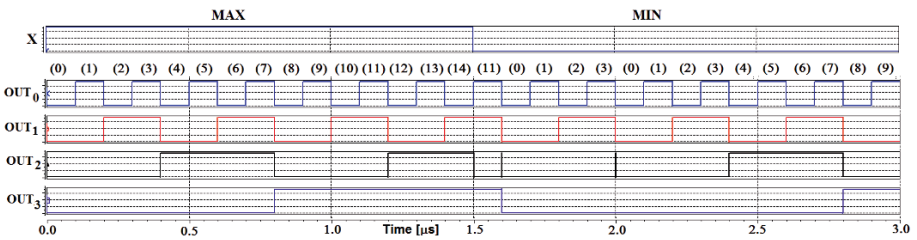


Fig. 7. Verification of the MIN and the MAX FL functions.

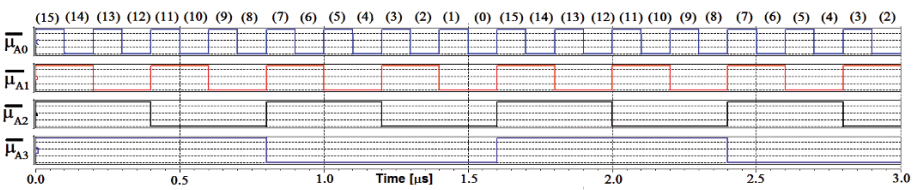


Fig. 8. Verification of the fuzzy complement function

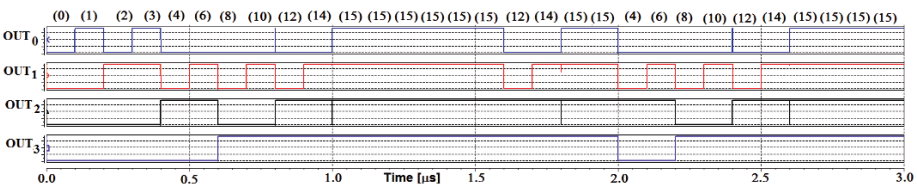


Fig. 9. Verification of the bounded sum function

was 4 bits, i.e. the real value of $\mathbb{1}$ was 0xF. During the tests the input signals varied from 0 to 0xF.

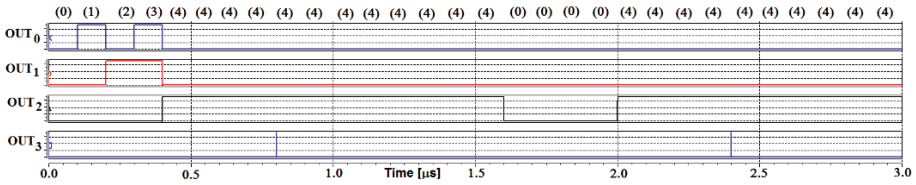


Fig. 10. Verification of the bounded difference function

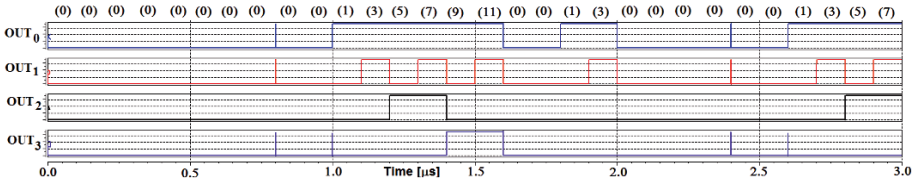


Fig. 11. Verification of the bounded product function

Verification on the logic level is an obvious test, and all the circuits worked properly. Our goal was to check also dynamic parameters i.e. delays introduced by particular operators, as well as the consumed energy. Selected results are shown in Fig. 12. The response time of particular operators varied from 0.25 ns for the complement operator to 0.6 ns for the bounded difference operator. There was also observed an influence of the PVT parameters. The difference between the worst case (S/1.2) and the best case (F/1.8 V) scenarios was about 120 %, however in each of the tested cases the circuit worked properly (at the logic level). Since the circuits are designed in the CMOS technology, therefore the static current is very small, while the circuits consume energy mostly during changing their state. For this reason, independently on data rate, the energy consumed per a single calculation is almost constant and varies from 0.54pJ to 1.26pJ, for bounded complement and bounded product, respectively. Taking the presented values into account, it can be estimated that in case of larger fuzzy systems in which, for example, ten operators are used in chains, the expected data rate will be even 200-400 MSamples/s.

4 Conclusions

In the paper we present a novel transistor-level implementation of selected fuzzy logic operators, suitable for very fast, low power dissipation applications. The proposed circuits operate fully asynchronously, which means that no clock generator is required to perform particular FL operations. The circuits can be used in larger fuzzy systems working in parallel, with whole chains of the operators working asynchronously.

Particular operators, designed in the CMOS 180 nm technology, are very fast and offer data rates at the level of even several GSamples/s. These parameters

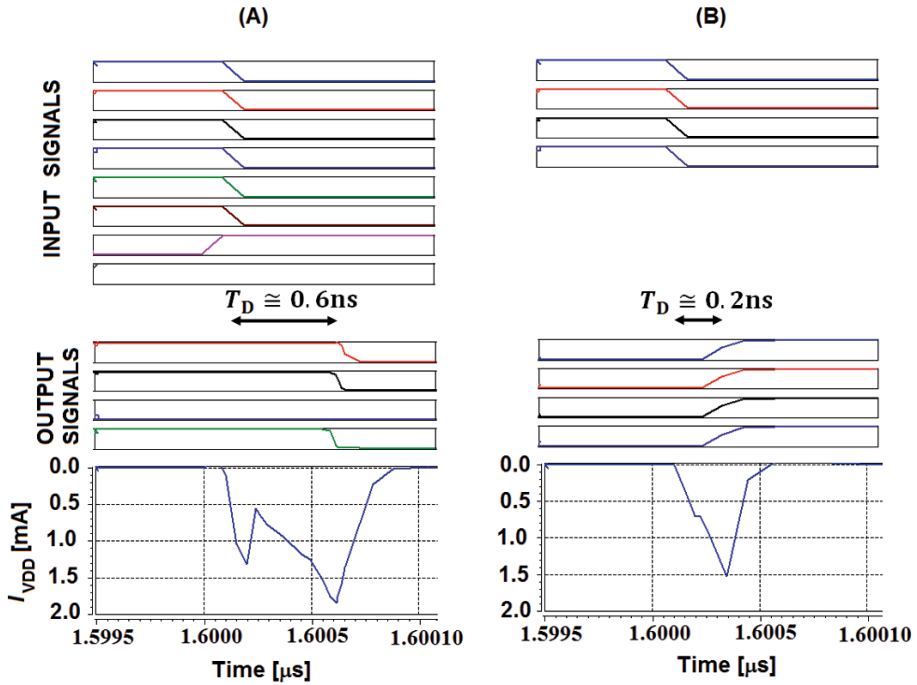


Fig. 12. Illustration of the delay introduced by the bounded product and bounded complement operator and corresponding current consumption. (A) - bounded product (B) - bounded complement

can be substantially improved if the circuits will be redesigned in newer CMOS technologies. The circuits were designed in full-custom style. The structures of the used full adders and full subtractors have been proposed by the authors. The aim was minimization of the number of transistors in these circuits.

References

1. Tan, Q. Wei, Q., Hu J., Aldred D.: Road vehicle detection using fuzzy logic rule-based method. International Conference on Fuzzy Systems and Knowledge Discovery, 3, (2010)
2. Chen, Z. Gomez, S.A., McCormick M.: A fuzzy logic controlled power electronic system for variable speed wind energy conversion systems, International Conference on Power Electronics and Variable Speed Drives, (2000)
3. Czaja, Z., Zaleski, D.: Employing a fuzzy logic based method to the fault diagnosis of analog parts of electronic embedded systems, IEEE Instrumentation & Measurement Technology Conference (IMTC), (2007)
4. Bose, B. K.: Fuzzy logic and neural networks in power electronics and drives, IEEE Industry Applications Magazine, 6,3, (2000)

5. Ciabattoni, L., Grisostomi, M., Ippoliti, G., Longhi, S.: A Fuzzy Logic tool for household electrical consumption modeling, Annual Conference of the IEEE Industrial Electronics Society (IECON), (2013)
6. Seker, H., Odetayo, M. O., Petrovic, d., Naguib, R. N. G.: A fuzzy logic based-method for prognostic decision making in breast and prostate cancers, IEEE Transactions on Information Technology in Biomedicine, 7,2, (2003)
7. Seizaburo, A., Tsunehisa, N., Hiroshi, S.: Diagnostic System of Breast Cancer based on Imaging Data of Mammography using Fuzzy Logic, World Automation Congress, (2006)
8. Li, T.H. S., Chen, Ch-Y., Lim, K-CH.: Combination of fuzzy logic control and back propagation neural networks for the autonomous driving control of car-like mobile robot systems, Proceedings of SICE Annual Conference, (2010)
9. Kayacan, E., Kayacan, E.L., Ramon, H., Saeys, W.: Adaptive Neuro-Fuzzy Control of a Spherical Rolling Robot Using Sliding-Mode-Control-Theory-Based Online Learning Algorithm, IEEE Transactions on Cybernetics, 43,1, (2013)
10. Allah Hooshmand, R., Parastegari, M., Forghani, Z.: Adaptive neuro-fuzzy inference system approach for simultaneous diagnosis of the type and location of faults in power transformers, IEEE Electrical Insulation Magazine, 28,5, (2012)
11. Yen, J., Langari, R., Zadeh L.A.: Industrial Applications of Fuzzy Logic and Intelligent Systems, IEEE Press, New York, (1995)
12. Nagaraj, R., Mayurappriyan, P.S., Jerome, J.: Microcontroller based fuzzy logic technique for dc-dc converter, International Conference on Power Electronics, (2006)
13. I. J. Rudas, Ildar Z. Batyrshin, A. Hernandez Zavala, O. Camacho Nieto, L. Horvth, L. Villa Vargas, "Generators of Fuzzy Operations for Hardware Implementation of Fuzzy Systems", Advances in Artificial Intelligence, 7th Mexican International Conference on Artificial Intelligence (MICAI), (2008)
14. Kandel, A., Langholz, G.: Fuzzy Hardware: Architectures and Applications, Kluwer Academic Publishers, New York, (1998)
15. Guo, S., Peters, L., Surmann, H.: Design and application of an analog fuzzy logic controller, IEEE Transactions on Fuzzy Systems, 4, 4, (1996)
16. Yamakawa, T. Miki, T.: The Current Mode Fuzzy Logic Integrated Circuits Fabricated by the Standard CMOS Process, IEEE Transactions on Computers, C-35, 2, (1986)
17. Jaworski, Z., Niewczas, M., Grygolec M., Kuzmich W.: Architecture of a testable analog fuzzy logic controller, IEEE Transactions on Fuzzy Systems, 4, 4, (1996)
18. Dlugosz, R., Pedrycz, W.: Lukasiewicz fuzzy logic networks and their ultra low power hardware implementation, Neurocomputing, Elsevier, 73, (2010)
19. Baturone, I., Sanchez-Solano, S., Barriga, A., Huertas, J.: Implementation of CMOS Fuzzy Controllers as Mixed-Signal Integrated Circuits, IEEE Transactions on Fuzzy Systems, 5, 1, (1997)

Part XI
Biomedical Applications of Control
Engineering

Optimization of Combined Anticancer Treatment Using Models With Multiple Control Delays

Helmut Maurer and Andrzej Świerniak

¹ University of Munster, Institute of Computational and Applied mathematics,
Einsteinstr. 62, 48149 Munster, Germany

² Silesian University of Technology, Department of Automatic Control, Akademicka
16, 44101 Gliwice, Poland

Abstract. In the paper we study some control properties of a two compartmental model of response to anticancer treatment which combines antiangiogenic and cytotoxic drugs and takes into account multiple delays in control. More precisely we formulate sufficient local controllability conditions for semilinear systems which result from the use of Hakufeldt et al. model, endowed with two control variables representing antiangiogenic modality, combined with chemotherapy which contains delays related to PK/PD effects and some clinical recommendations e.g. normalization of vascular network. Then the optimized protocols of the combined therapy for the model, considered as solutions of an optimal control problem for dynamical systems with delays in control, are found using necessary conditions of optimality and numerical calculations. Structural sensitivity of considered control properties and optimal solutions are also discussed.

1 Biological background

Cancer disease is the most common cause of death in industrialized countries. Cancers are fully developed malignant tumours with a specific capacity to invade and destroy the underlying mesenchyme (local invasion). The tumour cells need nutrients via the bloodstream and produce a range of proteins that stimulate the growth of blood vessels into the tumour, thus allowing continuous growth to occur. The new vessels are not well formed and are easily damaged so that the invading tumour cells may penetrate these and lymphatic vessels. This process (blood vessel formation from existing vascular network) is one of the hallmarks of cancer [1].

Tumour fragments may be carried in these vessels to local lymph nodes or to distant organs where they may produce secondary tumours. Over sixty years ago, Glenn Algire, studying physiological responses to tumour growth in mice at the National Cancer Institute, observed that the growth of tumour is dependent on the development of vascular supply. After observing the same phenomenon in 1971 Judah Folkman suggested the substantial potential of tumour angiogenesis as a therapeutic target [2].

Tumours, like normal tissues, have physiological constraints, on growth, such as access to oxygen and nutrients for metabolism. The diffusion of oxygen in tissues is limited to a distance of about $150\ \mu\text{m}$, thus tissue growth is restricted to a few cubic millimetres if no new vasculature is formed. For this reason, tumours remain in a dormant state restricted to a few millimetres in diameter unless they develop in a well-vascularised area or are able to recruit their own vasculature. For vascularisation to occur, the nearest vessel or capillary needs to become destabilised so that the endothelial cells lining the vessel can loosen from their neighbours, migrate through the extracellular matrix towards the tumour. Only after a tumour has recruited its own blood supply it can expand in size. Tumours do this via the production of angiogenic factors secreted into local tissues and stroma; this process has been termed the angiogenic switch. Since in normal healthy adults, the process of angiogenesis is very limited, thus it should, at least in theory, be possible to inhibit tumour angiogenesis without affecting normal tissues. Antiangiogenic therapies have become one of the most promising approaches in the anti-cancer drug development. Successful preclinical research data lead to clinical trials based on different strategies. Approaches currently under evaluation for inhibiting angiogenesis may either be direct (targeting cell surface bound proteins/receptors) or indirect (targeting growth factor molecules). Because angiogenesis is a complex process with multiple, sequential, and interdependent steps, this complexity creates many potential targets for inhibition. Therefore, an antiangiogenic effect can be achieved by targeting angiogenic stimulators, angiogenic receptors, extracellular matrix proteins, extracellular matrix proteolysis, control mechanisms of angiogenesis, or the endothelial cells directly. One of the considerations with this therapeutic approach is that by blocking a particular tumour cell property, such as VEGF production, this may be subject to inactivation over time by classic acquired resistance mechanisms since, for example, tumour cells can produce a number of different pro-angiogenic growth factors. Furthermore, VEGF is produced by many various cell types and may also be sequestered in the extracellular matrix and the interstitial spaces between tissues. However, because angiogenesis is dependent on an appropriate growth factor environment, and VEGF is the most important of these growth factors, the use of blocking molecules to VEGF is keenly investigated. Other strategies for targeting control of angiogenesis include interference with oncogenes and hypoxia response pathways. Hypoxia response pathways are promising antiangiogenic targets, particularly because hypoxic cells release many antiangiogenic factors and are also often resistant to radiotherapy and chemotherapy [3].

This illustrates the complexity of developing such novel approaches to cancer therapy. Blood vessel growth has turned out to be complex, with many factors playing a role in the process. This complexity in itself, however, leads to many opportunities for therapeutic targeting. Despite the fact that these approaches put forward an innovative idea for successful cancer treatment, at present there are a number of problems in clinical trials on humans that require very attentive studies and critical interpretations. Compounds that perform quite well in preclinical studies, fail to give similar results in patients with cancer. There are

more than a few reasons that can explain the presence of differences between preclinical and clinical outcomes [4].

Antiangiogenic agents make not such impressive results as in preclinical trials. Depending on a disease stage different results were obtained. In some cases slowed metastatic disease progression occurs, leading to progression-free survival and overall survival benefits compared with control, but it was not associated with survival improvements. Yet another important constrain in efficient antiangiogenic therapy is the accessibility of antiangiogenic agents. The genetic instability and high mutation rate of tumour cells is responsible, in part, for the frequent emergence of acquired drug resistance with conventional cytotoxic anticancer therapy. However, vascular endothelial cells, like bone marrow cells, are genetically stable and have a low mutation rate.

Therefore, Kerbel proposed in 1991 a hypothesis that antiangiogenic therapy would be a strategy to bypass drug resistance [5]. Unfortunately, contrary to Kerbel's hopes, two types of resistance have been observed. First one, evasive, include revascularization as a result of upregulation of alternative pro-angiogenic signals, protection of the tumor, increased metastasis, second one, intrinsic, includes rapid adaptive responses, in the case of pre-existing conditions defined by the absence of any beneficial effect of anti-angiogenic agents [6].

Therefore, nowadays antiangiogenic therapy is considered rather as an essential component of multidrug cancer therapy [7,8] especially with chemotherapy. Although tumor eradication in such combined therapy may be still the primary goal the chaotic structure of the angiogenically created network leads to another target for antiangiogenic agents. Namely using angiogenic inhibitors to normalization of the abnormal vasculature (the so-called pruning effect) facilitate drug delivery [9], [10].

Smaller dose of anti-angiogenic agents (bevacizumab 5 *mg/kg*) shows significantly different (higher) median survival from chemotherapy alone in the treatment group when the dose 10 *mg/kg* can even increase survival compared to chemotherapy alone in the treatment group. The continuous treatment with angiogenic inhibitors ultimately leads to a decrease in tumor blood flow and a decreased tumor uptake of co-administrated cytotoxic drugs. In the periodic therapy the main goal of anti-angiogenic agents is to normalize tumor vasculature. A number of antiangiogenic clinical trials currently in progress have been designed to compare the effects of a particular cytotoxic agent alone with the effects of the same agent in combination with an angiogenesis inhibitor. These effects of combination therapy, which have also been observed for the combination of radiation therapy and angiogenesis inhibitors [11], could play a significant role in the clinical evaluation and effects of angiogenesis inhibitors. It is also worth mentioning, that antiangiogenic therapy was found to be efficient for slowly growing tumours, which are difficult target for classical chemotherapy. The administration of cytotoxic drugs, often results in significant side effects. Drug side-effects may reflect either the primary anti-proliferative action of the drug, some less well understood but predictable toxicological effect or they may be entirely idiosyncratic. Whereas over the years of application, side-effects of chemotherapy are

already relatively well investigated, we still do not know much about side-effects of antiangiogenic therapy. Obvious complications might be related to menstruation, diabetes and wound healing. Nevertheless long-term effects of therapy require attention. Additionally it has been observed that antiangiogenic agents do require a very high dose to fulfil their function. As mentioned before drug resistance in cancer is common. Some tumours are inherently unresponsive to cytotoxic chemotherapy. Others may respond well initially but relapse rapidly with drug-resistant disease. Many factors have been implicated in cellular resistance and these mechanisms may be drug or class specific. Pharmacokinetic factors also contribute towards mechanisms of resistance. For example, it is important to realize that for many anticancer drugs the administered form of the drug is not necessarily the active form. Variability in, for example, levels of activating or inactivating enzymes in the host tissues and in the tumour can lead to significant additional inter- and intraindividual variation in terms of normal tissue toxicity and anti-tumour efficacy from such drugs. Generally pharmacokinetic effects should be taken into account in scheduling anticancer drugs. While cytotoxic drugs mostly have a half-life time of about few hours, the half life-time of antiangiogenic agents may vary over a wide range, from 15 minutes (e.g. angiostatin) up to 20 days (bevacizumab). Drug resistance may lead to lack of response at the time of treatment or, following an initial response the tumour regrows. On regrowth, a decision may be made whether to retreat with the same regimen or switch to second line therapy. This decision is usually based on the initial response to the drug and to the specific drug free interval.

In some sense drawbacks of chemotherapy (induced drug resistance, smaller efficiency for slowly growing tumours) could be supported by advantages of antiangiogenic therapy and drawbacks of this therapy could be at least slightly moderated by the advantages of chemotherapy.

We concentrate on the class of two-compartmental models proposed by Hahnfeldt et al. [13] with two control variables representing effects of two anticancer modalities and multiple delays introduced in these control variables to take into account PK/PD effects and additional requirements resulting from clinical recommendation (for example delay in use of cytotoxic agent sufficient for pruning vessels by antiangiogenic agent).

2 Models of combined therapy with multiple control delays

Hahnfeldt et al. in 1999 [13] proposed a model based on experimental data from anti-angiogenic therapy and non therapy trials of Lewis lung tumors in mice. Roughly speaking the main idea of this class of models is to incorporate the spatial aspects of the diffusion of factors that stimulate and inhibit angiogenesis into a non-spatial two-compartmental model for cancer cells and vascular endothelial cells. If p denotes the size of cancer cells population and q a parameter describing the size of vascular network then such growth could be expressed by Gompertz type growth equation. Second equation describes vascular network

growth, includes stimulators of angiogenesis, inhibitory factors secreted by tumor cells and natural mortality of the endothelial cells. In this model ξ denotes proliferation ability of the cells. Inhibitory factors are proportional to $p^{2/3}$, the tumor volume to the power $2/3$, because they concentrate nearby the area of the active surface between the tumor and vascular network. The effect of therapy in such models can be included in the form of control actions entering the system as multipliers in the bilinear terms. Since antiangiogenic agents disturb directly only the vascular network the control variable u is present only in the first equation. The second variable related to chemotherapy appears in both equations [14]. The coefficients φ, η, γ are non-negative constants (conversion factors) that relate the dosages of anti-angiogenic u and cytostatic v agents (φ is much greater than η).

$$\dot{p} = -\xi p \ln \left(\frac{p}{q} \right) - \varphi v p, \tag{1}$$

$$\dot{q} = b p - d q p^{2/3} - \mu q - \gamma u q - \eta v q. \tag{2}$$

Similar behavior could be obtained if Gompertz type growth is substituted by logistic type one:

$$\dot{p} = -\xi p \left(1 - \frac{p}{q} \right) - \varphi v p. \tag{3}$$

The modification of this model, proposed by d’Onofrio and Gandolfi [15], which also satisfies Hahnfeldt’s suggestions described above with the only difference which may be called vascularity based stimulation in contrast to tumour based stimulation in the original Hahnfeldt model.

$$\dot{q} = b q - d q p^{2/3} - \mu q - \gamma u q - \eta v q. \tag{4}$$

Combining the models of tumour growth and the associated models of vascular network growth we obtain a set of two-compartmental models properties of which have been compared in [16].

Yet another simplification of the original Hahnfeldt model was proposed by Ergun et al. in 2003 [17]. This model also satisfies assertions proposed in [13] but the dynamics of vessel carrying support is independent of the size of the tumor:

$$\dot{q} = b q^{2/3} - d q^{4/3} - \gamma u q - \eta v q. \tag{5}$$

Moreover this model does not contain the natural mortality factor.

In 2009, d’Onofrio and Gandolfi analyzed the role of vessel density [10] (which can modulate the effect of drugs) and the effect of vascular "pruning" (by using anti-angiogenic drug in a combined therapy). Too aggressive or sustained anti-angiogenic treatment may prune away vascular network, resulting in resistance to further treatment and in and inadequate for delivery of drugs or oxygen. This aspect is not included in original Hahnfeldt et al. model. It has been observed in simulation studies [18] based on functions described in [10] that the best properties of vascular network are when density (endothelial cells/cancer cells) is 2.

As we have already mentioned one of the main difficulties in planning such combined therapies is related to pharmacokinetic/pharmacodynamic (PK/PD) properties of the drugs. In [19] we have proposed to model this effects by including time delays different for different agents. A standard model that describes PK effect assumes that the concentration c of a drug given at a dose rate u has the form of the first order equation:

$$c = -ac + u, c(0) = 0 \quad (6)$$

where a is the so called clearance rate of the drug. PD models describe the effects of the drug concentrations $D(c)$ on the tumor cells and their role is to capture the behavior at low and high concentration. One possibility is to use sigmoidal functions e.g.

$$D(c) = \frac{E_{\max}c^n}{(EC_{50})^n + c^n} \quad (7)$$

where E_{\max} denotes the maximum effect EC_{50} denotes the concentration at which half of this maximum effect is realized and $n > 1$ is a positive integer. Combining these two effects (i.e. inertia with sigmoidal nonlinearity) leads to delay in control actions which we propose to model by a simple constant time-delay incontrol variables. The advantage of such approach is rather easy way in which a parameter describing PK/PD effect could be estimated. Although for many cytostatic drugs the half life time is rather short (few hours) but for Cisplatin which belongs to the most commonly used agents it changes from 30-100 hours. The variety for antiangiogenic agents is even more significant starting from 20 minutes for angiostatin and ending with 20 days for bevacizumab. The delays in the models may be introduced also to illustrate the idea of vessel pruning which demands administration of chemotherapy with delay with respect to antiangiogenic agents. To include delays in controls we may modify equations (1), (2), (3). In the simplest case we consider delays in chemotherapy protocols which is justified for example if we combine Sunitinib (angiogenic inhibitor) with Cisplatin or in both types of agents if we combine two different antiangiogenic agents e.g. Erlotinib and Bevacizumab with Cisplatin or Paclitaxel.

Thus (1-3) should be substituted by

$$\dot{p}(t) = f(p(t), q(t)) - \varphi p(t) v(t - h), p(0) = p_0, \quad (8)$$

$$\dot{q}(t) = b p(t) - q(t) (d p(t)^{2/3} + \mu + \gamma_1 u(t) + \gamma_2 u(t - h_1) + \eta v(t - h)), q(0) = q_0, \quad (9)$$

where f is a growth function given by one of the two growth functions:

$$\begin{aligned} \text{Gompertz growth function } f(p, q) &= -\xi p \ln(p/q), \\ \text{Logistic growth function } f(p, q) &= \xi p(1 - p/q). \end{aligned}$$

It is important to remember that for equations with delays the initial condition should contain not only conditions for p and q but also initial function for v in the interval $[-h, 0)$ and (possibly) for u in $[-h_1, 0)$. We will assume:

$$u(\tau) = 0 \text{ for } -h_1 \leq \tau < 0; v(\tau) = 0 \text{ for } -h \leq \tau < 0. \quad (10)$$

3 Optimal control problem

To our knowledge [17] was the first paper in which optimal protocol for combined anticancer therapy (antiangiogenic agents combined with radiotherapy) were discussed. The authors used a simplified model of angiogenesis (see section 2) combined with LQ model of effects of radiation therapy (see e.g. [20]), and found that optimal strategies for optimization problem with free final time combine bang-bang and singular controls. Ledzewicz and Schättler [21] presented a complete solution in the form of an optimal synthesis for control problem related to antiangiogenic therapy for this model. The same authors obtained a similar optimal strategy containing singular arcs for the original Hahnfeldt et al. model [22].

Meanwhile different results are obtained for the d'Onofrio-Gandolfi model (see section 2) in the case of a fixed time of antiangiogenic therapy [23]. The most important conclusion is that intermediate doses of a drug are not optimal and that the optimal protocol contains switches between maximal dose and no drug intervals. Singular arcs are not feasible since there are no finite intervals of constant solutions to the adjoint equations. Similar properties were found for the Hahnfeldt et al. model with logistic tumour growth [16]. In [16] and [24] the broad class of models from this family of models were analysed and the results from [21,22,23] were confirmed as special cases. Suboptimal strategies for the original Hahnfeldt et al. model for minimization of tumor volume with antiangiogenic therapy using bang-bang optimal controls were described in [25]. Simple suboptimal protocols for models with and without a linear pharmacokinetic equation are presented in [26] and [27]. For piecewise constant dosage protocols, a very good approximation to optimal solutions may be obtained; however, small doses have no significant effect on tumor development, but on the other hand a too high dosage is not efficient enough to justify its enormous cumulative cost. Preliminary results about optimal controls for a mathematical model that combines antiangiogenic therapy with a chemotherapeutic killing agent were presented in [28] and [14] for the d'Onofrio-Gandolfi model and the fixed treatment horizon. Once more the optimal strategy had no singular arcs. This becomes a multicontrol problem and the structure of a synthesis of optimal controls is significantly more complex than in the monotherapy. Nevertheless in [29] theoretical results leading to an optimal synthesis were presented for a family of models similarly as in [24] and numerical results for the original Hahnfeldt model and the free treatment time were presented. In [30] some results for systems with delays in control and the objective function combining final values of both state variables (similarly as in [23] and [28]), and fixed control horizon were presented. In this

paper we compare results for models with and without control delays, free and fixed terminal time, and discuss the effect of different cancer growth function. We formulate an optimal control problem similarly as in [24], i.e. only final cancer cell population is minimized. The solution is based on the necessary conditions given by a Maximum Principle for problems with delays and numerical procedure using a large-scale nonlinear programming (NLP), for a discretised problem and arc-parametrization method for direct optimization of switching times.

The NLP problem can be conveniently formulated with the help of the Applied Modeling Programming Language AMPL created by Fourer et al. [31] which is linked to the Interior-Point solver IPOPT developed by Wächter and Biegler [32]. We mostly use 5000 – 10000 grid points and the trapezoidal rule as integration method.

Alternatively, the control package NUDOCSS (cf. [33]) provides a highly efficient method for solving the discretized control problem, because it allows to implement higher order integration methods. In the second step the switching times are optimized directly using the arc-parametrization method [34]. This approach requires that a singular control can be determined in feedback form. In [35] optimal and suboptimal protocols for this class of problems were compared. In this section we present results of optimization for the Hahnfeldt et al. model with logistic type growth with multiple delays in the control variables. We compare the results for the model without delays and with delays and for free and fixed terminal time. In the former case we compare the results with the ones for the model with Gompertz type growth.

We consider the optimal control problem for a class of models given by (8), (9), (10).

To measure the total amount of control agents used, we introduce the two artificial state variables y and z defined by the differential equations

$$\dot{w}(t) = u(t), \quad w(0) = 0 \quad (11)$$

$$\dot{z}(t) = v(t), \quad z(0) = 0 \quad (12)$$

and prescribe the control constraints

$$0 \leq u(t) \leq u_{\max}, \quad 0 \leq v(t) \leq v_{\max} \quad \forall \quad 0 \leq t \leq T, \quad (13)$$

$$\int_0^T u(t)dt = w(T) \leq w_{\max}, \quad \int_0^T v(t)dt = z(T) \leq z_{\max}. \quad (14)$$

Then the optimal control problem [OC] consists in determining measurable (in practice piecewise continuous) control functions $u, v : [0, T] \rightarrow \mathbb{R}$ that:

$$J(u, v) = p(T) \quad (15)$$

We shall take the parameters from [28] and only adopt new parameters associated with the delayed control variables:

$$\begin{aligned} \xi &= 0.084, \quad b = 5.85, \quad d = 0.00873, \quad \gamma_1 = 0.15, \\ \gamma_2 &= 0.1, \quad \varphi = 0.1, \quad \eta = 0.1, \quad \mu = 0.02. \end{aligned}$$

In the non-delayed case with $h = h_1 = 0$ we set $\gamma_2 = 0$. Note that the control v enters only as a delayed control $v(t - h)$. Thus we can expect that the optimal control will shift the non-delayed control $v(t)$ to the left by h time units. This will be confirmed later by our computations.

Let us briefly discuss the necessary optimality conditions of a *Maximum Principle* as they were recently derived in [36] for an optimal control problem with multiple control and state delays. We denote by v_d the delayed control variable with $v_d(t) = v(t - h)$ and by u_d the delayed control variable with $u_d(t) = u(t - h_1)$. Denoting the state vector by $\underline{x} = (p, q, w, z) \in \mathbb{R}^4$ and the adjoint variable by $\underline{\lambda} = (\lambda_p, \lambda_q, \lambda_y, \lambda_z) \in \mathbb{R}^4$. The Hamiltonian of the delayed control problem is given by

$$H(\underline{x}, \underline{\lambda}, u, v, u_d, v_d) = \lambda_p(f(p, q) - \varphi p v_d) + \lambda_q(bp - q(dp^{2/3} + \gamma_1 u + \gamma_2 u_d + \eta v_d)) + \lambda_y u + \lambda_z v. \tag{16}$$

Since there is no delay in the state variables, the adjoint equations $\dot{\underline{\lambda}}(t) = -H_{\underline{x}}[t]$ do not contain the advanced times: which yields explicitly

$$\begin{aligned} \dot{\lambda}_p(t) &= -\lambda_p(t)(f_p(p(t), q(t)) - \varphi v(t - h)) - \lambda_q(t)(b - \frac{2}{3}q(t) dp(t)^{-1/3}), \\ \dot{\lambda}_q(t) &= -\lambda_p(t)f_q(p(t), q(t)) + \lambda_q(t)(dp(t)^{2/3} + \mu + \gamma_1 u(t) + \gamma_2 u(t - h_1) + \eta v(t - h)), \\ \dot{\lambda}_w(t) &= 0, \\ \dot{\lambda}_z(t) &= 0. \end{aligned} \tag{17}$$

Herein, the subscripts p and q in function f denote partial derivatives of $f(p, q)$ with respect to p and q (respectively). In view of the objective (15), the transversality conditions are given by

$$\lambda_p(T) = 1, \quad \lambda_q(T) = 0, \quad \lambda_w(T)(w(T) - w_{\max}) = 0, \quad \lambda_z(T)(z(T) - z_{\max}) = 0. \tag{18}$$

Our computations show that $w(T) = w_{\max}$, $z(T) = z_{\max}$ always holds for the chosen data. Then the last two equations in (18) mean that $\lambda_y(T)$ and $\lambda_z(T)$ are undetermined.

The optimal control $u(t)$ minimizes the sum (cf. [36])

$$H(\underline{x}(t), \underline{\lambda}(t), u, v(t), u(t-h_1), v(t-h)) + \chi_{[0, T-h_1]} H(\underline{x}(t+h_1), \underline{\lambda}(t+h_1), u(t+h_1), v(t+h_1), u, v$$

with respect to $u \in [0, u_{\max}]$, where $\chi_{[0, T-h_1]}$ denotes the characteristic function. Likewise, the optimal control $v(t)$ minimizes the sum

$$H(\underline{x}(t), \underline{\lambda}(t), u(t), v, u(t-h_1), v(t-h)) + \chi_{[0, T-h]} H(\underline{x}(t+h), \underline{\lambda}(t+h), u(t+h), v(t+h), u(t-h_1) +$$

with respect to $v \in [0, v_{\max}]$. Since both controls appear linearly in the Hamiltonian, we are lead to the switching functions

$$\begin{aligned} \phi_u(t) &= -\lambda_q(t)\gamma_1 q(t) - \lambda_y(t) - \chi_{[0, T-h_1]} \lambda_q(t+h_1) q(t+h_1)\gamma_2, \\ \phi_v(t) &= -\lambda_z(t) - \chi_{[0, T-h]} (\lambda_p(t+h)p(t+h)\varphi + \lambda_q(t+h)q(t+h)\eta) \end{aligned} \tag{19}$$

which determine the minimizing controls by the *control law*

$$\underline{u}(t) = \left\{ \begin{array}{ll} 0, & \text{if } \phi_{\underline{u}}(t) > 0 \\ \text{singular,} & \text{if } \phi_{\underline{u}}(t) = 0 \quad \forall t \in I_s \subset [0, T] \\ \underline{u}_{\max}, & \text{if } \phi_{\underline{u}}(t) < 0 \end{array} \right\}, \quad \underline{u} \in \{u, v\}. \quad (20)$$

The sign conditions of the control law will be checked for special cases.

In the following, we shall compare the solution for the Gompertz type growth function from [35] with solutions for the logistic growth function which turn out to be much simpler. We take the following initial conditions and control bounds for u ,

$$p(0) = 12000, \quad q(0) = 15000, \quad u_{\max} = 75, \quad w_{\max} = 300,$$

and discuss two cases of control bounds for v ;

$$\text{Case I : } v_{\max} = 1, \quad z_{\max} = 2; \quad \text{Case II : } v_{\max} = 2, \quad z_{\max} = 10.$$

3.1 Optimal control solution without delays for free final time T and Gompertz type growth function.

Since the terminal time T is free, the *singular control* u can be obtained in feedback form [29],

$$u = u_{\text{sing}}(p, q) = \frac{1}{\gamma} \left((\xi \ln \left(\frac{p}{q} \right) + b \frac{p}{q} + \frac{2}{3} \xi \frac{d}{b} \frac{q}{p^{1/3}} - (\mu + q^{2/3})) \right) + \frac{\varphi - \eta}{\gamma} v_c \quad (21)$$

provided that the control $v(t) = v_c$ is constant on a singular arc of u .

Case I :

The optimal controls have the following structure:

$$(u(t), v(t)) = \left\{ \begin{array}{ll} (u_{\max}, 0) & \text{for } 0 \leq t < t_1 \\ (u_{\text{sing}}(p(t), q(t)), 0) & \text{for } t_1 \leq t < t_2 \\ (u_{\text{sing}}(p(t), q(t)), v_{\max}) & \text{for } t_2 \leq t \leq t_3 \\ (0, v_{\max}) & \text{for } t_3 < t < T \end{array} \right\}. \quad (22)$$

The control $u(t)$ is a bang-singular-bang control, whereas $v(t)$ is a bang-bang control with only one switch at t_2 . Assuming that the control structure is known, we can directly optimize the switching times t_1, t_2, t_3 and the terminal time T to minimize the terminal value $p(T)$. Using the arc-parametrization method and the optimal control package NUDOCSS, we get the following numerical results:

$$p(T) = 7009.26, \quad q(T) = 7291.72, \quad T = 6.6713, \\ t_1 = 0.090503, \quad t_2 = 4.6713, \quad t_3 = 6.5090.$$

The numerical results are slightly different from those in [35], since here $\eta = 0.1$ whereas $\eta = 0$ in [35]. Note that second-order sufficient conditions [34] hold for

the Induced Optimization Problem with respect to the switching times t_1, t_2, t_3 and final time T . The optimal control (u, v) and the state variables (p, q) are shown in Figure 1.

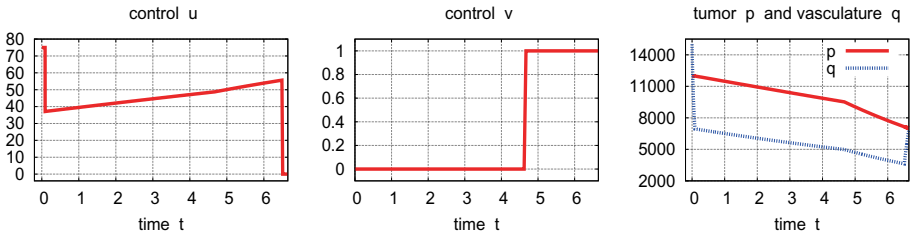


Fig. 1. Optimal solution for the Hahnfeldt model with Gompertz type growth function $f(p, q) = -\xi p \ln(p/q)$. Initial conditions $p(0) = 12000, q(0) = 15000$ and control bounds $u_{\max} = 75, w_{\max} = 300, v_{\max} = 1, z_{\max} = 2$. (a) control u , (b) control v , (c) tumor volume p and vasculature q .

Case II :

We get the same control structure as in (22).

The direct optimization of the switching times t_1, t_2, t_3 and the terminal time t_f via the arc-parametrization method and the control package NUDOCSS yields

$$p(T) = 3260.94, \quad q(T) = 3888.87, \quad T = 6.1441, \\ t_1 = 0.090503, \quad t_2 = 1.1441, \quad t_3 = 5.9826.$$

The optimal control (u, v) and state variables (p, q) are displayed in Figure 2.

Approximative control

We approximate the optimal control by the following simpler control with piecewise constant values,

$$(u(t), v(t)) = \left\{ \begin{array}{ll} (u_{\max}, 0) & \text{for } 0 \leq t < t_1 \\ (u_c, 0) & \text{for } t_1 \leq t < t_2 \\ (u_c, v_{\max}) & \text{for } t_2 \leq t \leq t_3 \\ (0, v_{\max}) & \text{for } t_3 < t < T \end{array} \right\} \quad (23)$$

Where switching times and terminal time are fixed to

$$t_1 = 0.1, \quad t_2 = 1.0, \quad t_3 = 5.85, \quad T = 6.0.$$

The control $u(t)$ has the constant value $u(t) = u_c$ on the interval $[t_1, t_3]$ which is a rather crude approximation of the singular control in Figure 2. Using again

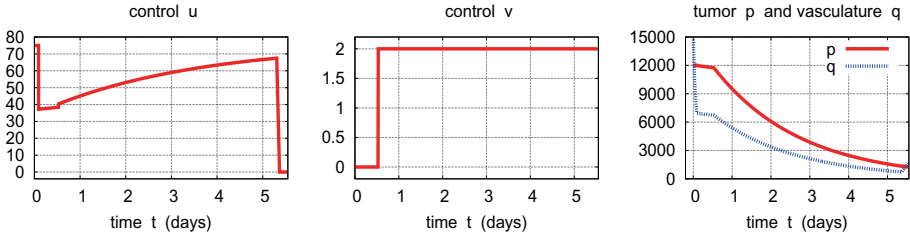


Fig. 2. Optimal solution for the Hahnfeldt model with Gompertz type growth function $f(p, q) = -\xi p \ln(p/q)$. Initial conditions $p(0) = 12000$, $q(0) = 15000$ and control bounds $u_{\max} = 75, w_{\max} = 300, v_{\max} = 2, z_{\max} = 10$. (a) control u , (b) control v , (c) tumor volume p and vasculature q .

the arc-parametrization method and the control package NUDOCSS, we get the following numerical results,

$$p(T) = 3268.73, q(T) = 3981.37, u_c = 50.878,$$

The optimal controls (u, v) and the state variables (p, q) are shown in Figure 3. Though (23) is only a crude approximation of the optimal control structure (22), the functional value $J(u) = p(T) = 3268.7$ differs not very much from the optimal value $J(u) = 3260.9$.

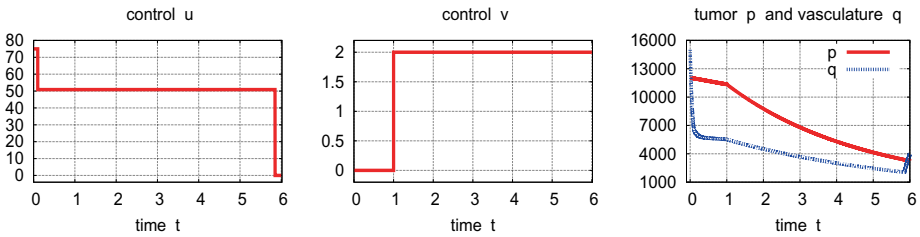


Fig. 3. Approximate control (23) for the Hahnfeldt model with Gompertz growth $F(p, q) = -\xi p \ln(p/q)$. Initial conditions $p(0) = 12000$, $q(0) = 15000$ and control bounds $u_{\max} = 75, w_{\max} = 300, v_{\max} = 2, z_{\max} = 10$. (a) control u , (b) control v , (c) tumor volume p and vasculature q .

3.2 Optimal control solution without delays for free final time and logistic type growth function

Since the terminal time T is free, it would be possible to obtain a formula of the singular control in feedback form. However, since we only find bang-bang controls for a logistic type growth function as predicted in [16], we omit such a formula.

Case I:

Solving the discretized control problem with $N = 10000$ by optimization code IPOPT, we find the following control structure for the control $(u(t), v(t))$:

$$(u(t), v(t)) = \begin{cases} (u_{\max}, 0) & \text{for } 0 \leq t < t_1 \\ (u_{\max}, v_{\max}) & \text{for } t_1 \leq t \leq t_2 \\ (0, v_{\max}) & \text{for } t_2 < t < T \end{cases} \quad (24)$$

Both controls u and v are bang-bang with only one switch at t_1 , respectively, at t_2 .

The optimal value $p(T) = 5887.9$ is much smaller than the value $p(T) = 7009.26$ for the Gompertz growth function. This is due to the fact that the logistic growth function produces a larger decrease in the tumor volume $p(t)$. The optimal control (u, v) and the state variables (p, q) are shown in Figure 4.

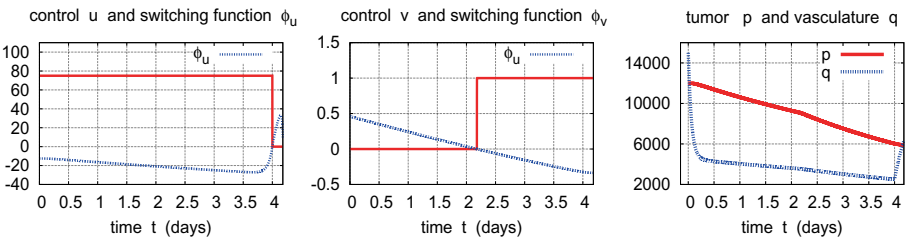


Fig. 4. Optimal solution for the Hahnfeldt model with logistic type growth function $f(p, q) = \xi p(1 - p/q)$. Initial conditions $p(0) = 12000$, $q(0) = 15000$ and control bounds $u_{\max} = 75$, $w_{\max} = 300$, $v_{\max} = 1$, $z_{\max} = 2$. (a) control u , (b) control v , (c) tumor volume p and vasculature q .

Case II :

The discretization approach with $N = 10000$ grid points and the optimization code IPOPT yield the following control structure for the control $(u(t), v(t))$:

$$(u(t), v(t)) = \begin{cases} (0, v_{max}) & \text{for } 0 \leq t < t_1 \\ (u_{max}, v_{max}) & \text{for } t_1 \leq t < t_2 \\ (0, v_{max}) & \text{for } t_2 < t < T \end{cases} \quad (25)$$

The direct optimization of the switching times t_1, t_2 and the terminal time T via the arc-parametrization method and the control package NUDOCCS yields

$$p(T) = 2840.69, q(T) = 5561.26, T = 5.000, \\ t_1 = 0.566, \quad t_2 = 4.566.$$

Again, the optimal value $p(T) = 2840.69$ is much smaller than that for the Gompertz type growth function. The optimal control (u, v) and the state variables (p, q) are displayed in Figure 5.

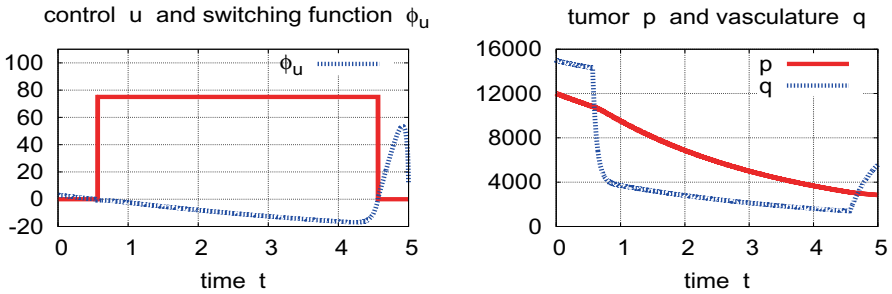


Fig. 5. Optimal solution for the Hahnfeldt model with logistic type growth $f(p, q) = \xi p(1 - p/q)$. Initial conditions $p(0) = 12000, q(0) = 15000$ and control bounds $u_{max} = 75, w_{max} = 300, v_{max} = 2, z_{max} = 10$. Optimal control $v(t) \equiv 2$. (a) control u and switching function ϕ_u satisfying the control law (20), (b) tumor volume p and vasculature q .

We note that the solution displayed in Figure 5 satisfies the second-order sufficient conditions (SSC) in [36], Chapter 7, since SSC hold for the Induced Optimization Problem with respect to t_1, t_2 and the strict bang-bang property holds:

$$\phi_u(t) > 0 \forall 0 \leq t < t_1, \dot{\phi}_u(t_1) < 0, \phi_u(t) < 0 \forall t_1 \leq t < t_2, \dot{\phi}_u(t_2) > 0, \\ \phi_u(t) > 0 \forall t_2 \leq t < T.$$

3.3 Optimal control solution without delays for fixed final time $T = 16$

In this section, we consider only the logistic type growth function in the dynamics.

In view of the rather large delay $h_1 = 10.6$ in the control u , we choose the final time $T = 16 > h_1$. We use the discretization approach to compute the solutions both for the non-delayed and delayed optimal control problem. We shall keep the initial conditions

$$p(0) = 12000, \quad q(0) = 15000$$

as in the previous sections but choose different control bounds to accommodate to the large time horizon:

$$u_{\max} = 40, \quad y_{\max} = 320, \quad v_{\max} = 2, \quad z_{\max} = 10.$$

The control structure is given by

$$(u(t), v(t)) = \left\{ \begin{array}{ll} (0, 0) & \text{for } 0 \leq t < t_1 \\ (u_{\max}, 0) & \text{for } t_1 \leq t < t_2 \\ (u_{\max}, v_{\max}) & \text{for } t_2 \leq t < t_3 \\ (0, v_{\max}) & \text{for } t_3 < t < T \end{array} \right\} \quad (26)$$

and the numerical results are

$$p(T) = 3254.42, \quad q(T) = 5527.80, \quad T = 16, \\ t_1 = 7.712, \quad t_2 = 11.00 \quad t_3 = 15.712.$$

Note that initial bang-bang arc with $u(t) = u_{\max}$ on $[0, 0.08]$ is very small.

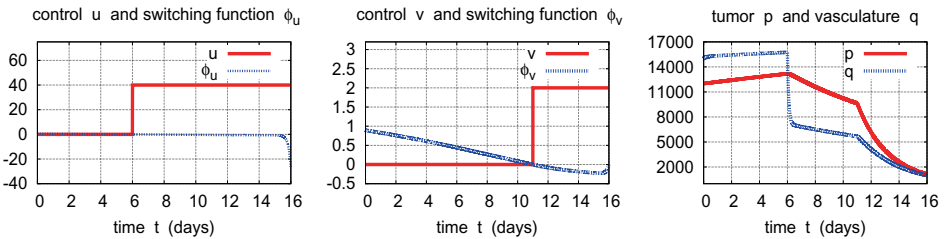


Fig. 6. Optimal solution for the Hahnfeldt model with logistic type growth function $f(p, q) = \xi p(1 - p/q)$ and fixed final time $T = 16$. Initial conditions $p(0) = 12000$, $q(0) = 15000$ and control bounds $u_{\max} = 40, w_{\max} = 320, v_{\max} = 2, z_{\max} = 10$. (a) anti-angiogenic control u and switching function ϕ_u satisfying the control law (20), (b) control v and switching function ϕ_v satisfying the control law (20), (c) tumor volume p and vasculature q .

3.4 Optimal control solution with fixed final time $T = 16$ and delays $h = 1.84$ in v and $h_1 = 10.6$ in u

The discretization approach with $N = 10000$ grid points yields the more complicated control structure shown in Figure 7. The control u has four bang-bang

arcs, whereas v has three bang-bang arcs. We obtain the numerical results

$$p(T) = 2733.44, \quad q(T) = 3856.97, \quad T = 16,$$

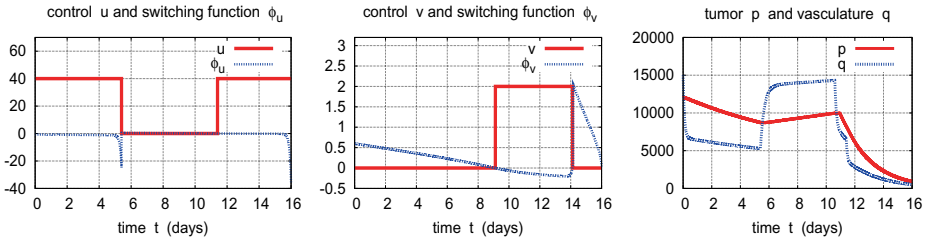


Fig. 7. Optimal solution for the Hahnfeldt model with logistic type growth with delays $h = 1.84$, $h_1 = 10.6$ and fixed final time $T = 16$. Initial conditions $p(0) = 12000$, $q(0) = 15000$ and control bounds $u_{\max} = 40$, $w_{\max} = 320$, $v_{\max} = 2$, $z_{\max} = 10$. (a) control u and switching function ϕ_u satisfying the control law (20), (b) control v and switching function ϕ_v satisfying the control law (20), (c) tumor volume p and vasculature q .

4 Discussion and Conclusions

This study was inspired by recently reported results of clinical trials with two angiogenic inhibitors characterized by different half-lives combined with chemotoxic agents [37]. In the paper we propose to describe the effects of the combined therapy by a two-compartmental model with two control variables with multiple delays which represent the differences in pharmacokinetics of different agents and different goals of the therapy. While the primary goal is related to eradication of tumor or at least survival benefits the secondary one is to normalize vasculature thereby facilitating chemotoxic drug delivery. It leads to a complex multi-control problem, complete solution of which is much more complicated than in the single control case. We have formulated an optimal control problem to ensure driving the dynamical system to a neighborhood of the required final state. We have used necessary conditions of optimality for systems with delays in control and constrains imposed on control and state variables and then, the optimal control was numerically found using a two-stage computational algorithm. The first stage is based on the large scale non-linear programming for a discretized version of the optimal control problem and the second one is related to switching times optimization by the arc-parametrization method. We have analyzed how our results are sensitive for a choice of type of growth equation used to model tumor dynamics. The Gompertz type growth which was the most

often used in modeling tumor growth is not mandatory to describe the unperturbed tumor growth slowdown with size observed in clinical and experimental data [20]. Its drawback is that for the small ratios of tumor volume and vascular carrying capacity the relative tumor growth capacity is unbounded. This feature is absent in the case when logistic type growth is used. Yet another advantage of using this model is absence of singular arcs in optimal treatment protocols of antiangiogenic treatment which are present in the case of Gompertz type growth function is used. In our study we have found that this property is true also in the case when two control variables (representing two anticancer modalities) with multiple delays are considered in the model. In this sense the optimal control problem is structurally sensitive, the use of Gompertz type growth leads to optimal controls with singular intervals (which are practically unrealizable) and the logistic type growth to pure bang-bang control. Moreover in the case of logistic growth we have analyzed both cases with free and fixed terminal time. Introduction of multiple delays in control variables in the models has led to some changes in understanding and checking conditions of optimality, and their numerical computation.

References

1. D. Hanahan, R.A. Weinberg, *Hallmarks of Cancer: The Next Generation*, Cell, **144** (2011), 647-670, 2011.
2. J. Folkman, *Anti-Angiogenesis: New Concept for Therapy of Solid Tumors*, Annals of Surgery, **175**, (1972), 409-416.
3. A.C. Billioux, U. Modlich, R. Bicknell, *The Cancer Handbook: Angiogenesis*, 2nd Edition, John Wiley & Sons, 2007.
4. V.T. Devita, J.Folkman, *Cancer: Principles and Practice of Oncology*, 6th edition, Lippincott Williams & Wilkins Publishers, 2001.
5. R.S. Kerbel, *Inhibition of tumor angiogenesis as a strategy to circumvent acquired resistance to anti-cancer therapies agent*, BioEssays, **13** (1991), 31-36.
6. G. Bergers and D. Hanahan, *Modes of resistance to antiangiogenic therapy*, Nature Reviews Cancer, **8**, (2008), 592-603.
7. G. Gasparini, R. Longo, M. Fanelli, and B. A. Teicher, *Combination of antiangiogenic therapy with other anticancer therapies: results, challenges, and open questions*, Journal of Clinical Oncology, **23**, (2005), 1295-1311.
8. L. S. Teng, K. T. Jin, K. F. He, H. H. Wang, J. Cao, and D. C. Yu, *Advances in combination of antiangiogenic agents targeting VEGF-binding and conventional chemotherapy and radiation for cancer treatment*, Journal of the Chinese Medical Association, **73**, (2010), 281-288.
9. J. Ma and D. J.Waxman, *Combination of antiangiogenesis with chemotherapy for more effective cancer treatment*, Molecular Cancer Therapeutics, **7**, (2008), 3670-3684.
10. A. d'Onofrio and A. Gandolfi, *Chemotherapy of vascularised tumours: role of vessel density and the effect of vascular "pruning"*, Journal of Theoretical Biology, **264**, (2010), 253-265.
11. US National Institutes of Health, Clinical Trials, 2012, <http://www.clinicaltrials.gov/>

12. M. Kimmel and A. Swierniak, *Control Theory Approach to Cancer Chemotherapy: Benefiting from Phase Dependence and Overcoming Drug Resistance*, *Tutorials in Mathematical Biosciences III: Cell Cycle, Proliferation, and Cancer* (A. Friedman-Ed.), *Lecture Notes in Mathematics, Mathematical Biosciences Subseries*, 1872, Springer, Heidelberg, (2006), 185-202
13. P. Hahnfeldt, D. Panigrahy, J. Folkman, and L. Hlatky, *Tumor development under angiogenic signaling: a dynamical theory of tumor growth, treatment response, and postvascular dormancy*, *Cancer Research*, **59**, (1999), 4770-4775.
14. A. Swierniak, *Direct and indirect control of cancer populations*, *Bulletin of the Polish Academy of Sciences: Technical Sciences*, **56**, (2008), 367-378, 2008.
15. A. d'Onofrio and A. Gandolfi, *Tumour eradication by antiangiogenic therapy: analysis and extensions of the model by Hahnfeldt et al. (1999)*, *Mathematical Biosciences*, **191**, (2004), 159-184.
16. A. Swierniak, *Comparison of six models of antiangiogenic therapy*, *Applicationes Mathematicae*, **36**, (2009), 333-348.
17. A. Ergun, K. Camphausen, and L. M. Wein, *Optimal scheduling of radiotherapy and angiogenic inhibitors*, *Bulletin of Mathematical Biology*, **65**, (2003), 407-424.
18. M. Dolbniak, A. Swierniak, *Comparison of simple models of periodic protocols for combined anticancer therapy*, *Computational and Mathematical Methods in Medicine*, 2013, Article ID 567213, doi: 11.1055/2013/567213.
19. A. Swierniak, J. Klamka, *Local controllability of models of combined anticancer therapy with delays in control*, *Math. Model. Nat. Phenom.*, **9**, (2014), 216-226.
20. R.K. Saches, L.R. Hlatky and P.Hahnfeldt, *Simple ODE models of tumor growth and antioangiogenic or radiation treatment*, *Math. Comput. Mod*, **33**, (1998), 1297-1304.
21. U. Ledzewicz and H. Schaettler, *Antiangiogenic therapy in cancer treatment as an optimal control problem*, *SIAM Journal on Control and Optimization*, **46**, (2007), 1052-1079.
22. U. Ledzewicz and H. Schaettler, *Analysis of optimal controls for a mathematical model of tumour anti-angiogenesis*, *Optimal Control Applications and Methods*, **29**, (2008), 41-57.
23. A. Swierniak, A. d'Onofrio, and A.Gandolfi, *Control problems related to tumor angiogenesis*, in: *Proc. of the 32nd Annual Conference on IEEE Industrial Electronics (IECON 2006)*, Paris, 677-681, November 2006.
24. U. Ledzewicz and H. Schaettler, *On the optimality of singular controls for a class of mathematical models for tumor antiangiogenesis*, *Discrete and Continuous Dynamical Systems, Series B*, **11**, (2009), 691-715.
25. U. Ledzewicz and H. Schaettler, *Optimal and suboptimal protocols for a class of mathematical models of tumor anti-angiogenesis*, *J. Theor. Biol.*, **252**, (2008), 295-301.
26. U. Ledzewicz, J. Marriott, H. Maurer, and H. Schaettler, *Realizable protocols for optimal administration of drugs in mathematical models for anti-angiogenic treatment*, *Mathematical Medicine and Biology*, **27**, (2010), 157-179.
27. U. Ledzewicz, H. Maurer, and H. Schaettler, *Minimizing tumor volume for a mathematical model of anti-angiogenesis with linear pharmacokinetics*, in: *Recent Advances in Optimization and its Applications in Engineering*, pp. 267-276, Springer, 2010.
28. A. Swierniak, *Modelling combined antiangiogenic and chemo-therapies*, in: *Proc. 14th National Conf. Appl. Math. Biol Medicine*, Leszno, 2008, 127-133.
29. A. d'Onofrio, U. Ledzewicz, H. Maurer, and H. Schaettler, *On optimal delivery of combination therapy for tumors*, *Math. Biosciences*, **222**, (2009), 13-26.

30. J. Klamka, H. Maurer, A. Swierniak, *Local controllability and optima Control for a model of combined anticancer therapy with Control delays*, Mathematical Biosciences and Engineering, **14**, 1, (2016).
31. R. Fourer, D.M. Gay and B.W. Kernighan, *AMPL: A Modeling Language for Mathematical Programming*, Duxbury Press, Brooks–Cole Publishing Company, 1993.
32. A. Wächter, and L.T. Biegler, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*, Mathematical Programming, **106** (2006), 25-57; cf. Ipopt home page (C. Laird and A. Wächter): <https://projects.coin-or.org/Ipopt>.
33. C. Büskens and H. Maurer, *SQP methods for solving optimal control problems with control and state constraints : adjoint variables, sensitivity analysis and real time control*, J. Comput. Appl. Math., **120**, (2000), 85-99.
34. H. Maurer, C. Büskens, J.H.R. Kim, and C.Y. Kaya, *Optimization methods for the verification of second order sufficient conditions for bang-bang control*, Optimal Control Appl. Meth., **26**, (2005), 129-156.
35. U. Ledzewicz, H. Maurer, and H. Schättler, *Optimal and suboptimal protocols for a mathematical model for tumor antiangiogenesis in combination with chemotherapy*, Mathematical Biosciences and Engineering, **8**, (2011), 307-323.
36. L. Göllmann and H. Maurer, *Theory and applications of optimal control problems with multiple time-delays*, Special Issue on Computational Methods for Optimization and Control, J. of Industrial and Management Optimization, **10**, No.2, (2014), 413-441.
37. US National Institutes of Health, Clinical Trials, (2014), <http://www.clinicaltrials.gov>

Hybrid Newton Observer in Analysis of Glucose Regulation System for ICU Patients

Jerzy Baranowski, Piotr Bania, Waldemar Bauer, Jędrzej Chiliński, and Paweł Piątek

AGH University of Science and Technology, Krakow, Poland,
jb@agh.edu.pl

Abstract. Glucose regulation system is one of the most important problems in ICUs. Nowadays either frequent testing and insulin shots or automatic insulin pumps are used to avoid hyperglycemia even for patients without history of diabetes. As it is impossible to directly measure the glucose level, there is a need for estimation tools ensuring appropriate quality of measurements.

In this paper the application of Hybrid Newton Observer is proposed. We use Intensive Care Units Minimal Model in order to estimate the value of sugar level. Three possible approaches to calculating the Jacobian are considered as this task consists the main issue in Newton observer implementation. The analysis is then confirmed with simulation.

Keywords: state estimation, Newton observer, jacobian, variational equation, adjoint equations, glycemia

1 Introduction

One of the most important problems in ICUs is the control of blood glucose level of patients. Many afflictions like heart attack or multi-organ disorder cause sudden rises in the glucose level. This situation is very dangerous because it increases the risk of infection, hampers blood coagulation and disturbs the metabolic balance. Studies have shown that the rigorous blood glucose level control substantially reduces the mortality rate in ICUs.

Model of blood sugar level of ICU patient was described by van Herpe et al. (see [26, 27]). The mathematical model of dynamic blood glucose level was considered, among the others, by Bergman et al. (see [4]), Kovács (see [16]), Makroglou et al. (see [19]) and Li and Kuang (see [18]). The nonlinear fractional order model of blood glucose-insulin level was discussed by N'Doye et al. (see [5, 21]). This model is an extension of so called minimal model developed by Bergman et al. in 1981 [4] for glucose metabolism modeling after intravenous glucose tolerance test, it was then adapted by Furler et al. in 1985 [12] to represent the diabetic state. This version, for purposes of Intensive Care Unit patient analysis, was developed by Van Herpe et al. in 2007 [26] and is called ICU-MM. More details regarding development of minimal model can be found in [16, 19] and [9]. For different approaches to glucose dynamics modeling see [10]

Control of blood sugar level was analyzed by many researchers focusing on different approaches. A closed loop control of blood glucose was considered among the others by Chee and Fernando (see [9]). The control of blood sugar level in the critically ill was presented by Haverbeke (see [14]) and Hovorka (see [15]). The fuzzy PID controller was also discussed by Li et al. (see [17]) and Mohamed et al. (see [1]). Non-integer controller was analyzed by Bauer et. al [3] and also see [11, 20, 23–25].

In this paper we discuss an application of Hybrid Newton Observer. This observer was proposed by Bıyık, and Arcaç [6, 7]. It was applied by authors to multiple problems, including hydraulic systems [2].

Rest of the paper is organized as follows. In the first section, the ICU-MM is described. This section includes also a brief description of model properties. In another section Hybrid Newton Observer is presented. Next section includes general remarks on Jacobian implementation and presents different methods of its implementation: finite difference approximation, application of variational equation and application of the Pontryagin's maximum principle. As an example of implementation, the results of estimated state of ICU-MM are presented. Paper ends with conclusions and some propositions on further works.

2 Mathematical model

We consider a so called ICU-MM (Intensive Care Unit Minimal Model), given by following four, nonlinear ordinary differential equations

$$\begin{aligned}\dot{G}(t) &= P_1(G_b - G(t)) - X(t)G(t) + \frac{F_G(t)}{V_G} \\ \dot{X}(t) &= -P_2X(t) + P_3(I_1(t) - I_b) \\ \dot{I}_1(t) &= \alpha \max(0, I_2(t)) - n(I_1(t) - I_b) + \frac{F_I(t)}{V_I} \\ \dot{I}_2(t) &= \beta\gamma(G(t) - h) - nI_2(t)\end{aligned}\tag{1}$$

where G and I_1 are the glucose and the insulin concentrations in the blood plasma. The variable X describes the effect of insulin on net glucose disappearance. The variable I_2 does not have a strictly defined clinical interpretation and was introduced for mathematical reasons, as a way of modeling the functional pancreatic insulin system (as usually ICU patients are not diabetic).

The parameters G_b and I_b denote the basal value of plasma glucose and plasma insulin, respectively. The model consists of two input variables: the exogenous insulin flow (F_I) and the carbohydrate (glucose) calories flow (F_G), both administered intravenously. The glucose distribution space and the insulin distribution volume are denoted as V_G and V_I , respectively. The coefficient P_1 represents the glucose effectiveness (i.e. the fractional clearance of glucose) when insulin remains at the basal level; P_2 and P_3 are the fractional rates of net remote insulin disappearance and insulin-dependent increase, respectively. Endogenous insulin is represented as the insulin flow that is released in proportion (by γ)

to the degree by which glycemia exceeds a glucose threshold level h that is corresponding to the normal glucose level of 7.5 mmol/dl. The time constant for insulin disappearance is denoted as n . In order to keep the correct units, an additional model coefficient, $\beta = 1$ min, is added. The coefficient α is a scaling factor for the second insulin variable I_2 . For more details see [26]. The model (1) is slightly modified in order to improve the clarity of equations (all parameters are now positive and grouping in first equation was changed). Parameters of the model are can be found in the work of Van Herpe et. al. [26].

Because of all the variables of (1) only glucose level is easily measured, other state variables have to be estimated.

3 Hybrid Newton Observer

The concept of hybrid Newton observer relies on an obvious fact that if a solution of differential equation, for x_0 initial condition takes form

$$x(t) = F(x_0, t) \quad (2)$$

then the output can be expressed as

$$y(t) = h(F(x_0, t)) \quad (3)$$

Knowing the system output values at different time instances one can form a system of nonlinear equations:

$$\begin{aligned} y(t_k) &= h(F(w_k, t_k)) \\ y(t_{k+1}) &= h(F(w_k, t_{k+1})) \\ &\vdots \\ y(t_{k+n-1}) &= h(F(w_k, t_{k+n-1})) \end{aligned} \quad (4)$$

One of the popular methods of numerical solutions for solving systems of algebraic equations is the Newton method (and its modifications, for examples see [22]). Because (2) is not given in analytic form it cannot be directly applied. However, Bıyık and Arcak (see [7]) proposed to determine solution (2) and in consequence (4) numerically through simulation. In such case it takes the following form. Let us denote

$$H(w_k) = \begin{bmatrix} h(w_k) \\ h(F(w_k, t_{k+1})) \\ \vdots \\ h(F(w_k, t_{k+n-1})) \end{bmatrix} \quad (5)$$

and

$$Y_k = \begin{bmatrix} y(t_k) \\ y(t_{k+1}) \\ \vdots \\ y(t_{k+n-1}) \end{bmatrix}$$

then the Newton method takes form (superscript is the Newton’s method iteration number)

$$\begin{aligned}
 w_k^{i+1} &= w_k^i + s_i \\
 \frac{d}{dw_k} H(w_k^i) s_i &= Y_k - H(w_k) \\
 i &= 0, 1, \dots, d
 \end{aligned}
 \tag{6}$$

Because we cannot iterate till convergence, the number ε of Newton iterations in every estimation step k is a chosen a priori design parameter.

4 Jacobian

Generally the only problem with application of Newton observer is computation of Jacobian of system of equations (4). In order to compute Jacobian, essentially one needs to compute derivatives of differential equation solutions regarding initial conditions. These gradients can be computed by:

- Forward difference approximations
- Variational equation
- Via a formula based on proof of Maximum principle

In following subsections these approaches will be described and briefly compared.

4.1 Finite difference approximation

The most basic way of determination of Jacobian is approximating it by finite differences

$$\frac{\partial H}{\partial w}(w_k^i) \approx \left[\begin{array}{c} \frac{H(w_k^i + e_1 \rho) - H(w_k^i)}{\rho} \\ \frac{H(w_k^i + e_2 \rho) - H(w_k^i)}{\rho} \\ \vdots \\ \frac{H(w_k^i + e_n \rho) - H(w_k^i)}{\rho} \end{array} \right]^T
 \tag{7}$$

where ρ is a small parameter close to zero, value of which should be determined in accordance to time scale, parameter values and dynamics of the model.

4.2 Application of variational equation

Much more sophisticated approach is to compute the Jacobian exactly. Let us consider a single equation of (4)

$$h(F(w_k, t_{k+l})) - y(t_{k+l}) = 0
 \tag{8}$$

Derivative of left hand side of (8) is given by

$$\begin{aligned} \nabla_{w_k}(h(F(w_k, t_{k+l})) - y(t_{k+l})) &= & (9) \\ &= \nabla_{w_k}(h(F(w_k, t_{k+l}))) = \end{aligned}$$

$$= \left(\frac{\partial F(w_k, t_{k+l})}{\partial w_k} \right)^T \nabla_x h(F(w_k, t_{k+l})) \tag{10}$$

in order to apply (10) one needs to determine derivative of (2) with respect to initial condition. In order to do so one needs to solve a so called *variational equation* (see [13]). Variational equation has form

$$\dot{\Psi}(t) = J(x(t))\Psi(t) \tag{11}$$

$$\Psi(t_k) = I \tag{12}$$

$$x(t) = F(w_k, t) \tag{13}$$

$$t \in [t_k, t_{k+n-1}] \tag{14}$$

where $J(x(t))$ is Jacobian matrix of right hand side of differential equation ???. Solution of variational equation has an important property, that is

$$\frac{dF(w_k, \tau)}{dw_k} = \Psi(\tau) \quad \forall \tau \in [t_k, t_{k+n-1}] \tag{15}$$

In other words solution of variational equation at τ determines the derivative of differential equation solution with respect to initial condition at τ . It should be noted that for linear systems $\dot{x}(t) = Ax(t)$ we get $\Psi(t) = e^{At}$.

Using solution of variational equation (11) and formula (10) we get the expressions for consecutive rows of Jacobian matrix:

$$\nabla_{w_k}(h(F(w_k, t_{k+l})))^T = \nabla_x h(F(w_k, t_{k+l}))^T \Psi(t_{k+l})$$

Jacobian matrix has the form

$$\frac{d}{dw_k} H(w_k) = \begin{bmatrix} \nabla_x h(F(w_k, t_k))^T \Psi(t_k) \\ \nabla_x h(F(w_k, t_{k+1}))^T \Psi(t_{k+1}) \\ \vdots \\ \nabla_x h(F(w_k, t_{k+n-1}))^T \Psi(t_{k+n-1}) \end{bmatrix} \tag{16}$$

4.3 Application of the Pontryagin’s Maximum Principle

This approach (PMP) actually does not uses the Maximum Principle, but is based on its proof (see [8]). Let us consider some performance indicator Q and its increment generated by change of initial condition. Specifically we will define

$$q(x(T)) = h(F(w_k, T)) \tag{17}$$

$$\dot{x}^* = f(x^*) + g(x^*)u \tag{18}$$

$$\Delta x = x^* - x \tag{19}$$

$$Q(x_0) = q(x(T)) \tag{20}$$

along with a Hamiltonian and adjoint equations

$$\mathcal{H}(\psi, x, u) = \psi^\top (f(x) + g(x)u) \tag{21}$$

$$\dot{\psi} = -\nabla_x \mathcal{H}(\psi, x, u) \tag{22}$$

$$\psi(T) = -\nabla q(x(T)) \tag{23}$$

Now we can analyse the increment of performance indicator

$$\begin{aligned} \Delta Q &= q(x^*(T)) - q(x(T)) = \\ &= \nabla q(x(T))\Delta x(T) + o(\Delta x(T)) = \\ &= -\psi(T)\Delta x(T) + o(\Delta x(T)) \end{aligned}$$

We will now make use of the fact, that

$$\psi(T)\Delta x(T) = \psi(0)\Delta x(0) + \int_0^T \dot{\psi}^\top \Delta x dt + \int_0^T \psi^\top \Delta \dot{x} dt$$

this relation can be used as follows

$$\begin{aligned} \Delta Q &= -\psi(0)\Delta x(0) - \int_0^T \dot{\psi}^\top \Delta x dt - \int_0^T \psi^\top \Delta \dot{x} dt + o(\Delta x(T)) = \\ &= \int_0^T \nabla_x \mathcal{H}(\psi, x, u)^\top \Delta x dt + \\ &\quad - \int_0^T \psi^\top (f(x^*) + g(x^*)u - f(x) + g(x)u) dt \\ &\quad - \psi(0)\Delta x(0) + o(\Delta x(T)) \end{aligned}$$

We have used the definitions adjoint equations (22). Now we will use the definition of Hamiltonian, and the definition of Frechet derivative

$$\begin{aligned} \Delta Q &= -\psi(0)\Delta x(0) - \int_0^T -\nabla_x \mathcal{H}(\psi, x, u)^\top \Delta x dt \\ &\quad - \int_0^T (\mathcal{H}(\psi, x + \Delta x, u) - \mathcal{H}(\psi, x, u)) dt + o(\Delta x(T)) = \\ &= -\psi(0)\Delta x(0) - \int_0^T o(\Delta x(t)) dt + o(\Delta x(T)) \end{aligned}$$

which finally leads to formula for increment

$$\Delta Q = -\psi(0)\Delta x(0) + o(\|\Delta x\|) \tag{24}$$

which, after taking a limit, becomes a formula for gradient of Q with respect to x_0

$$\nabla_{x_0} Q = -\psi(0) \tag{25}$$

With this computation we get another formula for Jacobian

$$\frac{d}{dw_k} H(w_k) = -[\psi^{t_k}(0) \ \psi^{t_{k+1}}(0) \ \dots \ \psi^{t_{k+n-1}}(0)]^\top$$

where ψ^τ denotes the adjoint variable solved on the interval $[t_k, \tau]$

4.4 Comparison of approaches

Those methods differ from each other, both in terms of their performance and computational complexity. Finite differences require solving system of n equations n times on the whole $[t_k, t_{k+n-1}]$ interval. It is very dependent on ρ . Variational equation approach requires solution of n^2 equations once on the $[t_k, t_{k+n-1}]$ interval and is the easiest to implement. This method was chosen for further analysis. PMP approach requires solving n systems of n equations backwards, but on different integration intervals. Theoretically it is least computationally expensive.

5 Simulations of the observer

We analysed operation of the hybrid Newton observer through numerical experiments. In order to perform simulations, the following assumptions were taken:

- Measurements were taken every 15 minutes.
- Because observability is only local, reasonable initial guess was taken in order to obtain convergence.
- Glucose evolution was based on natural reactions of body - no insulin was given (see figure 1).

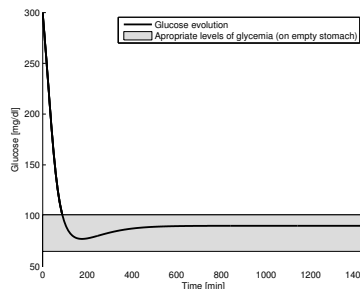


Fig. 1. Glucose evolution in 24 hour period

Jacobians were computed with the application of variational equation solved simultaneously with model equations. Example of Newton iterations are given in table 1.

As it can be seen in table 1, exact estimation is very quick. In further iterations initial guess is much better so only minimal corrections occur. Because system loses part of observability (of I_2) as it approaches the steady state, appropriate measures were taken to avoid this problem. Loss of observability happens

Table 1. Example of Newtonian iterations

x_0	w_0^1	w_0^2	w_0^3	w_0^4	w_0^5
300	200	300	300	300	300
0.0005	0.01	-0.0033	0.0004	0.0005	0.0005
58	56	-57.7166	60.617	58.0021	58
1.5600	-1.8966	20.6963	1.6996	1.5600	1.5600

when I_2 becomes negative and stops its effect on the system, so it cannot be estimated from the output. Figures 2 to 3 present the estimates and states at initial stage of glucose evolution. Because estimation is exact only values of Newton corrections are marked, true state and estimate overlap.

6 Conclusions and further work

Hybrid Newton observer was shown to be an excellent tool for state estimation in the ICU-MM model. With reasonable initial guess it showed convergence and found the exact solution. Also it managed to perform correctly regardless of unobservability of system in certain areas of state space. Further works should include a systematic comparison of Jacobian computation efficiency for different approaches, especially for large systems. Possible improvements of the general method are matter of discussion. It is possible to replace (6) with a variant of Newton-Raphson method, but the result would be a simplified, perhaps less effective version of Moving Horizon Estimation with a similar computation cost.

Work funded by AGH Statutory funds 11.11.120.396.

References

1. Al-Fandi, M., Jaradat, M.A., Sardahi, Y.: Optimal PID-Fuzzy Logic Controller for type 1 diabetic patients. In: International Symposium on Mechatronics and Its

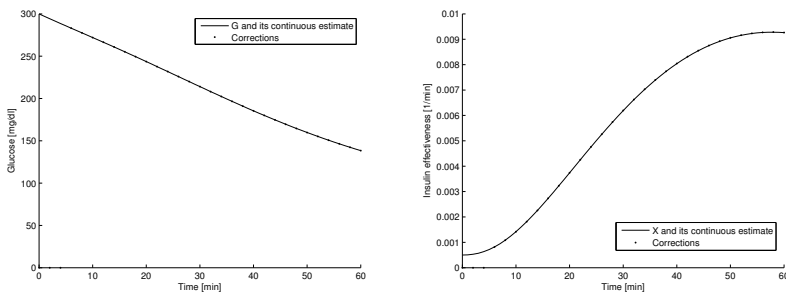


Fig. 2. Glucose and insulin effectiveness evolution and estimates

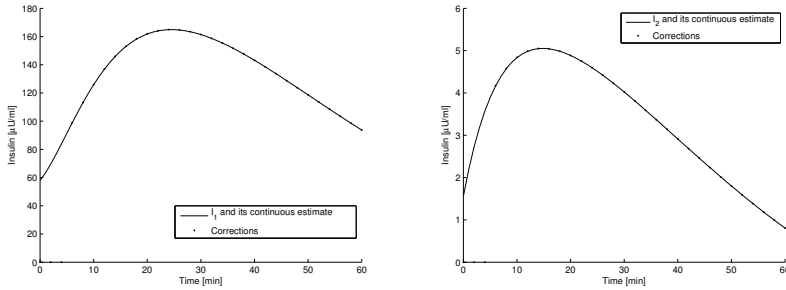


Fig. 3. Insulin and auxiliary variable I_2 evolution and estimates

Applications (2012)

2. Baranowski, J., Tutaj, A.: Comparison of continuous state estimation algorithms in a water tank system. In: Malinowski, K., Rutkowski, L. (eds.) Recent advances in control and automation, chap. 2, pp. 63–72. Akademicka Oficyna Wydawnicza EXIT (2008)
3. Bauer, W., Baranowski, J., Mitowski, W.: Advances in the Theory and Applications of Non-integer Order Systems: 5th Conference on Non-integer Order Calculus and Its Applications, Cracow, Poland, chap. Non-integer Order PI α D μ Control ICUMM, pp. 295–303. Springer International Publishing, Heidelberg (2013)
4. Bergman, R.N., Phillips, L.S., Cobelli, C.: Physiologic evaluation of factors controlling glucose tolerance in man: measurement of insulin sensitivity and beta-cell glucose sensitivity from the response to intravenous glucose. *J Clin Invest* 68(6), 1456–1467 (December 1981)
5. Biswas, P., Bhaumik, S., Patiyat, I.: Estimation of glucose and insulin concentration using nonlinear gaussian filters. In: 2016 IEEE First International Conference on Control, Measurement and Instrumentation (CMI). pp. 16–20 (Jan 2016)
6. Biyik, E., Arcaç, M.: Hybrid Newton Observer Design Using The Inexact Newton Method and GMRES. In: Proceedings of the 2006 American Control Conference. pp. 3334–3339. Minneapolis, Minnesota, USA (June 2006)
7. Biyik, E., Arcaç, M.: A hybrid redesign of Newton observers in the absence of an exact discrete-time model. *Systems & Control Letters* 55(6), 429–436 (June 2006)
8. Boltyanskii, V.: Mathematical methods of optimal control (1971)
9. Chee, F., Fernando, T.: Closed-Loop Control of Blood Glucose. *Lecture Notes in Control and Information Sciences*, Springer-Verlag, Berlin-Heidelberg (2007)
10. Clausen, W.H.O., De Gaetano, A., Vølund, A.: Within-patient variation of the pharmacokinetics of subcutaneously injected biphasic insulin aspart as assessed by compartmental modelling. *Diabetologia* 49(9), 2030–2038 (Sep 2006)
11. Dlugosz, M., Skrch, P.: The application of fractional-order models for thermal process modelling inside buildings. *Journal Of Building Physics* 39(5), 440–451 (MAR 2016)
12. Furler, S., Kraegen, E., Smallwood, R., Chisolm, D.: Blood glucose control by intermittent loop closure in the basal mode: Computer simulation studies with a diabetic model. *Diabetes Care* 8(6), 553–561 (1985)
13. Hairer, E., Nørsett, S., Wanner, G.: Solving Ordinary Differential Equations: I Nonstiff problems. Springer, 2 edn. (2000)

14. Haverbeke, N., Van Herpe, T., Diehl, M., Van den Berghe, G., De Moor, B.: Nonlinear model predictive control with moving horizon state and disturbance estimation - application to the normalization of blood glucose in the critically ill. In: Proceedings of the IFAC World Congress 2008 (IFAC08). pp. 9069–9074. Seoul, Korea (Jul 2008)
15. Hovorka, R., Chassin, L.J., Ellmerer, M., Plank, J., Wilinska, M.E.: A simulation model of glucose regulation in the critically ill. *Physiological Measurement* 29, 959–978 (2008)
16. Kovács, L.: Extension of the Bergman minimal model for the glucose-insulin interaction. *PERIODICA POLYTECHNICA SER. EL. ENG.* 50(1-2), 23–32 (2006)
17. Li, C., Hu, R.: Fuzzy-pid control for the regulation of blood glucose in diabetes. In: Proceedings of the 2009 WRI Global Congress on Intelligent Systems - Volume 02. pp. 170–174. GCIS '09, IEEE Computer Society, Washington, DC, USA (2009), <http://dx.doi.org/10.1109/GCIS.2009.280>
18. Li, J., Kuang, Y.: Systemically modeling the dynamics of plasma insulin in subcutaneous injection of insulin analogues for type 1 diabetes. *Mathematical Biosciences and Engineering* 6(1), 41–58 (January 2009)
19. Makroglou, A., Li, J., Kuang, Y.: Mathematical models and software tools for the glucose-insulin regulatory system and diabetes: an overview. *Applied Numerical Mathematics* 56, 559–573 (2006)
20. Mitkowski, W., Oprzedkiewicz, K.: Tuning of the Half-Order Robust PID Controller Dedicated to Oriented PV System. In: Latawiec, KJ and Lukaniszyn, M and Stanislawski, R (ed.) *Advances In Modelling And Control Of Non-Integer Order Systems. Lecture Notes in Electrical Engineering*, vol. 320, pp. 145–157. Opole Univ Technol, Dept Elect, Control & Comp Engrn (2015), 6th Conference on Non-Integer Order Calculus and its Applications (RRNR), Opole, POLAND, 2014
21. N'Doye, I., Voos, H., Darouach, M., Schneider, J., Knauf, N.: Static output feedback stabilization of nonlinear fractional-order glucose-insulin system. In: *Biomedical Engineering and Sciences (IECBES)*, 2012 IEEE EMBS Conference on. pp. 589–594 (Dec 2012)
22. Nocedal, J., Wright, S.: *Numerical Optimization*. Springer, 2 edn. (2006)
23. Oprzedkiewicz, K., Gawin, E.: A non integer order, state space model for one dimensional heat transfer process. *Archives Of Control Sciences* 26(2), 261–275 (JUN 2016)
24. Oprzedkiewicz, K., Mitkowski, W., Gawin, E.: An Estimation of Accuracy of Oustaloup Approximation. In: Szewczyk, R and Zielinski, C and Kaliczynska, M (ed.) *Challenges In Automation, Robotics And Measurement Techniques. Advances in Intelligent Systems and Computing*, vol. 440, pp. 299–307 (2016), International Conference Challenges in Automation, Robotics and Measurement Techniques (Automation), Warsaw, POLAND, MAR 02-04, 2016
25. Skruch, P., Dlugosz, M., Mitkowski, W.: Mathematical Methods For Verification Of Microprocessor-Based Pid Controllers For Improving Their Reliability. *Eksploatacja I Niezawodnosc-Maintenance And Reliability* 17(3), 327–333 (2015)
26. Van Herpe, T., Espinoza, M., Haverbeke, N., De Moor, B., Van den Berghe, G.: Glycemia prediction in critically ill patients using an adaptive modeling approach. *Journal of Diabetes Science and Technology* 1(3), 348–356 (May 2007)
27. Van Herpe, T., Haverbeke, N., Van den Berghe, G., De Moor, B.: Prediction performance comparison between three intensive care unit glucose models. In: Proceedings of the 7th IFAC Symposium on Modelling and Control in Biomedical Systems. Aalborg, Denmark (August 12 - 14 2009)

Intelligent system supporting diagnosis of malignant melanoma

Agnieszka Mikołajczyk*, Arkadiusz Kwasigroch*, Michał Grochowski*

*Gdańsk University of Technology

Abstract. Malignant melanomas are the most deadly type of skin cancers. Early diagnosis is a key for successful treatment and survival. The paper presents the system for supporting the process of diagnosis of skin lesions in order to detect a malignant melanoma. The paper describes the development process of an intelligent system purposed for the diagnosis of malignant melanoma. Presented system can be used as a decision support system for primary care physicians and as a system capable of self-examination of the skin with usage of dermatoscope. The system utilizes computational intelligence methods for proper classification of the dermoscopic features extracted from the medical images. The paper also proposes the extension of the well know ABCD method used for malignant melanoma diagnosis. The proposed system is tested on 126 and trained on 80 skin moles and the obtained results are very promising.

Keywords: diagnostics, decision support, image processing, artificial neural networks, melanoma malignant.

1 Introduction

Malignant melanomas are the most deadly type of skin cancers. Early detection is the key for further successful treatment. Regular skin examinations are very important, especially for people who are at higher risk of melanoma: Caucasian race, blue and green eyes, light hair, family melanoma history, advanced age, having atypical nevi [1],[2]. Unnoticed early enough malignant moles can be a cause of death. Although skin lesions are easily visible to the unaided eye, early-stage melanomas are often missed due to similarity in their appearance. On the other hand, many benign lesions are unnecessary examined by biopsy, however it is always worth examining just in case. A skin can be examined by a health care professional who can refer the patient to a dermatologist or by dermatologist himself. The most popular way of diagnosing skin cancers is a dermoscopy. The most popular methods of dermoscopy are the ABCD criterion, 7 point checklist, Menzies method [3],[4],[5]. In the last twenty years the interest of automatic computer-aided skin cancer detection and classification dynamically increased. For instance in [6] the authors tested automated method on 68 examples. The algorithm was based on ABCD method with 4 main features: Asymmetry (A), Border (B), Color (C) and Differential Structures (D). The system reached 73% of sensitivity and 70% specificity. Similarly, in [7] authors reports reaching a 73% accuracy (on 84

benign and 80 malignant lesions). In [8] a method based on texture analysis which reached high score of 92% accuracy on the test set consisting of 25 images is proposed. In [9] designed automatic system detecting blue-whitish veil regression structures. The method to detect irregular streaks which are important clues for Melanoma diagnosis using dermoscopy images was presented in [10]. The accuracy of the method reached 76.1% to 93.2% depending on data base. Although other approaches were developed in the past years, this problem is still open.

In the paper authors propose the new method for automatic malignant melanoma diagnostic based on the ABCD dermatoscopic method of Stoltz. Preliminary results were presented in [11], [12]. The main features of skin lesions are: the asymmetry, border and color. Based on extracted features, the neural network diagnosis system was learned to classify the skin lesions. The system was trained on 80 skin moles and tested on the 126 dermatoscopic and non-dermatoscopic images. It gets high final score with a sensitivity of 98% and specificity equal to 73%. Presented system can be used as a decision support system for primary care physicians and as a system capable of self-examination of the skin with dermatoscope.

2 Diagnosis of the malignant melanoma – fundamentals

The ABCD Stolz method is used for dermatoscopic differentiation between melanoma malignant lesions and benign moles. The rules are based on added scores from extracted features: Asymmetry (A), Borders (B), Color (C) and Differential structural components (D). In some works, the method is developed by another stage Evolution (E), in which the change in the mark during the last three months is analysed. The final diagnosis is generated based on the number of points calculated during the process of mole analysis. Authors of the paper propose the new set of features based on ABCD rule.

Asymmetry. In original ABCD method a symmetry is checked in 2 axes. It turns out that such an approach is often not sufficient. In proposed approach the shape of a lesion is compared to perfectly symmetric shape of a circle in order to avoid errors due to selecting wrong axes. The estimated radius of the circle and the center of the mass of the lesion are calculated based on detected binary image of a mole (see Fig. 1).

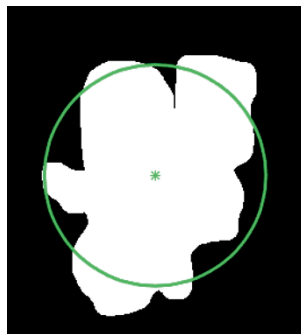


Fig. 1. An image with highlighted asymmetry feature

The symmetry feature is defined as follows:

$$A = \frac{\sum_{i=0}^k \sum_{j=0}^l -p_{i,j} + \sum_{i=0}^k \sum_{j=0}^l q_{i,j}}{\pi r^2} \quad (1)$$

where: A – symmetry feature, i, j – pixel coordinates, k, l – image size, $p_{i,j}$ – binary value of a pixel (i, j) inside a circle, $q_{i,j}$ – binary value of a pixel (i, j) outside a circle, r – radius [px].

Border. The benign moles have smooth, even edges without noticeable structures. According to the ABCD rule it is suggested to start the analysis of border from dividing it to 8 segments and then evaluate them separately. Such solution might lead to the various final scores accordingly to differently divided segments during the automatic analyses, hence it was changed.

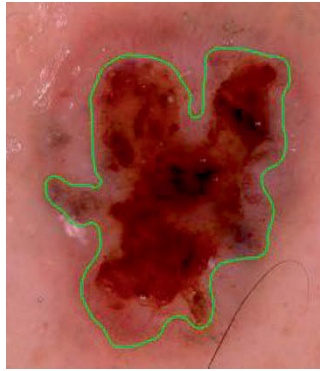


Fig. 2. An image with highlighted border feature

The Border feature is defined as follows

$$B = \frac{O * r}{2 * P} \quad (2)$$

where: B –border feature, O – perimeter of a circle [px], r – radius [px], P – area of a circle [px].

In case of perfectly smooth border the B feature would be equal to one. Every deformation of the contour will cause increase of this coefficient.

Color. The last of the proposed features is color. The presence of variety of colors (more than one or two) or uneven distribution of color can be a warning sign of melanoma. ABCD rule distinguishes 6 colors: light brown, dark brown, black, white, blue and red. in here presented approach Unfortunately in some cases described method can lead to overestimation of the result. Based on this remark authors proposed application of larger amount of features. The 4 additional color features are extracted:

- C_1 – the maximum change in the colors averaged on R, G and B channels,
- C_2 – the size of area with dominating blue color,
- C_3 – the size of area with dominating green color,
- C_4 – the maximum change of the brightness in the image.

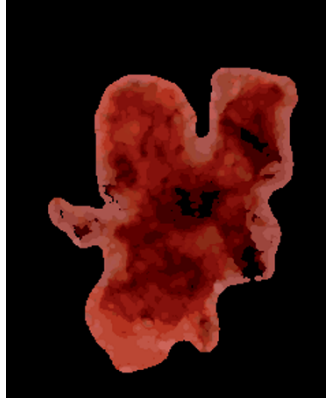


Fig. 3. An image with highlighted color feature

The first color feature is defined as follows

$$C_1 = \frac{R_{max}-R_{min} + G_{max}-G_{min} + B_{max}-B_{min}}{3} \quad (3)$$

where: C_1 – color feature, R_{max} , G_{max} , B_{max} , R_{min} , G_{min} , B_{min} are maximum and minimum R, G, B values.

Next two color features determine the size of area with dominating blue (C_2) and green (C_3) colors. The value of each pixel is checked in blue and green layer of an RGB image. If a value of a pixel is higher or equal than in other layers the value is added. Then, all summed up pixels are divided by the total number of pixels in the image. The process of calculating feature C_4 is conducted by the same way. The red component is omitted because for all kinds of moles it reaches similar values.

The last feature determines the maximum change of the brightness in the image:

$$C_4 = \frac{C_{max}-C_{min}}{2} \quad (4)$$

where: C_4 – color feature, C_{max} , C_{min} are the values of brightest and darkest pixels.

The last step in ABCD method, is analysis of differential structures of skin lesion. Due to lack of high resolution dermoscopic images, which would be sufficient for such analysis, this step was knowingly neglected. The information about malignant character

of lesion, is available as result of lesion asymmetry, border and color analysis. In described system, this interpretation is carried out by trained Artificial Neural Network. The successive steps of the algorithm are presented in a form of diagram in Fig. 4.

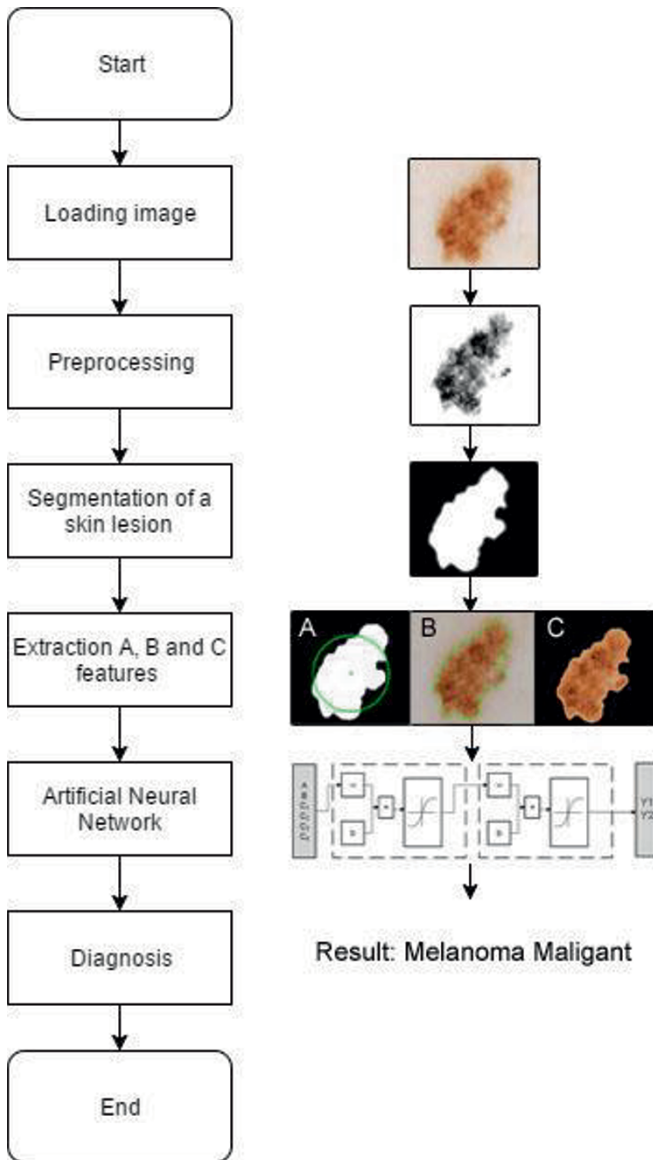


Fig. 4. Simplified block diagram of a system – all main steps of an algorithm with accompanying results of image processing

Presented system was implemented in MATLAB environment. The system is able to detect and recognize melanoma malignant on dermoscopic photography of a skin lesion. In the first steps, algorithm is preprocessing the images. Proper preprocessing can easily increase the quality of an image by reducing noise, deleting black frames around the picture and other unwanted elements such as hairs, light reflections, gel bubbles. The presence of such elements make the further steps of automatic diagnosis impossible. An example of the images hard to preprocess are presented in Fig. 5.



Fig. 5. An example of the images hard to preprocess

In proposed system firstly a contrast of the image is enhanced, then the numbers of colors in the picture are reduced and the photography is symmetrically blurred. After these operations the image is morphologically opened in order to delete the noise. Finally, prepared picture is converted to black and white format due to the next procedure: segmentation of a lesion. All described steps starting from the original image and resulting in extracted lesion without any components disturbing feature extraction process, are presented in Fig. 6.

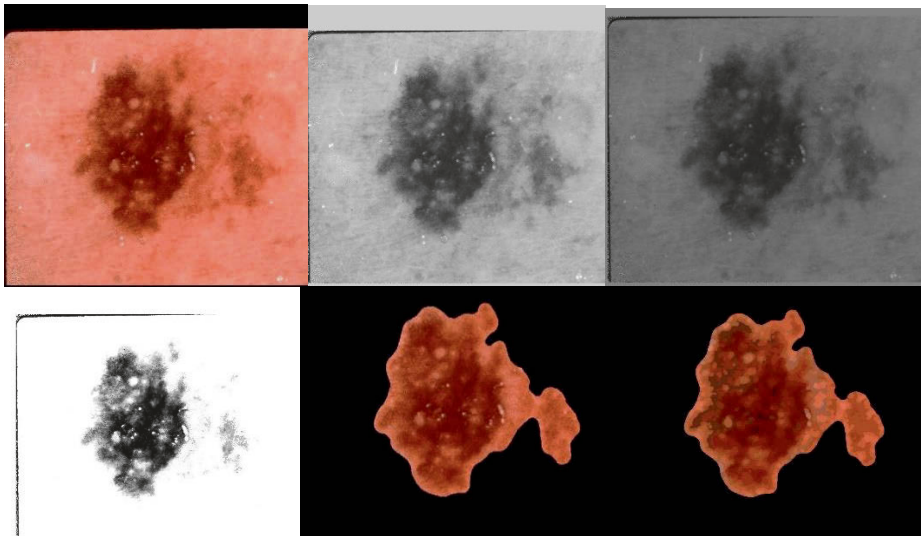


Fig. 6. Preprocessing – steps of an algorithm

2.1 Segmentation

Another important step in computer-aided diagnosis of melanoma is automated segmentation of dermoscopy images. The quality of a binary mask influences all the next steps and the final decision. The process of segmentation of the image is divided into few steps. Firstly, the photography is eroded and then thresholded with fixed parameter. All small objects within the image are deleted. Then the binary result of the previous operations are highly blurred. All existing holes in a lesion are filled in. Described steps are illustrated in the Fig. 7.

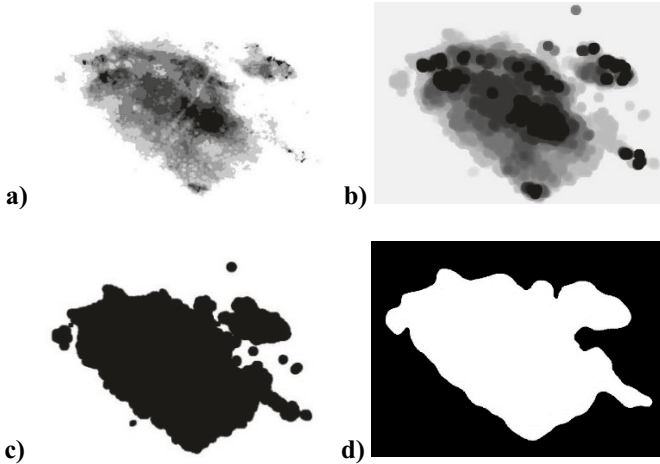


Fig. 7. Image segmentation – an example – a) an input image, b) after erosion, c) after threshold, c) segmented lesion

2.2 Feature extraction

All previous steps (preprocessing and segmentation) are carried out in order to extract the most important features of melanoma malignant skin lesions: symmetry, border and its color. Border of a skin lesion is determined by a contour of a proper segmented mole. The example contours are presented in Fig. 8.



Fig. 8. Example contours

Due to the described method of calculating the level of symmetry of the lesion, there is a need to find a center of a mass of binary image of the lesion. Then, system calculates the area of a lesion and estimate a radius of the ring from the equation for the area enclosed by the circle.

The values of all pixels are measured to determine the colors of the mole – percentage of blue and green tone pixels, the differences of the brightness and the variety of the colors. Before extracting the color features, all unwanted elements that disturb the diagnosis process like a skin and hairs are removed by adding a mask to the picture.

The color is very important feature of a malignant skin lesion. Benign lesions are often solid while malignant are mostly consisted of many different colors. The majority of benign moles are brown and dark brown, and do not have blue, white or black colors. Authors propose the k-means clustering method for unsupervised dividing colors into groups. The more each group is different from the others, the more attention the lesion should get. An image, divided into five clusters is presented in Fig. 9.

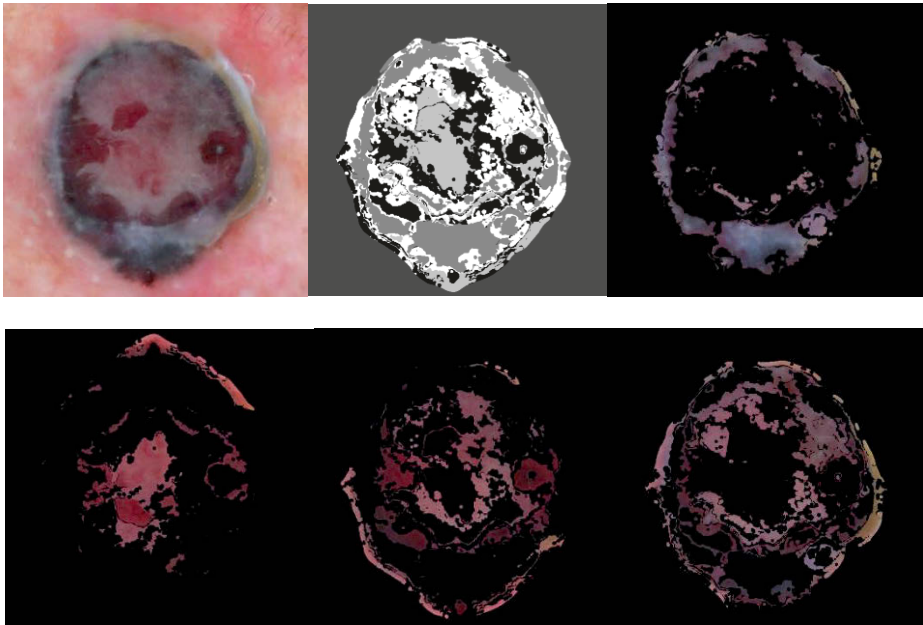


Fig. 9. Original image (upper left) and 5 related clusters

3 Decision support system

The features of symmetry, border and color are the base to draw the final decision. Decision support system is based on the feedforward neural network (NN). The neural network consists of 3 layers of neurons with sigmoidal activation functions. Determined at the pre-processing phase coefficients A , B , C_1 , C_2 , C_3 and C_4 are the NN inputs, while

the output is a vector (Y_1, Y_2) . During the training it was assumed that $Y_1=0$ and $Y_2=1$ means detection of malignant while $Y_1=1$ and $Y_2=0$ means benign.

Finally, it was assumed that detection of malignancy is when $Y_1 \in (0, 0.7)$ and $Y_2 \in (0.6, 1)$ and that the mole is benign when $Y_1 \in (0.3, 1)$ and $Y_2 \in (0, 0.4)$. If the returned value does not belong to the above ranges then the lesion is considered as suspicious.

The system was trained on 80 cases: 16 benign and 64 malignant and then tested on 126 images (100 malignant, 26 benign). Experiment resulted in 98 correctly classified melanomas and 19 benign moles. Sensitivity reached 98%, specificity 73.07%. In case of accepting occurrence of suspicious category as a correct result, the system reaches 99% of sensitivity and 73.07% of specificity.

4 Summary and concluding remarks

Authors presented intelligent system supporting the diagnosis of malignant melanoma along with a method based on ABCD rule. System reached high results with an accuracy of 92.3%. The system was trained on 80 skin moles and tested on the 126. The main features of skin lesions are: the asymmetry, border and color. The system is hidden behind the user-friendly and very intuitive Graphical User Interface, and might be used for self-examination of skin, as well as a decision support diagnostic tool.

Method is very accurate, even though it does not use information about mole pattern and structures. If said features were to be implemented, the application would surely give even more precise results, particularly for high resolution dermatoscopic images. In addition - increasing the number of training and testing images would benefit to system sensitivity.

Another way of improving system accuracy is taking into account information such as: age, weight, sex, type of skin and examined body part.

References

1. Greene, Mark H., et al. High risk of malignant melanoma in melanoma-prone families with dysplastic nevi. *Annals of internal medicine*, 1985, 102.4: 458-465.
2. Gellin GA, Kopf AW, Garfinkel L. Malignant Melanoma, A Controlled Study of Possibly Associated Factors. *Arch Dermatol*. 1969;99(1):43-48.
3. Stolz, W; Riemann, A; Cagnetta, A.B. ABCD rule of dermatoscopy-a new practical method for early recognition of malignant-melanoma. *European Journal of Dermatology*, 1994, 4.7: 521-527.
4. Johr, R.H. Dermoscopy: alternative melanocytic algorithms— the ABCD rule of dermatoscopy, menzies scoring method, and 7-point checklist. *Clinics in dermatology*, 2002, 20.3: 240-247.
5. Scott, H.J. The CASH (color, architecture, symmetry, and homogeneity) algorithm for dermoscopy. *Journal of the American Academy of Dermatology*, 2007, 56.1: 45-52.

6. Luís F., Caeiro Margalho, Guerra Rosad, Automatic System for Diagnosis of Skin Lesions Based on Dermoscopic Images.
7. Zortea, Maciel; Skrøvseth, Stein Olav; Godtliebsen, Fred. Automatic learning of spatial patterns for diagnosis of skin lesions. In: 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology. IEEE, 2010. p. 5601-5604.
8. Sheha, Mariam A.; Mabrouk, Mai S.; Sharawy, Amr. Automatic detection of melanoma skin cancer using texture analysis. *International Journal of Computer Applications*, 2012, 42.20: 22-26.
9. G. Di Leo, G. Fabbrocini, A. Paolillo, O. Rescigno and P. Sommella, "Towards an automatic diagnosis system for skin lesions: Estimation of blue-whitish veil and regression structures," 2009 6th International Multi-Conference on Systems, Signals and Devices, Djerba, 2009, pp. 1-6.
10. M. Sadeghi, T. K. Lee, D. McLean, H. Lui and M. S. Atkins, "Detection and Analysis of Irregular Streaks in Dermoscopic Images of Skin Lesions," in *IEEE Transactions on Medical Imaging*, vol. 32, no. 5, pp. 849-861, May 2013.
11. Mikołajczyk, A., Analiza znamion skórnych za pomocą metod przetwarzania obrazu i algorytmów inteligencji obliczeniowej, *Innowacyjne rozwiązania w obszarze automatyki, robotyki i pomiarów. Konkurs Młodzi Innowacyjni*, 2016, 87-104.
12. Mikołajczyk, A., Kwasigroch, A., Grochowski, M., System wspomagający diagnostykę czerniaka złośliwego przy pomocy metod przetwarzania obrazu i algorytmów inteligencji obliczeniowej. *Zeszyty Naukowe Politechniki Gdańskiej nr 51*, 2016, str. 119-122.

Applications of decision support systems in functional neurosurgery

Konrad A. Ciecierski PhD(✉)^{1,3} and Tomasz Mandat MD, PhD^{2,3}

¹ Research and Academic Computer Network, Warsaw

² Department of Neurosurgery, M. Skłodowska-Curie
Memorial Oncology Center, Warsaw

³ Department of Neurosurgery, Institute of Psychiatry and Neurology in Warsaw
konrad.ciecierski@gmail.com, tomaszmandat@yahoo.com

Abstract. Functional neurosurgery is used for treatment of conditions in central nervous system that arise from its improper physiology. One of the possible approaches is Deep Brain Stimulation (DBS). In this procedure a stimulating electrode is placed in desired brain's area to locally affect its activity. Among others, DBS can be used as a treatment for dystonia, depression, obsessive-compulsive disorder (*OCD*) and Parkinson's Disease (*PD*). In this paper authors focus on application of classifiers in Deep Brain Stimulation (*DBS*) for Parkinson's Disease (*PD*). In neurosurgical treatment of the Parkinson's Disease the target is a small (9 x 7 x 4 mm) deeply in brain situated structure called *Subthalamic Nucleus (STN)*. The goal of the Deep Brain Stimulation is the precise permanent placement of the stimulating electrode within target nucleus. As this structure poorly visible in CT⁴ or MRI⁵ it is usually stereotactically located using microelectrode recording. Several microelectrodes are parallelly inserted into the brain and then in measured steps advanced towards expected location of the nucleus. At each step, usually from 10 mm above expected center of the *STN*, the neuronal activity is recorded. Because *STN* has a distinct physiology, the signals recorded within it also present specific features. By extraction certain attributes from recordings provided by the microelectrodes, it is possible to construct a binary classifier that provides useful discrimination. This discrimination divides the recordings into two classes, i.e. those registered within the *STN* and those registered outside of it. From this it is known which microelectrodes and at which depths have passed through the *STN* and thus a physiological map of its surrounding is made.

Keywords: Decision support system, DBS⁶, STN⁷, DWT⁸ decomposition, Signal power, Classification, Random Forest

⁴ Computer Tomography

⁵ Magnetic Resonance Imaging

⁶ Deep Brain Stimulation

⁷ Subthalamic Nucleus

⁸ Discrete Wavelet Transform

Introduction

Parkinson Disease (PD) is chronic and advancing movement disorder. The risk factor of the disease increases with the age. As the average human life span elongates also the number of people affected with PD steadily increases. PD is primarily related to lack of the dopamine that is caused by death of specialized dopamine-producing cells within the brain. People as early as in their 40s, otherwise fully functional are seriously disabled and require continuous additional external support. The main treatment for the disease is pharmacological one. Unfortunately, in many cases the effectiveness of the treatment decreases with time while some other patients do not tolerate anti PD drugs well. In such cases, patients can be qualified for the surgical treatment of the PD disease. This kind of surgery is called DBS⁹. Goal of the surgery is the placement of the permanent stimulating electrode into the *STN*. This nucleus is a small – deep in brain placed – structure (Fig. 1) that does not show well in CT or MRI scans.

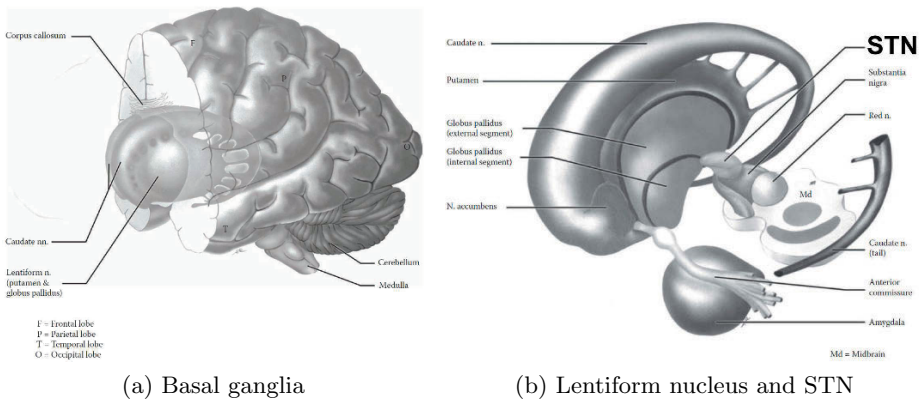


Fig. 1: Location of basal ganglia and STN

Atlas of Functional Neuroanatomy. ©2006 by Taylor & Francis Group, LLC

Having only an approximate location of the *STN*, during DBS surgery a set of parallel micro electrodes is inserted into patient's brain.

As they advance, the activity of surrounding neural tissue is recorded. Typically the recording starts 10 mm above the expected location of the *STN* and continues with 1 mm steps for following 15 mm. Localization of the *STN* is possible as it has distinct physiology and yields specific micro electrode recordings [1]. It still however requires an experienced neurologist / neurosurgeon to tell whether recorded signal comes from the *STN* or not [2].

That is why it is so important to provide some objective and human independent way to group and classify recorded signals. Analytic methods, shown in this

⁹ Deep Brain Stimulation

paper have been devised for that purpose. Taking as input recordings made by set of electrodes at subsequent depths they have proven the ability to provide correct information as to which of the recordings have been made inside the *STN*.

This is done by means of trained Random Forest classifier which taking as input 40 described in this paper attributes calculated for a given recording gives binary classification stating if it belongs to the *STN* or *MISS* class. Recordings qualified to the *STN* class are deemed to have been recorded within the *STN* area.

This information provides neurosurgeon with additional supporting knowledge about extent of the *STN* on the tracks of the micro electrodes and in such way helps to pinpoint the target of the DBS surgery.

1 Characteristics of the recorded signals

Recording electrode can detect action potentials occurring in cells within radius of $50 \mu\text{m}$ from the electrode's recording tip [3]. Neurons that are farther away contribute to the background noise present in the recording signal. Even in the close vicinity of the recording tip, within radius of $50 \mu\text{m}$ there can be over 100 neurons. All neighboring neurons summarily produce the electrical activity received by the electrode. From this, the recorded signal can roughly be divided into its background noise and high amplitude spiking activity. This high amplitude of spikes is of course relative to the level of the background noise. Generally spikes are in range below $500 \mu\text{v}$. This in turn means that during recording this signal has to be greatly amplified and is also very sensitive to external contamination. Such minuscule intrusions as touching the stereotactic frame, patient moving, speaking during the recording process or even his heartbeat cause presence of high amplitude artifacts in the recordings[2].

2 Recording's attributes

From the micro electrode recorded signal two kinds of information can be obtained. First one comes from spikes – neuronal action potentials – i.e. electrical activity of neurons being near the electrode recording point. Second information is derived from the background noise present in the signal. This background noise is a cumulative electrical activity coming from the more distant neurons. Both approaches require prior detection of the spikes [2][4]. First one uses information about spikes in a direct way. The background noise analysis requires prior spike removal. Removal of course implies prior spike detection.

For presented analysis a set of 30 characteristic attributes is defined for each recording:

- five attributes derived directly from the spike occurrence
- five attributes derived from the signal background
- four meta attributes derived indirectly from the signal background
- sixteen temporal features derived from the signal background

2.1 Spike detection and sorting

It is possible to detect spikes occurring in cells being within radius of around $50 \mu\text{m}$ from the electrode's recording tip [3]. Spikes from far neurons may have lower amplitude and be wider. Change of width is caused by the fact that with increasing distance more and more of the spike's high frequency components are lost [3]. Distant neurons contribute to the noise present in the recorded signals. Because of that, spikes coming from two neurons of the same type, with different distance from the electrode can be recorded with both different amplitude and width. Depending on the brain region, within radius of $50 \mu\text{m}$ one might find well over 100 neurons. This neurons can produce spikes in different patterns and frequencies. It is also not uncommon that couple of neurons produce spikes within few milliseconds of each other and that the recorded spikes would overlap. Because of the above, it is particularly difficult to detect and sort the spikes. In this paper described is method of spike detection based on their amplitude and general shape. Amplitude approach requires however that the signal has to be firstly high pass filtered. This procedure ensures that all spikes have similar absolute amplitude. The high pass filtering is DWT based [5] and removes all frequencies below 187.5 Hz . Looking at Fig. 2a and 2b it is clearly apparent that *DWT* filtering removed low frequency oscillations.

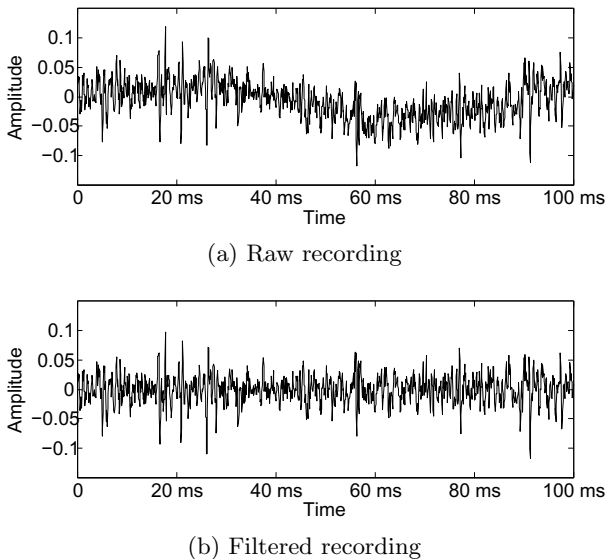


Fig. 2: *DWT* high pass filtering

Additionally some high frequency components of the signal [6] are also removed

during fine tuned multi-band DWT filtering to cancel noise that can disturb shapes of the individual spikes.

Assuming that unwanted frequency components have already been filtered out from the signal, one may attempt to detect spikes using amplitude analysis. Assuming that X denotes input signal, the spike detection threshold is defined as $V_{thr} = 4 \sigma(X)$ where $\sigma(X)$ is given by equation 1.

$$\sigma(X) = \frac{\text{median}(|x_1|, \dots, |x_n|)}{0.6745} \quad \sigma_n X = x_1, \dots, x_n \quad (1)$$

This estimation, given here by equation 1 has been introduced in [7] and as explained in [8] greatly diminishes the influence of the spikes on the calculated standard deviation value.

Spike sorting has been described in detail in [4][6][8], its goal is to group found spikes according to their shape. Shape of the spike produced by a neuron derives from its morphology and in natural conditions does not change [9]. Knowing that, one can assume that if in a given recording there are observed several different spike shape classes, then they must have been generated by a few different neuron cells in the vicinity of the electrode. This helps to discriminate situations where near the electrode are several moderately active cells from a case where in the neighborhood there is one but highly active cell (such cells are expected to be found within the STN [9][10][11]).

For spike sorting in this work the wavelet approach based on [8] is used. Wavelet approach has been found to provide better results than the *PCA* one [12]. Signals recorded by micro electrodes are sampled with $24KHz$ which means that there are 24 samples for each millisecond. Assuming that spike lasts for $0.5 ms$ before reaching its maximal absolute amplitude and than for $1.1 ms$ after it, then its amplitude vector contains $[0.5 * 24] + 1 + [1.1 * 24] = 39$ samples. 12 samples for pre-maximal part, one for maximum and 26 for post-maximal part of the spike.

Process of spike sorting consists of several stages. First all detected spikes are held as a set of 39 element vectors. Then each vector is right zero padded to the length of 64 to satisfy the needs of wavelet transformation. Wavelet transformation is performed fully i.e. six times for each vector to obtain its transformation. Having thus obtained set of wavelet vectors the 10 wavelet coefficients best suited for clustering are chosen using the Kolomogorov–Smirnov test for normality [8].

Clustering is then made five times with target cluster number in $1 \dots 5$. Finally, clustering with greatest mean of silhouette value is chosen and spike classes obtained. Clusters containing less than 15 spikes are discarded. In the last step similar clusters are merged together [6].

2.2 Spike derived attributes

Characteristics derived from the spike occurrences focus on the observed spiking frequency. Assuming that for recording rec , the total number of spikes detected in it is denoted as $SpkCnt(rec)$ and its length in seconds is $RecLen(rec)$ then the first attribute is given by equation 2.

$$AvgSpkRate(rec) = \frac{SpkCnt(rec)}{RecLen(rec)} \quad (2)$$

Assuming that for recording rec spikes occurred chronologically at times t_0, \dots, t_{n-1} and that $s_j = t_{j+1} - t_j$ then a set of intra-spike intervals S is defined by equation 3.

$$S(rec) = \{s_0, \dots, s_{n-2}\} \quad (3)$$

Let us define now the sets of *bursting* intra-spike intervals as follows:

$$S_{brst}(rec) = \{s_j \in S(rec) : s_j \leq 33ms\} \quad (4)$$

Having the above defined, the final two spike characteristic attributes can be given:

$$BurstRatio(rec) = \frac{\overline{\overline{S_{brst}(rec)}}}{\overline{\overline{S(rec)}}} \quad (5)$$

For recordings having no spikes, *BurstRatio* is defined with zero value. Besides those two spike based attributes are two named *AvgSpkRateScMax* and *BurstRatioScMax* which are biggest values of *AvgSpkRate* and *BurstRatio* calculated for single cell. There is also *MPWR* attribute – power of the derived meta signal, it has been explained in detail in [13]. As it has been shown in [14] is also possible to construct classifier that basing mainly on spiking activity detects recordings from brain region situated below the STN. This information is important as it informs neurosurgeon that electrodes need not to be advanced further [2][9].

2.3 Background based attributes

Background based attributes are calculated basing upon the signal's background noise. Cells that are farther than $50 \mu m$ away from electrode's lead are too far away for their spikes to be clearly registered. Their summary electrical activity creates the background noise ever present in signals obtained from microrecording. This background activity is a rough measure of amount and activity of neuron cells in the broader vicinity of the electrode. Increased values of background attributes are hallmark of the *STN* as it contains many densely packed and highly active neurons [2][9][11].

Because contaminating artifacts causes increase in both power and amplitude of the signal, its removal is critically important to avoid occurrence of falsely high values of background based attributes. Without removal of the artifacts, their presence can increase value of any of the background attributes by the order of the magnitude.

Artifacts reside mainly in two frequency bands $(0, 187\frac{1}{2}) Hz$ and $(187\frac{1}{2}, 375) Hz$. Normally proper for each frequency band *DWT* coefficients have uniform amplitude that fits into $[-\frac{3}{2}\sigma, \frac{3}{2}\sigma]$. Looking for coefficients with amplitudes exceeding

that threshold should therefore provide information about localization of artifacts in a given recording. Time bound relation between *DWT* coefficients and signal samples allows for exact artifacts removal [6]15].

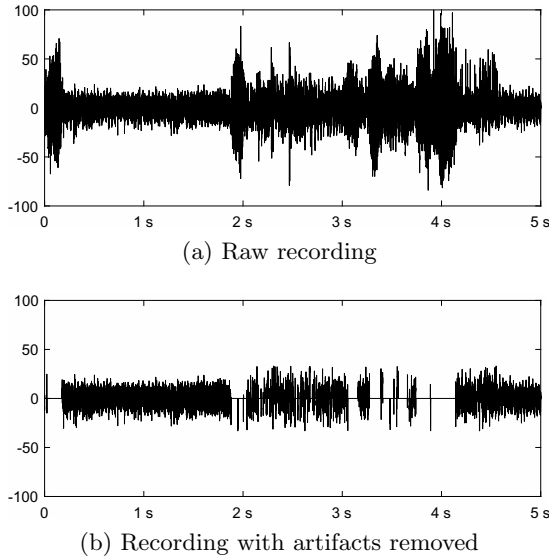


Fig. 3: *DWT* high pass filtering

After the procedure of artifact removal has been specified, it is safe to define five background based attributes:

PRC80 80th percentile of amplitude's module.

PRCDLT difference between 95th and 80th percentile of amplitude's module in proportion to 95th percentile of amplitude's module.

RMS Root Mean Square of the signal.

LFB power of the signal's background in range 0 – 500 Hz.

HFB power of the signal's background in range 500 – 3000 Hz.

The power of the signal in frequency bands is calculated basing upon values of corresponding wavelet components in signal's decomposition [6].

All of background based attributes are normalized on two levels, firstly they are normalized according to the length of the recording. In the second step the normalization is done on the electrode level. As it was described in the introduction, the recording starts at depths that is about 10 mm above the STN. In this way, first few millimeters of the tract of the electrode goes through the brain area with generally uniformly low activity. For each background attribute its value is normalized according to its average value in the first five depths of the

track of the electrode [6][16].

Below on the Fig. 4 are the values of the *LFB* attribute calculated for single electrode pass. The STN has been traversed by the electrode at depths ranging from $-2000 \mu m$ to $+2000 \mu m$.

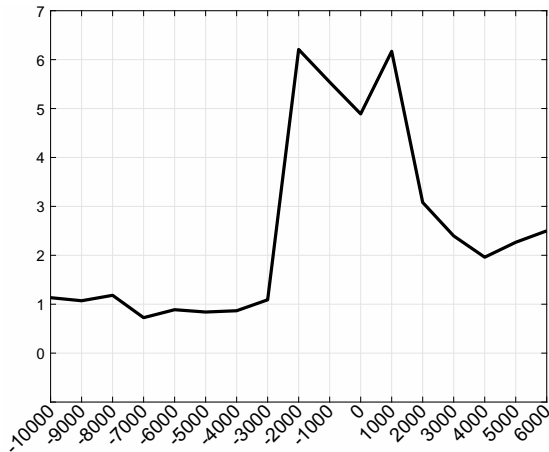


Fig. 4: LFP attribute at consecutive depths

2.4 Meta attributes

To obtain attributes that are less affected by local variations at certain depths the concept of derived meta-attribute is introduced. Meta attribute is calculated as 1-padded, five element wide moving average of the proper direct attribute.

Assume that recordings from given electrode have been made at consecutive depths $d_0 \dots d_{n-1}$ with constant step of $1000 \mu m$ between adjacent ones. Recording from those depths are then given as a list $(rec_{d_0}, \dots, rec_{d_{n-1}})$. Padding is defined in such way that $\forall_{j < d_0} Attr(rec_j) = 1$ and $\forall_{j > d_{n-1}} Attr(rec_j) = 1$. Then, finally, the meta attribute $MAttr(rec_j)$ for the attribute $Attr(rec_j)$ is given by equation 6.

$$MAttr(rec_j) = \frac{\sum_{k=-2}^2 Attr(rec_{j+1000k})}{5} \quad (6)$$

Meta attributes value for some of attributes defined in section 2.3, namely PRC_{80} , RMS , LFB and HFB are thus named $MPRC_{80}$, $MRMS$, $MLFB$ and $MHFB$.

2.5 Temporal attributes

As it can be observed on Fig. 4 at the dorsal/ventral borders of the STN the value of the attributes (especially those background based) might show significant increase/decrease. Temporal attributes were designed especially for detection of edges of the STN.

For electrode E at depth D and attribute X , the temporal attribute

X_{1U} holds maximal increase of attribute X over distance of 1 mm detected by an electrode E at depths dorsal to D .

X_{2U} holds maximal increase of attribute X over distance of 2 mm detected by an electrode E at depths dorsal to D .

X_{1D} holds maximal decrease of attribute X over distance of 1 mm detected by an electrode E at depths dorsal to D .

X_{2D} holds maximal decrease of attribute X over distance of 2 mm detected by an electrode E at depths dorsal to D .

To minimize influence of random fluctuations the change affecting any above temporal attribute must be above certain threshold. In this way those attribute in scope of all recordings provided from single electrode change at the point of its entry and leave of the STN.

Temporal attributes were defined for PRC_{80} , RMS , LFB and HFB attributes from section 2.3.

3 Classification results and evaluation

Using described attributes the Random Forest classifier has been constructed. The classifier has been trained using 12494 recordings made during 115 surgeries. The evaluation has been done on 5614 recordings made during another 57 surgeries. The achieved sensitivity was 88 % and specificity 96 %.

With temporal attributes excluded from consideration the data also clusters well according to the STN label. The sensitivity provided by K-Means clustering in such case is 0.965, still clustering is not specific, the specificity at level 0.52 is definitively unacceptable.

4 Summary and conclusions

Obtained results are not based of cross reference of a single data set. After the training phase, the solution has been used and validated during over 50 surgical procedures. The achieved specificity and sensitivity fulfill the strict demands imposed by the medical procedures. Software basing upon described work is currently being used during surgeries in Institute of Psychiatry and Neurology in Warsaw.

References

1. T. Mandat, T. Tykocki, H. Koziara et. al. Subthalamic deep brain stimulation for the treatment of Parkinson disease, *Neurologia i neurochirurgia polska* 45.1 (2011): 32-36
2. Z. Israel, K. J. Burchiel. *Microelectrode Recording in Movement Disorder Surgery*, Thieme Medical Publishers, 2004
3. Klas H. Pettersen, Gaute T. Einevoll. Amplitude Variability and Extracellular Low-Pass Filtering of Neuronal Spikes *Biophysical Journal*, 2008 94: 784-802
4. Alexander B. Wiltschko, Gregory J. Gage, and Joshua D. Berke. Wavelet Filtering before Spike Detection Preserves Waveform Shape and Enhances Single-Unit Discrimination *J Neurosci Methods*. 2008, 173: 34-40
5. A. Jensen, A. Ia Cour-Harbo. *Ripples in Mathematics*, Springer-Verlag, 2001
6. K. Ciecierski, *Decision Support System for surgical treatment of Parkinsons disease*, PhD thesis, Warsaw University of technology Press, 2013
7. D. L. Donoho, De-Noising by Soft-Thresholding, *IEEE Transactions On Information Theory*, May 1995, 41(3):613-627
8. R. Quian quiroga, Z. Nadasdy, Y. Ben-Shaul, *Unsupervised Spike Detection and Sorting with Wavelets and Superparamagnetic Clustering*, MIT Press, 2004
9. R. Nieuwenhuys, C. Huijzen, and J. Voogd. *The Human Central Nervous System*, Springer, 2008
10. P. Novak, A. W. Przybyszewski, A. Barborica, P. Ravin, L. Margolin, J. G. Pilitsis Localization of the subthalamic nucleus in Parkinson disease using multiunit activity, *Journal of the neurological sciences* 310.1 (2011): 44-49.
11. A. Zaidel, A. Spivak, L. Shpigelman, H. Bergman, and Z. Israel Delimiting sub-territories of the human subthalamic nucleus by means of microelectrode recordings and a Hidden Markov Model, *Movement disorders* 24, no. 12 (2009): 1785-1793.
12. A. Pavlov, V.A. Makarov, I. Makarova, and F. Panetsos, Sorting of neural spikes: When wavelet based methods outperform principal component analysis. *Natural Computing*, 6(3):269-281, 2007.
13. K. Ciecierski, Z. W. Raś, A. W. Przybyszewski, Foundations of recommender system for STN localization during DBS surgery in Parkinson's patients, *Foundations of Intelligent Systems, ISMIS 2012 Symposium, LNAI, Vol. 7661*, Springer, 2012, 234-243
14. K. Ciecierski, T. Mandat. Detection of SNr Recordings Basing upon Spike Shape Classes and Signals Background. *International Conference on Brain and Health Informatics*. Springer International Publishing, 2016, 336-345
15. K. Ciecierski, T. Mandat, R. Rola, Z. W. Raś, A. W. Przybyszewski Computer aided subthalamic nucleus (STN) localization during deep brain stimulation (DBS) surgery in Parkinson's patients, *Annales Academiae Medicae Silesiensis*, 2014, 68, 5: 275-283
16. K. Ciecierski, Z. W. Raś, A. W. Przybyszewski, Foundations of automatic system for intrasurgical localization of subthalamic nucleus in Parkinson patients, *Web Intelligence and Agent Systems*, 2014/1, IOS Press, 2014, 63-82
17. Klas H. Pettersen, Gaute T. Einevoll. Amplitude Variability and Extracellular Low-Pass Filtering of Neuronal Spikes *Biophysical Journal*, 2008 94: 784-802
18. Claude Bdard, Helmut Krger and Alain Destexhe. Modeling Extracellular Field Potentials and the Frequency-Filtering Properties of Extracellular Space *Biophysical Journal*, 2004 86: 1829-1842

Deep convolutional neural networks as a decision support tool in medical problems – malignant melanoma case study

Arkadiusz Kwasigroch*, Agnieszka Mikołajczyk*, Michał Grochowski*

*Gdańsk University of Technology

Abstract. The paper presents utilization of one of the latest tool from the group of Machine learning techniques, namely Deep Convolutional Neural Networks (CNN), in process of decision making in selected medical problems. After the survey of the most successful applications of CNN in solving medical problems, the paper focuses on the very difficult problem of automatic analyses of the skin lesions. The authors propose the CNN structure and the way to cope with the insufficient number of learning data. The research was carried out and validated on the data base of over 10000 images. The efficiency of the proposed approach reaches 84%.

Keywords: deep learning, deep convolutional neural network, image analysis, machine learning, malignant melanoma, transfer learning

1 Introduction

Deep neural networks (DNN) are currently the main tool for image analysis. The most popular deep learning architecture is convolutional neural network (CNN) [1]. The one of the first application of convolutional network was bank the check reading system, created in 1993. By the late 90's system was reading over 10% of all checks in US [2]. The number of applications increased over time. Google used deep learning algorithms for face and license plates recognition in StreetView images. Detected faces and plates were then blurred for privacy protection [3].

Current huge interest in deep learning began in 2012, when convolutional neural network won famous ImageNet Large-Scale Visual Recognition Challenge. Deployment of deep learning algorithms enabled for decreasing of classification error from 26.1% to 15.3%, comparing with the conventional methods [4]. Since that time, deep learning (DL) methods develop rapidly. Research on deep learning is supported by academic and commercial environment. Convolutional neural networks are currently the state of the art tool for image recognition.

Deep convolutional neural networks are used not only in computer vision tasks. Deep architectures achieved satisfactory results in the fields such as: speech recognition [5], natural language processing [6], machine translation [7], reinforcement learning [8], speech synthesis [9]. Moreover, the system based on deep learning has de-

feated world champion in game of GO, that is considered much more complex than chess [10]. This event is considered as a milestone in artificial intelligence research.

Deep learning has produced promising results in different medical tasks. The algorithms have been adapted to medical image analysis such as MRI, dermatoscopic images, standard images. Deep neural networks were used for: mammography mass lesion classification [11], tissue classification [12], detection of diabetic retinopathy in retinal fundus photographs [13], Alzheimer disease classification [14].

In this paper we propose deep convolutional neural network for malignant melanoma classification. Skin cancer is the most common cancer in the world. Malignant melanomas represent from 5 to 8% of the skin cancer. Early diagnosis is a key to successful treatment and allows to improve survival rate. Too late or not diagnosed malignant melanoma can cause death.

The paper is organized as follows: in Sect. 2 we describe fundamentals of deep convolutional networks. The details of implemented system are described in Sect. 3. In Sect. 4 we present achieved results. We conclude the paper in Sect. 5.

2 Deep convolutional neural network – fundamentals

Deep neural networks are composed of multiple modules (Fig.1.), aligned one after other. Each module consists of following layers:

- convolutional layer,
- nonlinear layer,
- pooling (subsampling) layer

The sequence of these layers is repeated several times leading to multilayer structure with very large number of layers. The last part of DNN is conventional multilayer feedforward neural network with added dropout layer between its layers.

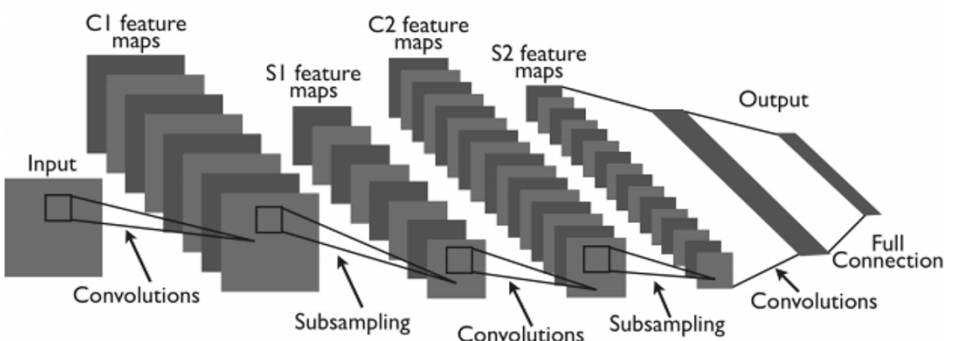


Fig. 1. Deep convolutional neural network architecture [2]

Convolutional layer.

The input and output of convolutional layer are sets of 2D matrices called feature maps. Convolutional layer consist of fixed number of filters, chosen before the training process. Each filter detects a particular feature and produces feature map on its output. The filter is described by the set of weights (kernel). The size of the kernel is chosen by user (typically 3x3) and also is dependent on dimension of the previous layer. For example: if chosen size is 3x3, the number of features maps in the previous layer is n than the kernel has a size of $3 \times 3 \times n$. The filter moves over the image and performs operation of discrete convolution in each position.

The main advantage of deep learning lies in avoiding the laborious manual pre-processing and extracting of the important features. Convolutional filters are trainable feature extractors, learned from the data. As mentioned, each filter detects particular features. For example, in the case of face detection, first convolutional layer detects low level features such as shapes, edges, colors, medium layers react to higher level features such as eyes, mouth, nose.

Nonlinear layer.

The nonlinear layer performs nonlinear transformation on outputs of convolutional layers in order to increase the approximation abilities of the network. The rectified linear unit (ReLU) is currently the most popular nonlinear function used in deep learning. ReLU function is described by equation $f(x)=\max(0,x)$. An important advantages of rectified linear unit are: fast computation compared to sigmoid and tanh functions, function is non-saturating, constant gradient that solve problem of vanishing gradient. Deep convolutional networks with ReLUs train several times faster than their equivalents with tanh units [4].

Pooling layer.

The pooling layer decreases the size of feature maps. The most popular pooling operation is MaxPooling, that partitions images into a sets of 2x2 non overlapping rectangles. In that case maximum value from every rectangle creates down-sampled image. Such pooling layer reduces the size of feature map by 4 times.

Dropout layer.

Dropout layers were employed to prevent overfitting. That technique relies on a mechanism, in which connections are randomly removed during the training. The number of dropped connection is controlled by parameter called dropout rate (in our application set to 0.5). The connections are dropped out only during training process. Dropout layers are removed after the training [15].

Fully connected layer.

The stack of convolutional modules is followed by the classification module. Typically classification module consists of 1-2 fully (conventionally) connected layers followed by final softmax layer (multiclass classification) or sigmoid neuron (binary classification).

3 CNN approach for automatic diagnosis of the malignant melanoma

3.1 Database and preprocessing

The dataset used in this paper during the research was created by the International Skin Imaging Collaboration [16]. The dataset consist of high-quality, dermatoscopic images, collected from clinics in Europe, Australia and United States, acquired from patients of various age and sex. The images are annotated by high-skilled experts.

The database was split into the training and test sets. The training set contains 9300 benign images and 670 malignant. The test set contains 100 images of each class. Examples of benign and malignant lesions are shown in Fig.2.

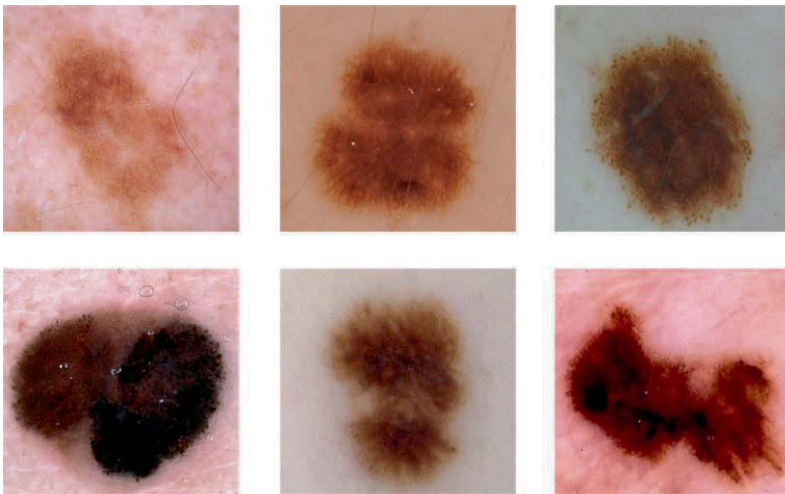


Fig. 2. Examples of benign (first row) and malignant (second row) lesions

Up-sampling. Unfortunately, the database is highly imbalanced what is disadvantageous for the learning process. In order to overcome this problem we have performed an up-sampling of under-represented class by adding copies of images to make equal quantity of photos in each class.

Scaling. Another problem with this database is various resolution of images in the dataset. Therefore, it was necessary to resample the images by fixing a constant resolution (224x224). Due to improving the learning process the values of the pixels (0-255) were rescaled to the range of 0–1. Moreover, afterwards they were scaled to have 0 mean and variance equal 1.

Data augmentation. In order to reduce the risk of overfitting we perform data augmentation process. Data were augmented by the number of random image transformations: rotations (-30–30 degrees), width and height shift (0.1 of the image size), zoom (90%–110%), horizontal and vertical flips.

3.2 Deep convolutional network architecture

During the research we took advantage of the DNN structure called VGG19 [17]. Then, we modify this structure by adding 2 convolutional layers and 2 fully connected layers and one sigmoid neuron at the network output.

The neural network takes RGB 224x224 image as an input and generate output that indicates diagnosis – benign or malignant lesion. The architecture of employed network is presented in table 1. The neural network contains about 241M trainable parameters.

Table 1. Neural network architecture for melanoma diagnosis.

Input (224x224x3)
64 conv
64 conv
maxpooling
128 conv
128 conv
maxpooling
256 conv
256 conv
256 conv
256 conv
maxpooling
512 conv
512 conv
512 conv
512 conv
maxpooling
512 conv
512 conv
512 conv
512 conv
1024 conv
1024 conv
4048 FC
dropout
4048 FC
dropout
Output (sigmoid neuron)

The number before layer name refers to the number of filters or the number of neurons. The kernel sizes of convolutional filters are set to $3 \times 3 \times n$, where n is the number of feature maps from the previous layer. We use ReLU activation function in fully-connected layers.

3.3 Learning algorithm

During the training the minibatch gradient descent algorithm with Nesterov momentum was used. The batch size was set to 8, momentum to 0.9. Training algorithm minimizes binary cross entropy function. Learning rate was initially set to 0.01 and was decreased 10 times when training loss stopped decrease.

3.4 Transfer learning influence on classification results.

It is difficult task to train deep neural network with randomly initialized weight, especially on relatively small dataset. To overcome this problem, we used a technique called transfer learning. This approach consists in using weights of the pre-trained on huge dataset neural network as the initial condition for the learning of the target network. It turns out that this approach greatly facilitates the proper learning neural network. It is interesting that deep network trained on classes such as car, ship, dog etc. perform well on e.g. medical dataset.

4 Results

4.1 Classification results

During the research we have designed two networks CNN1 and CNN2. Both networks have the same architecture – convolutional modules took from VGG19 modified by adding 2 convolutional layers and fully-connected module. They differ in weight initialization before the training process. In CNN1 network all weights are initialized randomly, while in the second network CNN2 the weights are transferred from the trained VGG19 convolutional modules. Added layers are initialized randomly. The use of pre-trained neural network caused significant increase in the neural network accuracy, from 70.5% (CNN1) to 84% (CNN2).

In order to better evaluate the system performance the sensitivity (rate of properly classified malignant instances) and specificity (rate of properly classified benign instances) were calculated. In case of CNN1, the sensitivity=0.87 and specificity=0.54, while in case of CNN2, the sensitivity=0.87 and specificity=0.81. Higher sensitivity coefficient is desirable in medical applications, because incorrectly diagnosed malignant lesion can affect patient health. Fig. 3 shown ROC curves of classification process for CNN1 (blue), CNN2 (red) and random guessing (dotted line).

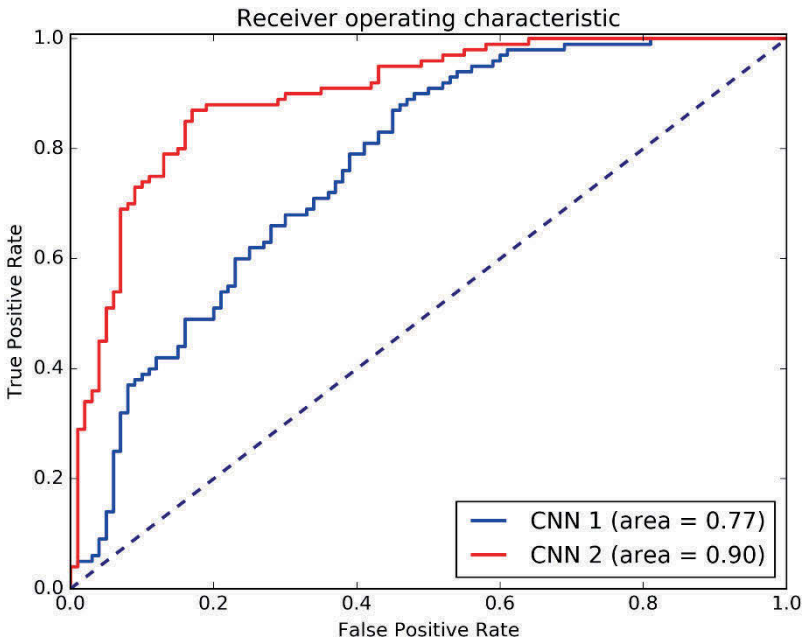


Fig. 3. ROC curve of classification system

4.2 Learned features

Convolutional layers act as the feature extractors. The images and their representation in convolutional layers are shown in Fig. 4. It is easy to notice that deep network learns how to segment skin, lesion, hairs in images (low level features). The medium and high level features are hardly interpretable by human.

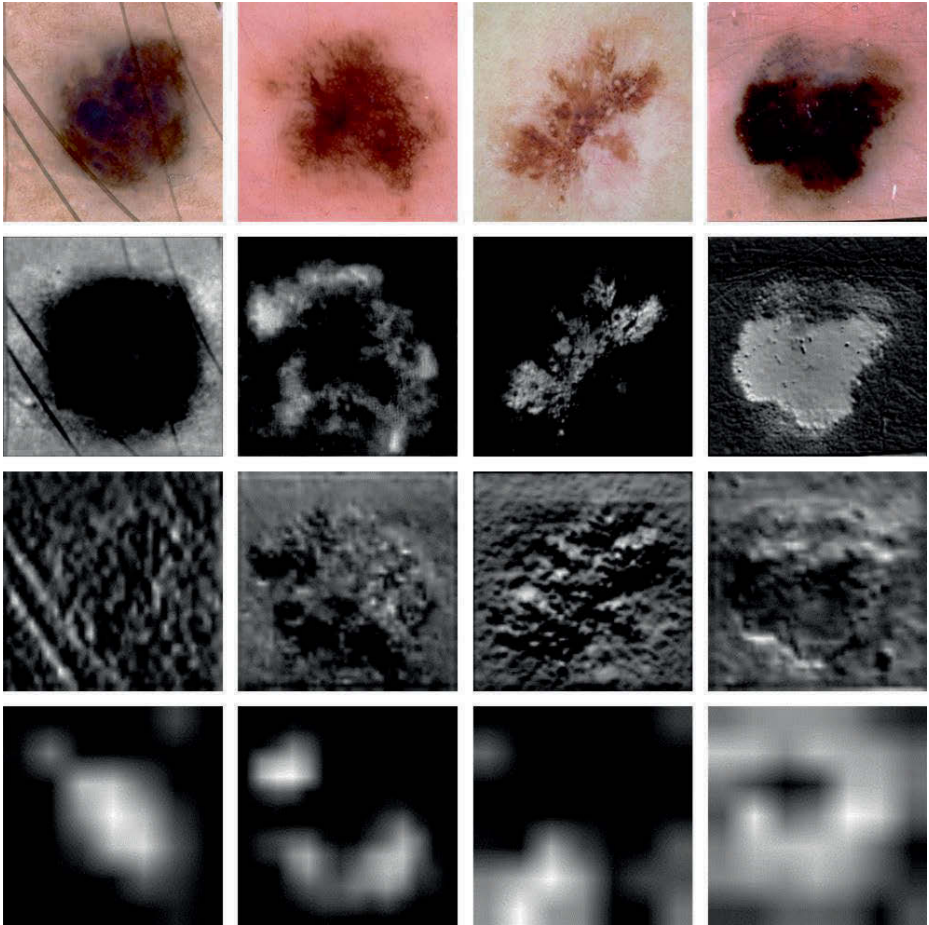


Fig. 4. Outputs of some convolutional filters. First row – original image, next rows – extracted features from low level to high level

4.3 Implementation details

System was built using Python Keras library [18] running on top of Theano library [19]. Keras allows for easy and fast prototyping of neural networks. During the simulations we used Nvidia CUDA library that allow parallel computing on GPU. Deep neural network was trained on GPU, while data augmentation was performed

by CPU. Training process took about 2h, longer than 2h training caused overfitting. Employment of GPU caused 40 times faster training compared with the CPU. Training was performed on a system equipped with: GeForce GTX 980Ti GPU with 6GB memory, Intel Core i7-4930K processor and 16GB RAM memory. The file that contains weight of the network takes up about 1.8 GB.

5 Summary and concluding remarks

Deep learning systems gain popularity in the field of image analysis. Rising computational power and availability of huge dataset allowed to benefits from deep learning algorithms.

In this paper, we proposed deep learning approach to skin lesion classification. Unlike traditional systems that employ hand-designed features, deep convolutional networks use trainable convolutional layers as feature extractor. Deep neural network can takes as input raw images with little preprocessing.

The use of pre-trained networks caused high increase in accuracy and allows minimizing adverse influence of small dataset. Achieved results are higher than that achieved by classic neural networks.

Good results encourage to applying deep learning in other medical images analysis tasks. Deep learning methods are data consuming, the rise of amount of data, provided by medical environment, will cause growth in accuracy.

Authors current research are focused on taking an advantage of SVM and clustering methods for medium and high level features analysis. Another aspect of consideration is finding even better deep learning architectures such as residual deep networks that achieved promising results in a field of image analysis.

References

1. Y. LeCun, Y. Bengio, and G. Hinton, 'Deep learning', *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
2. Y. LeCun, K. Kavukcuoglu, and C. Faret, 'Convolutional networks and applications in vision', in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, 2010, pp. 253–256.
3. A. Frome *et al.*, 'Large-scale privacy protection in Google Street View', in *2009 IEEE 12th International Conference on Computer Vision*, 2009, pp. 2373–2380.
4. A. Krizhevsky, I. Sutskever, and G. E. Hinton, 'Imagenet classification with deep convolutional neural networks', in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
5. W. Xiong *et al.*, 'Achieving human parity in conversational speech recognition', *ArXiv Prepr. ArXiv161005256*, 2016.
6. A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, 'Very Deep Convolutional Networks for Natural Language Processing', *ArXiv Prepr. ArXiv160601781*, 2016.

7. J. Chung, K. Cho, and Y. Bengio, 'A character-level decoder without explicit segmentation for neural machine translation', *ArXiv Prepr. ArXiv160306147*, 2016.
8. S. Gu, E. Holly, T. Lillicrap, and S. Levine, 'Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates', *ArXiv Prepr. ArXiv161000633*, 2016.
9. A. van den Oord *et al.*, 'Wavenet: A generative model for raw audio', *ArXiv Prepr. ArXiv160903499*, 2016.
10. D. Silver *et al.*, 'Mastering the game of Go with deep neural networks and tree search', *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.
11. J. Arevalo, F. A. González, R. Ramos-Pollán, J. L. Oliveira, and M. A. Guevara Lopez, 'Representation learning for mammography mass lesion classification with convolutional neural networks', *Comput. Methods Programs Biomed.*, vol. 127, pp. 248–257, Apr. 2016.
12. X. Yang *et al.*, 'A Deep Learning Approach for Tumor Tissue Image Classification', presented at the Biomedical Engineering, 2016.
13. V. Gulshan *et al.*, 'Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs', *JAMA*, vol. 316, no. 22, pp. 2402–2410, Dec. 2016.
14. H.-I. Suk, S.-W. Lee, D. Shen, and Alzheimer's Disease Neuroimaging Initiative, 'Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis', *NeuroImage*, vol. 101, pp. 569–582, Nov. 2014.
15. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, 'Dropout: A Simple Way to Prevent Neural Networks from Overfitting', *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.
16. 'ISIC Archive'. [Online]. Available: <https://isic-archive.com/>. [Accessed: 16-Nov-2016].
17. K. Simonyan and A. Zisserman, 'Very Deep Convolutional Networks for Large-Scale Image Recognition', *ArXiv14091556 Cs*, Sep. 2014.
18. F. Chollet, 'Keras'. [Online]. Available: <https://github.com/fchollet/keras>. [Accessed: 10-Nov-2016].
19. The Theano Development Team *et al.*, 'Theano: A Python framework for fast computation of mathematical expressions', *ArXiv160502688 Cs*, May 2016.

Part XII
Engineering education and teaching

Sharing the Experience in International Students Education: Robotics Program

Teresa Zielińska

Warsaw University of Technology, Faculty of Power and Aerospace Engineering ,
teresaz@meil.pw.edu.pl, <http://www.meil.pw.edu.pl>

Abstract. Quality of university education influences the economic competitiveness in knowledge-based societies. Unfortunately the curricula in engineering are often too slow to responding to the actual research and industrial needs, therefore fail in shaping the careers of tomorrow. The European Commission supports EU countries and higher education institutions in modernizing education programmes to provide graduates with university degrees and employable skills. This paper presents the experience gained during realization of specially designed international European Master on Advanced Robotics – EMARO+ (Erasmus+ program, former Erasmus Mundus program) supported by the European Commission. EMARO+ accepts only the best applicants from all over the World. The paper presents the program structure and the candidate selection criteria. The program is compared with other higher degree studies in robotics, available to international students. Examples of thesis are discussed. The current research trends in robotics are summarized to illustrate the needs of modern curricula in robotics. Short discussion of the key-factors influencing the quality of internationalization in the teaching process are considered.

Keywords: education, robotics, control and robotics, modern robotics

1 Introduction

In this paper we share our experience gained during the realization of the international European Master on Advanced Robotics – EMARO+ (earlier Erasmus Mundus program, now ERASMUS+) where competition is high and the local Robotics Master Program. The master programs available in the world are rarely devoted to robotics in general – they usually pertain only to certain aspects of robotics. EMARO was the first EU program in robotics – it started its operation in 2008, after the approval by the EU.

2 Acceptation method

The tuition fees and scholarships for some number of top applicants are supported by the European Union. Each year about 20 applicants from all over

the world compete for one fully supported placement. The applicants must submit their transcript of records, English language certificates, CV, motivation and supporting letters. Materials are independently analyzed and evaluated by all institutions running the program. The criteria are clearly defined together with the grading scheme. The highest weight in the evaluation (weight 40%) has the academic potential which takes into account the applicant GPA (Grade Point Average), and ranking at the end of the undergraduate studies. Additional achievements such as the publication of research papers, successes in robotics related competitions are also taken into account. The relevance of the obtained undergraduate degree to the EMARO/EMARO+ studies is the next evaluation item (weight 10%). Here the analysis checks whether the candidate has taken subjects important to the study of robotics, such as: classic mechanics, fundamentals of control, programming methods. Prior study of mechatronics or introduction to robotics are additional positive factors. Quality of the home institution, where the candidate completed the first degree studies, is also important (it contributes up to 20% to the overall grade). The remaining components that are evaluated are other aspects of the CV (e.g. participation in robotic competitions, publications), the motivation and recommendation letters. The grades obtained for English language tests are important too.

3 Program structure

The program consists of four balanced semesters with 30 ECTS assigned to each one of them. It is jointly realised by four European Universities: Ecole Centrale de Nantes (ECN), France (the coordinator), University of Genova (UG), Italy, Jaume I University (UJI), Castellon, Spain, Warsaw University of Technology (WUT), Poland. Taking into account the student's choice he/she spends the first year of studies in one of those universities, then the second year in another one. Moreover, the diploma work to be done during the fourth semester of studies can be done in the laboratories of one of the Asian partners, i.e. Keio University, Japan, or Shanghai Jiao Tong University, China. The best candidates of the program obtain funding support of their stay in Asia, including the travel costs. The program of the first year is the same in all partner institutions, but the second year is differentiated. Selecting the university for the second year the students fulfil their specific interests. ECN specializes mainly in sensor-based robotics, UG in the ICT tools for robotics, UJI in autonomous robot development methods, and WUT in mechanical design and biologically inspired robotics. The program of the first year provides fundamental knowledge: modeling and control of manipulators, computer vision for robotics, real time systems, signal processing, artificial neural networks, optimization methods, robot programming methods, mechanical design, mobile robots. The local language classes with an introduction to local culture and history are also offered. In the second semester, besides regular subjects, the students are also involved in the so called Group Project. In WUT each student must produce the kinematic model of a 6DOF manipulator, then they must implement the so obtained direct and inverse kine-

matic problem solution in the robot control system together with the synthesis of end-effector trajectories for the selected demo tasks. The results are tested in a real prototype. Quality and efficiency of the implementation are assessed. The program of second year at WUT includes advanced mechanical design, dynamics of multi-body systems, biomechanics, biorobotics and fundamentals of research planning. The last subject trains the students to do research work, including writing of a research paper and a research project proposal, as well as preparing and giving the presentation. The students are explained how to select the subject of the diploma thesis, taking into account their skills and interests. During this course the students decide about the subject of their diploma and complete the preliminary report, which is an introduction to the thesis. This part is jointly overseen by this course teacher and the diploma thesis supervisor. The topic of the diploma work is selected based on the written proposals prepared by the available supervisors.

The overall aim of EMARO+ is to deliver to the students a wide range of knowledge, stimulate team work and communication skills, encourage cultural tolerance and openness towards other cultures. The teaching modules have usually 5 ECTS. A significant portion of the program contains projects and laboratory work. Developing students' creativity is at the focus of the program, so that they will be well prepared for their future work in the emerging robotics fields. The students who have obtained interesting results are encouraged to publish them, writing papers with the assistance of their supervisors. Each year additional courses are offered by visiting professors. The students can also participate in elective courses.

4 Quality evaluation

After each semester quality evaluation of each course is made using an anonymous Internet questionnaire. Teaching methodology, quality of teaching materials and accessibility of teaching staff is assessed. The results are processed and passed on to the teachers. Additional written remarks are analyzed. The quality evaluation results are discussed once a year during the meeting of program management board. During this meeting students' progress, teaching and management problems are discussed to improve the program quality. Diploma thesis are co-supervised by the staff from the university where the student spent his/her first year of the studies and the supervisor from the second year institution. Wherever possible assistance in further careers is provided to the EMARO graduates. Those careers are monitored by staying in touch with the graduates. The graduates often undertake doctoral studies or jobs at research centers – some of them are directly employed by universities or by industry. The quality of the program is increased not only by improving the quality of teaching but also by providing additional services offered to the students. Each year there are several social meetings organized for them. Fresh students obtain help on arrival facilitating their introduction to the university life.

5 Quality in international education

EMARO+ fulfills basic requirements for quality in education [4, 5]. There are four key-factors influencing the quality of internationalized teaching process: **admission system** must provide students with the required and equilibrated background, the system must apply clear admission criteria with well defined and consequently kept thresholds, careful analysis of the quality of applicants through the submitted documents is important; **adaptation embedded into the teaching** process with special focus on support offered to fresh students and with local language classes offering introduction to the local culture; **quality of teachers** in the international classroom – they must be good academics with ample teaching and international experience and a thorough knowledge of the subject, open, flexible and interested in teaching international students; **quality of internationalized curricula**, in that – besides program contents – good quality of teaching materials, international supervision and availability of additional courses offered by visiting staff. It is important to create students' awareness and tolerance to different cultures, for example by showing interest in foreign students' culture and by joint celebration of local events or holidays. The assistance in arranging the internships and assistance in job search are also crucial for creating good quality studies.

6 Other robotic programs

There are several second degree programs devoted to areas related to robotics, such as Artificial Intelligence (offered by several universities in the Netherlands), Automation Engineering (Tampere Univ. of Technology, Finland), Real Time, Steering and Supervision (Ecole Centrale de Nantes, France), Machine Learning (KTH, Sweden), Autonomous Systems (Technical Univ. Dortmund).

Master in Computer and Automation Engineering – Robotics and Automation in Sienna University (Italy) besides typical subjects, similar to the core subjects of EMARO+ first year, offers the courses on human-centered robotics, sensors and microsystems and on bio-informatics. The program focuses on software engineering and on learning methods for creating robot intelligence. It lacks mechanical design and design of robot control systems. The program Systems, Control and Robotics offered by KTH, Stockholm, has the track Robotics and Autonomous Systems centered on control, sensing and perception. The core courses are: applied estimation, image analysis and computer vision, robotics and autonomous system, as well as artificial intelligence. The Robotics, Cognition, Intelligence program offered by the Technical University of Munich includes: informatics, mechanical engineering, electrical engineering, fundamental robotics cognition, and intelligent autonomous systems, as the core courses. Another program by the same university – Advanced Construction and Building Technology – Automation, Robotics, Services focuses on intelligent assistive technologies for industry and housing. The students learn how to integrate intelligent systems (including robots) in daily life and environments. Automation and Robotics, also

by Munich, includes the courses on advanced engineering mathematics, control theory and applications, computer systems, fundamentals of robotics, scientific programming in Matlab. The graduates are prepared to introduce and use robots as the key components in modern factories. This short overview indicates that stress is laid on educating specialists for developing existing robotic systems by enhancing abilities, autonomy and intelligence. Less stress is laid on educating the designers of control and mechanical systems.

7 Education and modern trends in robotics

Our days robotics is undergoing a major transformation. An intensive deployment of robots in new applications is expected, e.g. drones, especially in agriculture [13]. The future is seen in cloud robotics, freeing robots from their computational limits, by that increasing their abilities [14]. This will enable the development of sensory-rich robots with high adaptability and autonomy. Deep learning algorithms will allow the robots to access and process efficiently the data available in media, by that increasing the robots perception abilities, what also implies enhancing significantly their “understanding”. Robots will learn faster and more efficiently. Knowledge sharing between robots is another new trend. Different robots possessing different sensors and different acting abilities will share their “knowledge”, by that improving their learning processes. The future is open for personal robots being helpers or companions. In industrial production the future is seen in robots cooperating directly with humans – cobots. Biologically inspired robotics enables the development of novel control methods and non-conventional robot structures. Morphological computations are a new trend in bio-inspired robotics [2]. Behavior being the result of perception, processing and acting is seen here as the traditional approach. In morphological computations behavior depends significantly on the mechanical properties – the change of the robot shape (the so called morphological adaptation), and adequate arrangement of perception, actuation and processing units. Material properties are here very important, very often soft materials are used (thus a new term was coined – softrobotics) [1]. In reconfigurable, and/or soft robots the typical model-based approaches are insufficient. Morphological computations focus on the role of deformation of soft bodies and use reaction forces to obtain the needed behaviors. By this the control complexity is reduced, but the control philosophy is very different than before. The action is obtained not only by producing the actuating signals but by passive and active compliance and changes of the body shape. The analogies to soft bodied animals and plants are often used as reference in such research [3]. Biological CPG (Central Pattern Generator) generating motion rhythms based on mechanical synergies is referred as an example (and inspiration) for creating the motion patterns and control methods. Looking at robot perception and reasoning, neuromorphic computations methods are seen as a promising approach.

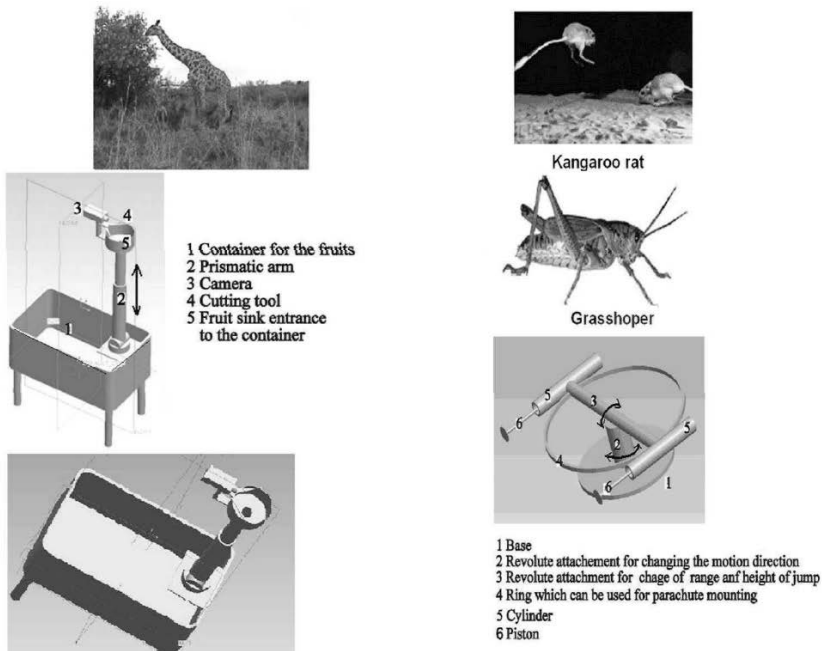


Fig. 1. Fruit picking robot (left). Jumping robot, for motion over undulating terrain (right)

8 EMARO+ and robotics of the future

In comparison to other robotics masters programs available in Europe, EMARO+ recognises the importance of mechanical design fundamentals. The program is biased towards developing creativity that should result in invention of new robot constructions. The Biorobotics and Biomechanics courses offered within EMARO+ in Warsaw University of Technology deliver the knowledge needed to follow recent robotics trends. Biorobotics course enables the students to learn: – classification of land animals from the point of view of locomotion, – basic features of locomotion including soft bodied animals [7], – biological fundamentals of motion control, – biologically inspired walking machines and their design solutions focusing on leg structures. The students learn how the animals adapted to the living conditions by modifying their body structures and motion abilities. The CPG based motion generation method for bipeds [6] is explained. Different structures of control systems, including hardware and software, are presented. Postural stabilization methods in animals and walking machines, reaction forces in biology and force control methods applied in walking machines are discussed [7, 8, 10–12]. Summary of biologically inspired robots is delivered. Possessing the knowledge about motion principles, sensing and motion synthesis in biology, students are requested to propose the concept of biologically inspired robot dedicated to some useful task [9]. Fig.1, left, shows a fruit picking robot

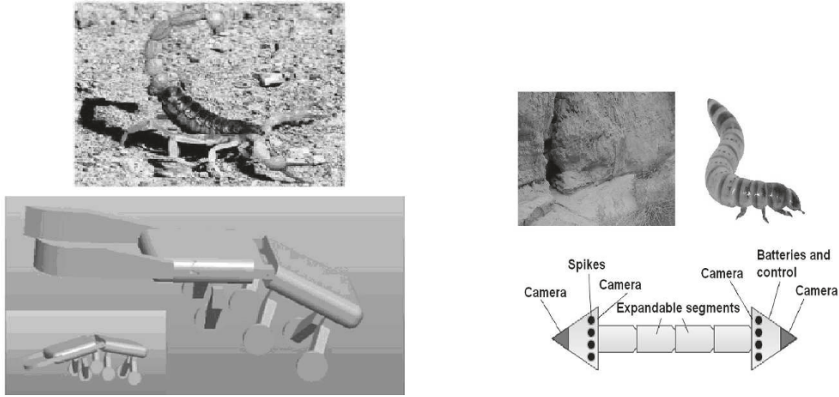


Fig. 2. Robot for rubble exploration (left). Cave climber (right).

– its inspiration was a giraffe. The robot has four supporting legs, a fruit container being its trunk, and a telescopic pipe forming its neck (a prismatic arm) delivering the picked fruits to the container. The robot illustrated in Fig.1, left, was inspired by a grasshopper and kangaroo rat. It was proposed as a small, lightweight robot for exploration. Up-down rotation (3) of the upper ring is used for controlling the height and range of jump, revolution to the side (2) influences the jump direction. Heavier base (1) creates the landing support. The motion is obtained by quickly releasing the pistons (6) mounted in the cylinders (5). For those two mentioned projects students used the knowledge gained during classes, about structure of vertebrate bodies and their locomotion principles. A robot inspired by a scorpion (Fig.2 left) was proposed for the purpose of searching an area affected by a natural disaster. The robot should be able to dig in the rubble and explore underground passages. Its trunk can bend, what provides better mobility. This mimics the motions of invertebrates. Interesting worm-like robot was proposed for exploration of rocks and caves (cave climber). The robot consists of several expandable segments connected by revolute joints. Both ends of the robots are equipped with heads containing batteries and control equipment. Four cameras are used for environment recognition (Fig.2, right). The adaptation to the surface is obtained due to the revolute connection of the segments. The grasp (surface attachment) is realized by spikes (or hooks) located at both ends of the body. The spikes are released by ejection springs and subsequently withdrawn by a reverse motion caused by an electromagnet. Robot applies peristaltic motion expanding and contracting its segments as the earthworm does (Fig.3). Besides expansion and contraction of the body segments an earthworms uses semi-legs to support the displacement. The proposed robot does not have such limbs, but uses its spikes as grippers when climbing.

Another idea is associated with a crawling robot – Fig.4, left, which moves like an inchworm or leech. Both ends of the body have suckers. Robot sucks itself to the ground by one body end and propels its body forward, then attaches to

the ground by the “front” sucker, releases the “hind” sucker and drags the body towards the attached part. The robot uses under-pressure to attach itself to the surface, while mechanically released and contracted springs push the body. The material out of which the body is made and the spring mechanism were not specified – unfortunately the students are not offered a courses on materials sciences.

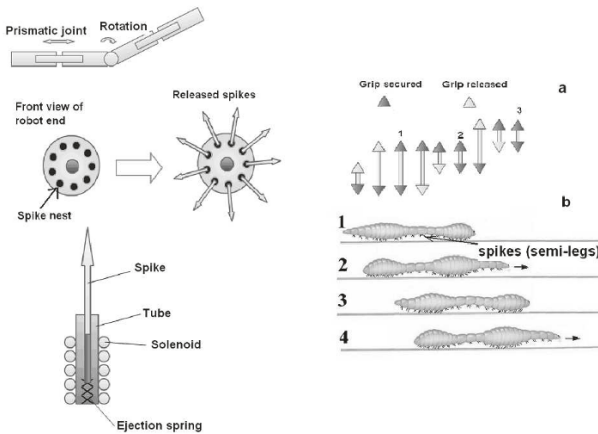


Fig. 3. Cave climber segments and illustration of spikes release system (left). Climbing method of the robot and peristaltic movement of the earthworm (right).

The last three projects are based on the studied by students principles of locomotion of: acelomates (animals without body cavity and without skeletons, which move by reshaping the body), pseudocoelomates (animals with elementary traces of body cavity, which also move by reshaping the body) and simple invertebrates which are coelomates (animals with external skeletons and body cavity). A robot which moves on a cobweb made of wires is inspired by a spider. It was proposed for silent security monitoring tasks – Fig.4. Wires are stretched above the monitored area. The robot is attached to them by its legs. Motion of the robot was obtained by an adequate backward-forward leg motions. However the mechanism attaching the legs to the wires was not specified. Given below is a selection of diploma topics completed within EMARO+/EMARO and Robotics program. It presents a sample of students’ research interest:

- Analysis of Postural Equilibrium Criteria: ZMP and CoP. Alejandro Gonzalez de Alba. Warsaw 2011.
- Robust Control of 3DOF Helicopter. Marcin Odelga. Nantes 2011.
- High Level Distributed Control for a Mobile Robots Wwarm. Abdul Salam El Khotob. Warsaw 2011.
- Machine Learning for Humanoid Robot Walking: Development of a Walking Strategy. Marcelo Gaudenzi de Faria. Warsaw 2012.

- Motion Planning and Simulation for a 6-DOF Earthquake Simulator. Victor-Cristian Rosenzweig. Warsaw 2012.
- Control Development for a Bicycle Robot Taking Into Account Slip Phenomenon. Nadezda Rukavishnikova. Warsaw 2012.
- 3D optical Flow Processing for Bio-inspired Robot Motion Control. Yanniss Rousseau. Warsaw 2012.
- Recursive Formulation for Efficient Forward Dynamics Simulation of Articulatedbody Robotic Systems. Ronald T.Mallea. Warsaw 2013.
- Whole-body Online Human Motion Imitation by a Humanoid Robot Using Task Specification. Louise P.Poubel. Nantes 2013.
- Online Imitation with Balance Constraint Using Optimization (dynamic approach). Anand Vazhapillu Sureshbabu. Nantes 2013.
- Visual Servoing of the Monash Epicyclic-parallel Manipulator. Alessia Vignolo. Warsaw 2014.
- Use of Artificial Vibrissae (whiskers) for Obstacles Avoiding. Fabio Vernamonte. Warsaw 2014.
- Minimalistic Tendon-Driven Hopping Using Bio-Inspired Actuator. Marie Charbonneau. Warsaw 2014.
- Control of Underwater Robotic Swarm with Electric Sense. Denajda Aliaj. Warsaw 2014
- Robot Motion Synthesis Using Human Ground Reaction Forces. Ramamoorthy Luxman. Warsaw 2015.
- Simulation and Design of Control System of Dual Arm Robot Actuated by Human Gesture Recognition. Kuldeeb Jashan. Warsaw 2016.

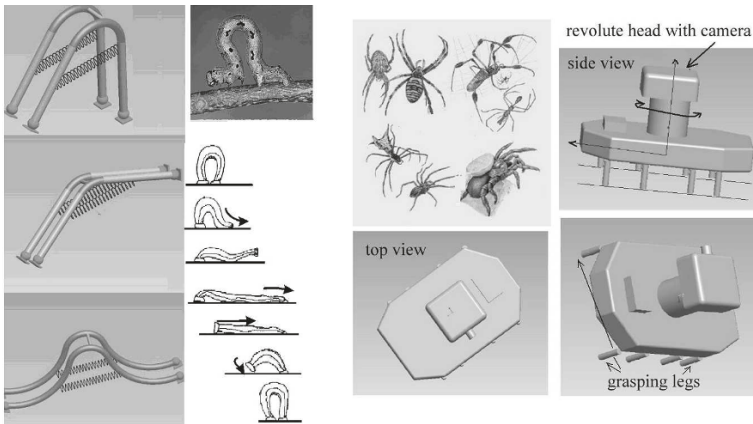


Fig. 4. Crawling robot (left). Spider robot (right).

9 Conclusions

According to popular opinion in the research community, the nearest future belongs to soft, reconfigurable robots. To follow such a trend the future roboticist must possess not only the typical engineering knowledge, but also the knowledge of biomechanics and biology. The expertise in new materials is also an advantage. Another developmental trend in robotics concerns the intensive use of Internet resources and efficient information sharing – for that good programming skills for serving modern methods of data mining and processing are needed. The attractiveness of an international teaching program in robotics depends on two major factors: the overall quality of the program, as it was described in section 5, and the attractiveness of the program contents, taking into account the current developmental trends. EMARO/EMARO+ aims to fulfill the above: by adequate management, by involving in the teaching process experts in robotics, and by cooperating with recognized universities and laboratories.

EMARO/EMARO+ graduates often enrol in doctoral studies, choosing research subjects related to biorobotics (e.g. hypnoidal hand design, bio-prosthesis, human gait analysis, development of novel robots).

References

1. J.Rossiter, H.Hauser: Soft Robotics - The Next Industrial Revolution? IEEE Robotics and Automation, Vol.23, pp.17-20. No.3. Sept. 2016.
2. S.Mintchev, D.Floreano: Adaptive Morphology. IEEE Robotics and Automation, Vol.23, pp.42-54. No.3. Sept. 2016.
3. C.I.Laschi, B.Mazzolai: Lessons from Animals and Plants. IEEE Robotics and Automation, Vol.23, pp.107-114. No.3. Sept. 2016.
4. Handbook of Quality. Support services related to the Quality of ERASMUS MUNDUS Master Courses and the preparation of quality guidelines. ECOTEC 2008.
5. Learning our Lesson: Review of Quality Teaching in Higher Education. OECD 2015
6. T.Zielinska: Coupled Oscillators Utilised as Gait Rhythm Generators of Two Legged Walking Machine. Int. J. Biological Cybernetics. vol.4, no.3, pp.263-273, 1996.
7. T.Zielinska: Motion Synthesis. Walking: Biological and Technological Aspects, CISM Courses & Lect. no.467. Eds. F.Pfeiffer, T.Zielinska. pp.151-187, Springer 2004.
8. Zielinska T., C-M,Chew, Kryczka P., Jargio T.: Robot Gait Synthesis Using the Scheme of Human Motion Skills Development. Mechanism and Machine Theory, Elsevier, vol.44.no.3, pp.541-558, 2009.
9. Bio-robotics projects by students: Ajay Kumar Tanwani, Bruno Kuljis Panzone, Arman Omrani, Sadat Fakhr, Michal J.Swiercz, Hisham Ezzat Mohammed, David Ezenwoye Maduabuchi. Warsaw University of Technology, 2010/11.
10. <http://www.liralab.it/teaching/ROBOTICA/docs/>
11. Yunhui Liu, Dong Sun: Biologically Inspired Robotics. Taylor and Francis 2011.
12. Bocolato G. et al.: 3D control for a Tentacle Robot, 3rd Int.Conf. Applied Mathematics, Simulation, Modelling, Circuits, Systems and Signals, pp.100-10, 2009.
13. T.Zielinska: Professional and Personal Service Robots, The Int. J. of Robotics Applications and Technologies, Vol.4, iss.1, 2016. IGI Global Publ., Hershey, USA.
14. Zielinski C., Szykiewicz W., Figat M., Szlenk M., Kornuta T., Kasprzak W., Stefanczyk M., Zielinska T., Figat J.: Reconfigurable Control Architecture for Exploratory Robots. Int.Work. Robot Motion and Control, RoMoCo, 2015. pp. 130-135.

Improving the Success Rate of Student Software Projects through Developing Effort Estimation Practices

Paweł Skruch, Marek Długosz and Wojciech Mitkowski

AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics,
Computer Science and Biomedical Engineering
Department of Automatics and Biomedical Engineering
al. A. Mickiewicza 30/B1, 30-059 Krakow, Poland
pawel.skruch@agh.edu.pl, mdlugosz@agh.edu.pl,
wojciech.mitkowski@agh.edu.pl

Abstract. Software projects have become an integral part of undergraduate and graduate engineering programs at technical universities. This trend in engineering education is driven by the increasingly usage of computer technology in almost every aspect of daily life. Based on the teaching experience in the last few years, it has been found out, that a surprising number of software related projects conducted by students at technical universities had been delivered late. Such situations usually stand for lower rating for the project, retest, or even prevent being promoted to the next study level. The result of the analysis has shown that lack of effort estimation or inaccurate estimation can be considered as the main reasons preventing successful and on-time project completion. In the paper, a course is proposed that covers required topics for understanding effort estimation concept. The majority of the course is based on the case studies of software engineering practices. The course is completed by a teacher guide to manage student software projects.

Keywords: estimation, effort, software project, student project

1 Introduction

Project work at technical universities is an integral part of the undergraduate and graduate student experience and it is also a useful supplement to the lectures. The project work shall expose students to the practical problems as they can become crucial in industry engineering programs the graduate students will work on. Although the students are ultimately responsible for their success or failure but teachers supervising student projects have a special responsibility to create an environment that shall maximise the opportunities for success.

The extensive use of electronics and software within the last decades has revolutionised the implementation and scope of many educational programs at technical universities. Existing undergraduate and graduate science and engineering

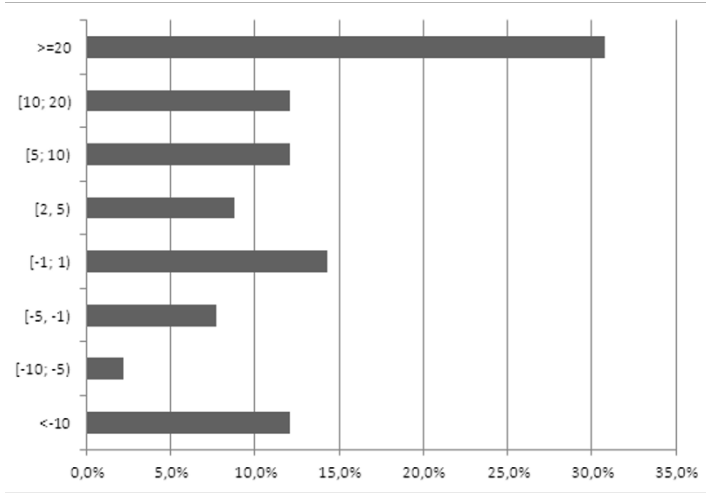


Fig. 1. Time delay (in days) in delivering the outcomes of 91 software related projects conducted by students of AGH University of Science and Technology. The results include 15 projects realised during the master thesis process, 12 projects as part of Bachelor program curriculum, and 60 projects conducted as part of other courses. Negative time delay means that the project had been finished before the due date.

programs need to incorporate more material on software engineering [14]. This is especially true for programs that rely heavily on computation, information, communications and software.

The initial motivation of the paper comes from the observation that a surprising number of software related projects conducted by students at technical universities is finished and delivered to the teacher late. The observation has been supported by the analysis of 91 software projects realised by the students of the Faculty of Electrical Engineering, Automatics, Computer Science and Electronics at the AGH University of Science and Technology, Krakow, Poland in the years 2007 – 2011. The results of the analysis are presented on Fig. 1 and they include 15 projects realised during the master thesis process, 12 projects as part of Bachelor program curriculum, and 60 projects conducted as part of other courses. Fig. 1 shows that only 36% of projects were finished on time or before the due date and the rest 64% were delivered late where: 9% had the delay between 2 and 5 days, 12% had the delay between 5 and 10 days, the same number of projects had the delay between 10 and 20 days, and almost 31% had been delivered with more that 20 days of delay.

The main motivation of the paper is to determine which factors have influence on timely completion of student projects in order to implement corrective actions to enhance their success rate. The industry data combined with the experience gained by Authors in the international software company allow formulating the following hypotheses: (1) implementation of effort estimation practices helps

defining the scope of work, supports better planning and controlling the project realised by the students; (2) accurate estimates and historical data are essential for successful execution of the project.

The problem of effort estimation has been the topic of hundreds of publications, numerous monographs, and several comprehensive textbooks, such as [1, 2, 4, 9, 15]. A systematic review that identifies 304 papers in 76 journals on the topic of software estimation can be found in [10]. These books and papers contain numerous techniques, models and approaches that have been developed within the past four decades to support accurate software estimates. The research being done in the area of software estimation is intended to help developers, testers and managers to make good decisions how to control the project to hit its target. Despite well-known phenomenon named as student syndrome there has not been a lot of attention to student education that would minimise the impact of this bad habit before a graduate student starts working on a real project. Student syndrome [7] occurs when the person with the task waits until the last possible moment to start. It happens very often that students spend their entire academic career waiting until the night before a project is due and then starting it. However, some authors [8], [12] have postulated that software engineering education can be enhanced through university-industry collaboration and implementation of many industry best practices. The idea can be applied to development methods, such as peer reviews [6], project management [3], [16], processes and tools [17]. This paper shows that implementation of effort estimation practices can also become a rewarding experience for students with significant improvement in their future employability.

The rest of the paper is organized as follows. Section 2 illustrates a typical flow of management activities for student projects and provides an overview of project parameters that have the biggest influence on required effort. In the following section, a course sequence incorporating estimation theory and practice is described. Section 4 contains implementation results including assessment data, a teacher guide to manage student software projects and an illustrative example. Conclusions complete the paper.

2 Student Software Project

To gain an understanding of a software project conducted by students in a university setting it is important to examine its characteristic that makes it different from other software projects developed by engineers in an industry environment. The student projects are analysed on the basis of numerous factors that have the biggest influence on the effort estimation.

2.1 Project Management

Fig. 2 illustrates a typical process management flow diagram for student projects. The process areas are documented pictorially as activities. Each activity block has identified responsibility roles that are denoted at the top of the block.

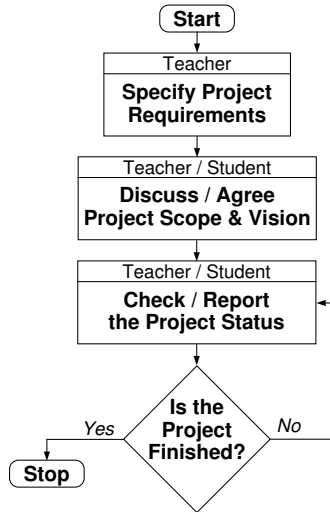


Fig. 2. A typical flow of management activities for student projects

In the first step, project requirements are specified. Once the project is approved by the local authorities, it is communicated to students. A teacher (supervisor) either accepts his first student who indicates his willingness to undertake a project or chooses one from a group of students. Next, the teacher sets up a meeting where the project scope and vision are discussed and the project work is committed by the student. It shall be noticed, that required effort needed to finalize the project is usually done by the teacher in the form of individual expert judgment. Formal analysis as well as the involvement of the student who will actually do the work usually does not take place. The student starts the work and reports the project status on regular or semi-regular meetings. After completing all tasks the project is evaluated by the teacher.

2.2 Project Characteristics

This part of the section describes some parameters of the Cocomo II estimating model [2] that have the biggest influence on overall effort needed to achieve the project targets. Understanding how project size, its complexity, kind of software being developed, and programming language influence the effort estimation helps improving its accuracy.

(1) *Project size.* The size of the software being built is considered as the most significant determinant of required effort [15]. A student software project can be characterised as a relatively small project that does not require extensive communication and extended documentation. Such project consists of approximately 1 to 5 students and lasts 1 to 6 months. The project size is usually below 100 000 lines of code therefore the effect of diseconomy of scale can be in most

cases neglected. Barry Boehm [1] defines diseconomy of scale as a decrease in productivity based on the increasing size of a project and an increasing number of project team members (programmers). Student projects are also not affected by so called *cone of uncertainty* [15] that defines statistically predictable levels of project estimate uncertainty at each stage of the project. However, this statement remains true under the assumption that the project requirements are well specified at the beginning of the project and remain stable during the development.

(2) *Complexity*. Time constraints imposed by the semester and limitation of number of students force the teacher to deal with not complex projects. It means that the software being developed does not need to follow industry standards, it is not safety critical, and it is not developed with the goal of later reuse. Moreover, budget and platform constraints usually do not play the key role in choosing software tools, architecture design, programming language, etc.

(3) *Kind of software being developed*. Kind of software being developed has significant influence on software productivity measure which is defined as the number of lines of code delivered per staff month that results in an acceptable and usable system [5]. This factor shall be definitely taken into account when assessing the overall effort as the student projects comprise a wide range of applications. Steve McConnell in his book [15] provides software productivity rates for common project types. These data show for example that a team developing software for internet or intranet applications might generate code 10 to 20 times faster than a team working for embedded system applications.

(4) *Programming language*. Students usually have the choice about the programming language they will use. Therefore this factor might become relevant in effort estimation. Some programming languages generate more functionality per line of code than others [15]. The project team's experience with the specific language and tools has about 40 % impact on the overall productivity rate of the project [15]. The choice of programming language might determine the choice of tool set and environment what in the worst case can increase the total project effort by about 50 % [2].

3 Improvement Course

This section describes the organisation of the course sequence in which the problem of effort estimation has been presented. The issues related to project effort estimation have not been discussed in a separate and independent unit but they have been incorporated into the lecture sequence describing the entire project life cycle. The course sequence under the common title "Computer Science and Electronics in Automotive" has been organised as elective lectures in cooperation with professional engineers of Delphi Technical Centre, Krakow, in Poland. The presented topics have strongly relied on real problems, real projects, and real applications taken from the automotive sector. The course has been offered to students of Automatics and Robotics specialisation area in the years

2007 – 2010 and Electronics and Telecommunications specialisation area in the years 2007 – 2009 as a part of the graduate studies.

The primary goal of the course is to provide students with the opportunity to confront and apply what they have learned in previous courses to the definition, design, construction, verification and validation of a significant real-world software system. The course topics focus on embedded systems that are widely used in automotive design and development from concept to production so they are to a large extent representative for the software engineering industry sector. Successful completion of this course will enable participants to understand the importance of having defined processes within an engineering organization and the rationale for process improvement.

The secondary goal of the course is to expose students to effort estimation problems as they play the crucial role in successful and on-time project completion.

The program of the elective lectures under the common title "Computer Science and Electronics in Automotive" is briefly outlined below. The first lectures are intended as a broad introduction into the subject of automotive electronic systems. Some of the typical processes, methods, techniques, and tools that are commonly used in the automotive industry for software development, hardware design, verification and validation are part of the following lectures. As the typical vehicle software system architecture is a distributed system, one of the lectures is devoted to the vehicle communication networks. The last part of the course highlights aspects related to the effort estimation as they play the key role in successful project execution. In order to achieve the course objectives, it is crucial, that the estimation theory is verified using real and practical examples.

The course is composed of lectures and class exercises with ample opportunity for participant questions and discussion. It comprises of 50% theory and 50% practice. The practical part includes demonstration of embedded electronics by representatives from the industry. In addition, students are required to form themselves into groups, which may vary in size from one to five students, and undertake one project. In the project work students are required to independently work out problem formulation of their own choice. At the end of the period of the project work, students are asked to prepare a project report. With the help of the e-learning platform (available via the link <http://moodle.cel.agh.edu.pl>) students have also the opportunity to further discuss their issues with lecturers, students, and practitioners.

4 Results

The course has been organised as a elective lecture so the students have decided which course they would like to choose. There was on average 15 different courses available and the students had to choose only one course. Fig. 3 presents the evolution of the number of enrolled students per academic year. The first edition was on the winter semester of the academic year 2008/2009. In this edition the course had 39 students enrolled from Automatics and Robotics (A&R) special-

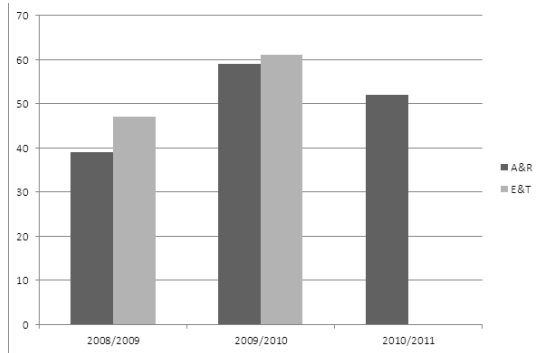


Fig. 3. Evolution of the number of enrolled students on the course per academic year

isation area and 47 students enrolled from Electronics and Telecommunications (E&T) specialisation area. In the following academic year 2009/2010 the course had, respectively, 59 and 61 enrolled students. In the last edition, that took place on the winter semester of the academic year 2010/2011 there was 52 students enrolled from A&R specialisation area. These numbers indicate that on average 50% of the total number of students have chosen this course.

At the end of the course, students were asked to complete the questionnaire concerning effectiveness of the course. The students were asked to assess and rate the course in the following categories: *Content*, *Lecturer*, *Student*, *General*. The *Content* category contains the questions concerning the organisation of the lectures, the effectiveness of visual aids and handouts, and the quality of the presented material and provided exercises. The results are presented in Table 1. In the next category, the students are asked to evaluate the way how the key concepts have been presented and illustrated by the lecturer. The results are included in Table 2. The *Student* category is devoted to assess and rate student's success in the course. Table 3 provides quantitative results of this assessment. Table 4 gives an overview about the overall course rate.

Based on the results obtained from the assessments, the course can be ranked as being above average and it can enable the students to use provided information in the future.

5 A Teacher Guide to Manage Student Projects

The use of effort estimation practices can substantially enhance the success of student projects. The teacher should take the advantage of that by following the improved process management flow diagram as illustrated on Fig. 4. The main changes in comparison to 'a typical flow' are indicated on the figure as *gray* rectangles. The improved methodology distinguishes effort estimation made by the teacher and the students as independent activities. Despite well-known evidence that estimates prepared by those who will do the work are better than estimates

Table 1. Evaluation of the course effectiveness: Content

The material was well organized			
Strongly Agree 38 %	Agree 61 %	Disagree 1 %	Strongly Disagree 0 %
Total: 120 answers			
Effectiveness of visual aids/handouts was sufficient			
Strongly Agree 32 %	Agree 66 %	Disagree 2 %	Strongly Disagree 0 %
Total: 121 answers			
Key concepts were reinforced by adequate exercises			
Strongly Agree 31 %	Agree 64 %	Disagree 5 %	Strongly Disagree 0 %
Total: 116 answers			
Lecture organisation (time, place) was convenient			
Strongly Agree 24 %	Agree 56 %	Disagree 15 %	Strongly Disagree 5 %
Total: 117 answers			

Table 2. Evaluation of the course effectiveness: Lecturer

The lecturer's delivery was clear			
Strongly Agree 49 %	Agree 47 %	Disagree 4 %	Strongly Disagree 0 %
Total: 118 answers			
Questions were answered by the lecturer in a consistent and clear way			
Strongly Agree 31 %	Agree 62 %	Disagree 7 %	Strongly Disagree 0 %
Total: 99 answers			

prepared by anyone else [13], estimates derived collaboratively by the students and their supervisor can eliminate misunderstanding of the project scope and the teacher's expectations. Indeed, if the variations in teacher' and students' estimates are not acceptable, then either the project scope shall be modified or the assumptions made by the teacher and the students are significantly different. The next important step is to reestimate the remaining effort a few times during the project as it is normal that reality usually differs from the plans. Reestimation can improve the accuracy of the effort needed for the remaining part of the project by using the actual productivity rate. If these differences do not affect the scheduled project completion time, there is no reason to worry about them, but it is important to be alerted early enough if the differences from initial plans constitute a danger for the project to be completed on time.

Table 3. Evaluation of the course effectiveness: Student

The presented information was useful for me			
Strongly Agree 24 %	Agree 64 %	Disagree 9 %	Strongly Disagree 3 %
Total: 119 answers			
I think that the lecture I have received will enable me to use these information or method			
Strongly Agree 19 %	Agree 65 %	Disagree 13 %	Strongly Disagree 3 %
Total: 119 answers			

Table 4. Evaluation of the course effectiveness: General

Overall course rating			
Excellent 17 %	Good 79 %	Sufficient 3 %	Unsatisfactory 1 %
Total: 118 answers			

6 Illustrative Example

This section provides a simple example that illustrates the way how the estimation can be done in practice.

It is considered a project that has been approved by the local authorities and communicated to students. A teacher has selected a student and set up a meeting to discuss the project scope. At the meeting, the student has been asked to create initial estimates.

The student has decided to separate large tasks into smaller tasks and estimate them separately. This estimation approach is known as decomposition by feature or tasks [15] and it is the practice of separating an estimate into multiple pieces, estimating each piece individually, and then recombining the individual estimates into an aggregate estimate. In the considered example, the project activities have been categorised as follows: *Requirement Analysis*, *Design*, *Graphical User Interface*, *Creating Database*, *Coding*, *Testing*, *Bug Fixing*, *Documentation* (see Table 5). To improve the accuracy of estimates, it is important to decompose estimates into tasks that will require no more than few days of effort. This level of granularity shall expose most of the relevant project activities.

The student has come back to the teacher with the estimation data that are presented in Table 5 in column *Most Likely Case*. Using his experience he has noticed that some tasks in the project can be done faster and other tasks, because of some unexpected events, can take longer than expected. Therefore he has created best case and worst case estimates that shall cover these situations (the columns *Base Case* and *Worst Case* in Table 5). Creating best case and worst case estimates stimulates thinking about the full range of possible outcomes. Single-point estimates can be then taken in the middle of the range from best case to worst case or the technique called the Program Evaluation and Review

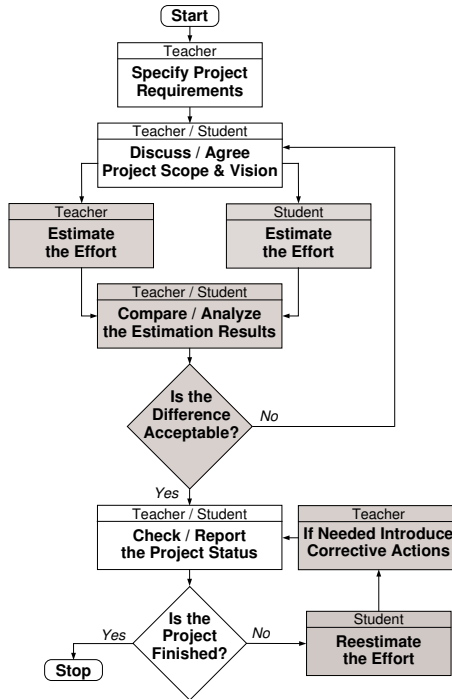


Fig. 4. Improved flow of management activities for student projects. Additional activities in comparison to 'a typical flow' are indicated as *gray* rectangles.

Technique (PERT) [11] can be used to compute so called expected case (the column *Expected Case* in Table 5):

$$EC = \frac{BC + (4 \times MLC) + WC}{6}. \tag{1}$$

Once the project has been started the student has kept the list of his estimates and filled in the actual results (the column *Actual Effort* in Table 5). Here it should be noticed that actual data can refined the personal estimation abilities in reestimation sessions. Historical data obtained from the past projects as well as the data from the current projects are essential to improve estimation accuracy. The historical data account for organisational (academy) influences, avoids subjectivity and unfounded optimism, and provide the information about real student productivity.

7 Conclusions

Project effort estimation is not a new phenomenon. It is the evolution of the best practices which have been refined over the past forty years. Since the effort

Table 5. Example of spreadsheet for project effort estimation. The effort is given in man-days.

Feature or Task	Estimated Effort				Actual Effort
	Best Case (BC)	Most Likely Case (MLC)	Worst Case (WC)	Expected Case (EC)	
Requirement Analysis	1.0	2.0	3.0	2.0	2.5
Design	1.0	1.5	3.0	1.7	1.5
Graphical User Interface	2.0	3.0	5.0	3.2	3.0
Creating Database	0.5	1.0	2.0	1.1	1.0
Coding	5.0	7.0	10.0	7.2	8.0
Testing	2.0	3.0	4.0	3.0	2.5
Bug Fixing	1.0	2.0	5.0	2.3	2.5
Documentation	2.0	4.0	7.0	4.2	4.0
Total	14.5	23.5	39.0	24.6	21.0

estimation activities are standard practices in the industry it would be natural to emphasize this problem in the university environment. In the paper, a course sequence has been proposed that expose students to exciting and important issues that impact software engineer's life on a daily basis. The information presented in the lectures, with particular emphasis on effort estimation practices, can be useful for both students and teachers since all of them make the contribution to the project success. It shall be also added, that several students received jobs in big international software companies after completing this course.

References

1. Boehm, B.: Software Engineering Economics. Prentice Hall, Englewood Cliffs, USA (1981)
2. Boehm, B., Abts, C., Brown, A., Chulani, S., Clark, B., Horowitz, E., Madachy, R., Reifer, D., Steece, B.: Software Cost Estimation with Cocomo II. Prentice Hall, Englewood Cliffs, USA (2000)
3. Cheng, Y., Lin, J.C.: A constrained and guided approach for managing software engineering course projects. IEEE Transactions on Education 53(3), 430–436 (2010)
4. DeMarco, T.: Controlling Software Projects: Management, Measurement and Estimation. Prentice Hall, Englewood Cliffs, USA (1982)
5. Department of the Air Force, Software Technology Support Center: Guidelines for successful acquisition and management of software-intensive systems: weapon systems, command and control systems, management information systems. Version 3.0. <http://www.stsc.hill.af.mil/resources> [09 August 2012] (2000)
6. Garousi, V.: Applying peer reviews in software engineering education: an experiment and lessons learned. IEEE Transactions on Education 53(2), 182–193 (2010)

7. Goldratt, E.: *Critical Chain*. The North River Press, Massachusetts, USA (1997)
8. Harrison, J.: Enhancing software development project courses via industry participation. In: *Proceedings of 10th Conference on Software Engineering Education & Training*, Virginia, USA. pp. 192–203 (1997)
9. Jones, C.: *Estimating Software Costs*. McGraw Hill, New York, USA (1998)
10. Jorgensen, M., Shepperd, M.: A systematic review of software development cost estimation studies. *IEEE Transactions on Software Engineering* 33(1), 33–53 (2007)
11. Kerzner, H.: *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. John Wiley & Sons, New York, USA (2003)
12. Kornecki, A., Hirmanpour, I., Towhidnadjad, M., Boyd, R., Ghiorzi, T., Margolis, L.: Strengthening software engineering education through academic industry collaboration. In: *Proceedings of 10th Conference on Software Engineering Education & Training*, Virginia, USA. pp. 204–211 (1997)
13. Lederer, A., Prasad, J.: Nine management guidelines for better cost estimating. *Communications of the ACM* 35(2), 51–59 (1992)
14. Long, L.: The critical need for software engineering education. *The Journal of Defense Software Engineering* 21(1), 6–10 (2008)
15. McConnell, S.: *Software Estimation: Demystifying the Black Art*. Microsoft Press, Washington, USA (2006)
16. Pfahl, D., Laitenberger, O., Ruhe, G., Dorsch, J., Krivobokova, T.: Evaluating the learning effectiveness of using simulations in software project management education: results from a twice replicated experiment. *Information and Software Technology* 46(2), 127–147 (2004)
17. Skruch, P.: An educational tool for teaching vehicle electronic system architecture. *International Journal of Electrical Engineering Education* 48(2), 174–183 (2011)

Author Index

A

Amanowicz, Marek, 611
Augusta, Petr, 734

B

Babiarz, Artur, 434
Bąk, Adam, 722
Banach, Banach, 77
Bania, Piotr, 55, 818
Baranowski, Jerzy, 818
Bartoszewicz, Andrzej, 4
Bauer, Waldemar, 818
Bazydło, Piotr, 503
Borawski, Kamil, 118
Boski, Marcin, 754
Bożejko, Wojciech, 370, 703
Brasel, Michał, 21
Bryniarska, Anna, 681
Byrski, Witold, 536

C

Cellmer, Agata, 77
Chaber, Patryk, 315, 378
Chiliński, Jędrzej, 818
Cichy, Błażej, 734
Ciecierski, Konrad A., 838
Ciekowski, Maciej, 45
Czerwiński, Kamil, 599

D

Długosz, Marek, 168, 178, 224, 869
Długosz, Rafał, 787
Domański, Paweł D., 128
Domek, Stefan, 442
Drzewiecki, Marcin, 570
Duda, Jozef, 89
Duzinkiewicz, Kazimierz, 241, 344, 631
Dworak, Paweł, 21

E

Emirsajłow, Zbigniew, 98

G

Gałkowski, Krzysztof, 734, 744
Gawin, Edyta, 425
Gawron, Tomasz, 473
Ghosh, Sandip, 21
Gniewkowski, Łukasz, 703
Grega, Wojciech, 138
Grochowski, Michał, 631, 828, 848

H

Hasiewicz, Zygmunt, 527
Hojda, Maciej, 483

J

Jabłoński, Mateusz, 590
Jarmakiewicz, Jacek, 611
Jaroszewski, Krzysztof, 661
Jaskuła, Marek, 4
Jastrzębski, Marcin, 388
Jezierski, Andrzej, 65
Juszczuk, Dymitr, 344

K

Kabziński, Jacek, 388
Kacprzyk, Janusz, 503
Kaczorek, Tadeusz, 401
Karla, Tomasz, 344
Kasprzyk, Jerzy, 214
Klamka, Jerzy, 455
Klemiato, Maciej, 364
Kluska, Jacek, 712
Kobylarz, Anna, 631
Kogut, Krzysztof, Dr., 190
Kofek, Krzysztof, 190
Komuta, Tomasz, 493

Kościelny, Jan Maciej, 550
 Kowalczyk, Marcin, 353
 Kowalczuk, Zdzisław, 722
 Kowalewski, Adam, 98
 Kowalów, Damian, 621
 Kozdraś, Bartłomiej, 527
 Krakowiak, Anna, 98
 Krauze, Piotr, 214
 Kreft, Wojciech, 560
 Kruszewski, Damian, 774
 Kryjak, Tomasz, 353
 Kuczkowski, Lukasz, 641
 Kulczycki, Piotr, 774
 Kulkowski, Karol, 631
 Kwasięroch, Arkadiusz, 828, 848

L

Łakomy, Krzysztof, 251
 Lasiecka, Irena, 3
 Lasota, Krzysztof, 503
 Ławryńczuk, Maciej, 315, 378, 599
 Łęgowski, Adrian, 434
 Leśniewski, Piotr, 4
 Liwski, Marcin, 669

M

Mączka, Tomasz, 712
 Majdzik, Paweł, 281
 Mandat, Tomasz, 838
 Markiewicz, Paweł, 178, 224
 Marusak, Piotr M., 128
 Maurer, Helmut, 799
 Mazur, Krzysztof, 291
 Michałek, Maciej Marcin, 251, 473
 Mikołajczyk, Agnieszka, 828, 848
 Mitkowski, Wojciech, 168, 460, 869
 Mozaryn, Jakub, 65
 Mzyk, Grzegorz, 527

N

Novikov, Dmitry A., 693

O

Oprędkiewicz, Krzysztof, 425

P

Paszke, Wojciech, 754
 Patan, Maciej, 621
 Pawelczyk, Marek, 291
 Pawluszewicz, Ewa, 415
 Pazera, Marcin, 651
 Piątek, Paweł, 818

Piesik, Emilian, 669
 Pietrala, Mateusz, 4
 Piotrowski, Robert, 77
 Plamowski, Sebastian, 335

R

Rafajłowicz, Ewaryst, 261
 Rafajłowicz, Wojciech, 108, 261
 Rogers, Eric, 734, 744, 754
 Rosół, Maciej, 190
 Rotter, Paweł, 364
 Rutkowski, Tomasz A., 241
 Rzońca, Dariusz, 303

S

Sadolewski, Jan, 303
 Skruch, Paweł, 155, 168, 178, 224, 787, 869
 Skubalska-Rafajłowicz, Ewa, 580
 Smierzchalski, Roman, 641
 Sokołowski, Jan, 98
 Sokółski, Paweł, 241
 Stec, Andrzej, 303
 Stetter, Ralf, 281
 Śmierchalski, Roman, 514
 Świder, Zbigniew, 14, 303
 Sulikowski, Bartłomiej, 744
 Suski, Damian, 65
 Swierniak, Andrzej, 799
 Syfert, Michał, 550
 Sylwester, Frączek, 200
 Szewczyk, Przemysław, 271
 Szyber, Anna, 550

T

Talaska, Tomasz, 787
 Tarnawski, Jarosław, 344
 Tatara, Marek, 722
 Tatjewski, Piotr, 33
 Trybus, Bartosz, 303
 Trybus, Leszek, 14, 303
 Turnau, Andrzej, 190
 Tutaj, Andrzej, 138

U

Uchroński, Mariusz, 370

W

Wachel, Paweł, 527
 Wiktorowicz, Krzysztof, 764
 Wilczyński, Przemysław, 514
 Winiarski, Tomasz, 493
 Witczak, Marcin, 651

Witkowska, Anna, 514
Wodecki, Mieczysław, 370, 703
Wojtulewicz, Andrzej, 325
Wrona, Stanisław, 291
Wyrwał, Janusz, 214

Z
Żabiński, Tomasz, 712
Zdzisław, Kowalczuk, 200
Zielinska, Teresa, 859
Zieliński, Cezary, 493